An Empirical Study on Factors Affecting Learning of Programming: The Impact of Learning Culture on Outcomes

by

Ritu Sharma

Thesis

Submitted to Flinders University

for the degree of

Doctor of Philosophy

College of Science and Engineering

09 August 2018

CONTENTS

DECLARATION

I certify that this thesis does not incorporate without acknowledgment any material previously submitted for a degree or diploma in any university; and that to the best of my knowledge and belief it does not contain any material previously published or written by another person except where due reference is made in the text.

ACKNOWLEDGEMENTS

Completion of this doctoral dissertation was possible with the support of several people. I would like to express my sincere gratitude to all of them. First of all, I am extremely grateful to my research guide, Dr. Haifeng Shen, Senior Lecturer, School of Computer Science, Engineering and Mathematics, Flinders University, for his valuable guidance, patience, motivation, enthusiasm, and immense knowledge. I would like to thank him for encouraging my research.

I am also thankful to Dr. Robert Goodwin under whose guidance I started this research. His guidance, motivation and patience along with vast experience helped me in the time of research and writing of this thesis. Both my supervisors are exceptionally supportive and their scholarly inputs and consistent encouragement helped me throughout the course of this study. I could not have imagined having better advisors and mentors for my Ph.D. study.

I am very grateful to Dr. Maninder Singh, for supporting this research by allowing me to conduct this research. This research would not have been possible without his support.

My family has been very important to me in the pursuit of this project. I owe a lot to my parents, who encouraged and helped me at every stage of my personal and academic life and longed to see this achievement come true. They have been motivating and encouraging me to study and rise since I was five years old. They are the ultimate role models.

Most importantly, I wish to thank my husband Mandeep Sharma and Son Tanish, who have always been an inspiration. A special thanks to my brother and sister-in-law for their guidance.

I would also like to thank my teacher and mentor Mr. Sanjay Kotha. As a mentor he taught me more than I will ever be able to give him credit for here. Mr. Balhinder Singh Bawa for his excellent teaching in my Master’s Degree. I feel privileged to be his student once, a feeling that every student of his will second.

Above all, I owe it all to Almighty God and my Guru Sant Baljit Singh Ji for granting me the wisdom, health and strength to undertake this research task and enabling me to its completion.

ABSTRACT

The processes and emotions undertaken when learning programming are considered to be inherently different from those experienced when learning other topics, subjects and courses. Whilst many past studies have analysed factors that may affect learning of programming, the discussions that started around the 1970s are still on-going today. Current educators are still trying to define the key factors in learning programming. However, as most of these factors have been studied in a specific learning culture, it is imperative now to understand whether both educational and cultural differences can influence the learning of programming.

In this study, we report the results of a comparative study between two universities representing different learning cultures: one in Australia and one in India. Each takes a particular approach to the culture of learning programming and therefore each acts as a foil to the other, supporting and making visible the elements of this thesis. A learning culture generally refers to how teachers select their pedagogy and how students receive instruction. In our study, it consists of the teaching methodology used to teach programming, the assessment structure, the attendance structure and the examination structure. The factors considered are prior programming experience, gender, family background, preliminary preparation and revision, family background and study choices. The need for a strong comparison between university students who have different learning cultures motivates us to choose Australia and India as the countries in which to conduct this study. It is important to study if the factors affecting learning programming are similar or different in these two universities, as learning programming is considered a difficult task, but most of the research conducted to date has focused on a single learning culture. The Universities chosen for this study are Flinders University from Australia and Thapar University from India. These shall be referred as Australian University and Indian University throughout the thesis.

The study has been conducted in two parts. The first part analyses the factors chosen on the basis of factors described in Tinto’s model. The second part of the study is formed on the basis of central part of Tinto’s model.

The results of the first part of the study show that prior programming experience, gender, reason to study programming, attendance, and revision have different effects, while activities

performed in the lecture theatre and preliminary preparation before lectures and laboratories have parallel effects in the two universities. These findings help us gain insight as to whether certain factors are dependent/independent of learning culture, so that educators can focus on the specific factors that will help students learn programming more effectively in the context of a particular learning culture. If the factors are dependent on culture, then the factors that positively affect student performance in a particular learning culture may be taught in a manner that will positively affect the performance of students.

The research will also be valuable to the lecturers teaching programming as the identified factors may be built into the teaching methodology. Before making the comparison, the similarities and differences between the two chosen Universities were studied in terms of methods of teaching programming, education culture, examination structure and assessment structure.

The second part of the study was designed on the basis of the central part of Tinto's conceptual model. It studied the use of social media as a tool to enhance student engagement and serve as an additional resource of peer to peer interaction and social integration in the process of learning programming. This approach is then compared with the discussion feature of a Course Management System (CMS) system used at Australian University.

Various forms of social media were studied and Facebook was chosen. The secondary purpose of this study was to explore if the use of CMS or Facebook may help improve student engagement and serve as an additional source of support while learning programming. Facebook can be helpful to those students who find themselves unable to solve a particular problem. Thus, early access to help may ease the process of learning programming, save time and provide motivation to progress further. It may also be beneficial to the lecturer as a mechanism for tracking the students' progress. Monitoring engagement in social media may help identify those students who are not involved in social learning, so that appropriate support can be provided for those who need it.

The research model used for this research, defined the framework of the research questions, which were in turn tested against the null and alternative hypothesis. The data for the first part of the study was collected from both Australian University, Australia, and Indian University, India, across three academic semesters. The total number of respondents from Australian University was 198 and the total number of participants from Indian University was 94. The combined results of the three semesters for each University were merged for the analysis performed on the combined data of each University. The results suggest that most of the factors affecting the performance of students are different for each University. This suggests that the factors affecting the learning of programming are context- and culture-dependent.

The other part of the study was conducted across four semesters and was open to Australian University students only as the study couldn't be conducted at Indian University for ethical reasons. The use of mobile phones is prohibited in the academic area, which made it difficult to conduct this study. It investigated the use of a Facebook group as an additional resource for learning programming alongside CMS. This study concluded that both Facebook and CMS may enhance student engagement and serve as additional resources for peer to peer interaction and social integration in the process of learning programming, but the students preferred CMS over Facebook.

After the completion of the study, some significant factors were extracted for both Universities, which may prove helpful to the process of learning programming for first year students. A few common factors were also identified, which suggest that focusing on those factors may be beneficial to students. From the second part of the study, it was learnt that CMS as well as Facebook may help improve student engagement and serve as an additional resource in learning programming. It was also found that the students preferred CMS provided by the university, as compared with Facebook, as a key mechanism through which to communicate with their peers. Thus, it may be helpful to students if they are encouraged to use CMS to communicate with each other, and beneficial to the lecturer who can use CMS to keep a track of the students' discussions and learning.

LIST OF ABBREVIATIONS AND CONVENTIONS IN WRITING

ACSW Australasian Computer Science Week

ACM Association of Computer Machinery

ATAR Australian Tertiary Admission Rank

BASIC Beginners All Purpose Symbolic Instruction Code

BCS British Computer Society

CMS   Course Management System

CSR- GHRDC Competition Success Review Global Human Resource Development Centre

DF Degrees of freedom

EDUCAUSE Center for Applied Research (ECAR)

E-learning electronic learning

FLO Flinders learning Online

JPL Java programming Laboratory

ID Intellectual Development

IEEE Institute of Electrical and Electronics Engineers

LC Leaving Certificate

LTM Long-term memory

M Mean

Moodle Modular Object-Oriented Dynamic Learning Environment

NTCET Northern Territory Certificate of Education and Training

p p-value

SACE South Australian Certificate of Education

SE Standard deviation

STEM Science, technology, engineering and mathematics

SNS Social Networking Sites

STM Short-term memory

QQ plots quantile-quantile plots

Wald Wald chi-square statistic

ESL English Second Language

LIST OF FIGURES

LIST OF TABLES

PUBLICATIONS

Sharma, R., Goodwin, R., & Wilkinson, B. (2014, January). Can a Facebook Group Serve as an Additional Learning Resource for Introductory Programming Students? In International Conference on Infocomm Technologies in Competitive Strategies (ICT). Proceedings (p. 114). Global Science and Technology Forum.

Ritu Sharma, Haifeng Shen, and Robert Goodwin. 2016. Voluntary participation in discussion forums as an engagement indicator: an empirical study of teaching first-year programming. In Proceedings of the 28th Australian Conference on Computer-Human Interaction (OzCHI '16). ACM, New York, NY, USA, 489-493. DOI: https://doi.org/10.1145/3010915.3010967

Sharma, R. and Shen, H., 2018. Does Education Culture Influence Factors in Learning Programming: A Comparative Study between Two Universities across Continents. *International Journal of Learning, Teaching and Educational Research*, *17*(2).

Sharma, R. and Shen, H., 2018. The Interplay of Factors Affecting Learning of Introductory Programming: A comparative study of an Australian and an Indian University.*13th International conference IEEE ICCSE2018*

CHAPTER 1 : INTRODUCTION

This chapter introduces the motivation for this study, and presents the research aims, research questions, research methodology, research scope, research significance and how this dissertation is organised. This chapter also gives a brief introduction to the problem as stated in the studies conducted specifically in the area of learning programming.

1.1 Statement of problem

The underlying problem found in the literature is how to engage students with learning programming, when they find the process alienating. Few computing educators of any experience would argue that students find learning to program easy. Most teachers will be accustomed to the struggles of their first year students, as they battle in vain to control the most basic of skills and many would have seen students in later years carefully choosing options so as to minimise the risk of being asked to undertake any programming (Jenkins, 2002). Many of the factors that may affect learning programming have been explored in the literature, yet students continue to struggle to learn programming. This study aims to identify the factors that were analysed in the past and explore some additional factors that may affect current learning of programming. Tinto’s conceptual model forms the basis of this study (Tinto, 1975). The factors were carefully chosen and studied, based on Tinto’s model. This study also proposes to analyse the effect of these factors for universities within different learning cultures. A comparison was made between Australian University, Australia, and Indian University, India, to analyse if the factors chosen for study have similar or dissimilar effects on performance of the students in learning programming in both these diverse universities. A learning culture generally refers to both teacher pedagogy and student mechanisms for learning. In our study, the learning culture consists of the teaching methodology used to teach programming, the assessment structure, the attendance structure and the examination structure. The effect of these chosen factors on student performance in terms of grades/scores was studied.

The second part of the research studied peer-to-peer interaction and social integration, which constitute the central part of the Tinto’s model. The second part of the study was only conducted at Australian University, as support for this part of the project could not be achieved from Indian University. The peer-to-peer interaction and social integration were studied by analysing the use of a type of CMS provided by Australian University named FLO (Flinders Learning Online) and was compared with a form of social media named Facebook. The use of these two systems to improve retention by enhancing student engagement and to serve as an additional resource for learning programming by introducing peer-to-peer learning within and outside the classroom by using CMS and Facebook was also studied. As a result, it was found that the use of both the CMS chat feature and Facebook may enhance student engagement and serve as an additional source of peer to peer interaction and social integration in the process of learning programming. The study also discovered that the students preferred communicating with each other on the CMS provided by the university; the presence of a lecturer(s) on the communication network may further motivate students to initiate conversations and ask for help when required. CMS may prove beneficial to the students by providing them with an additional source of help while learning programming, and the benefits may extend to the lecturer who is able to learn about the progress of the students, enabling them to offer targeted students additional assistance.

1.2 Introduction to Research Problem

In computer science, an expected outcome of a student's education is programming skill (McCracken et al., 2001). Programming is one of the many skills that computer science students are expected to master. To be technology literate, it is argued that learning to program still plays an important role (Lau and Yuen, 2011). In addition, most science, technology, engineering and mathematics (STEM) programs expect the students to acquire programming skills as a part of their education (McCracken et al., 2001). This makes learning programming crucial to STEM students. The literature suggests that programming is considered a difficult skill to acquire. A large number of students face difficulty in learning programming. Educators in the history of teaching programming have identified programming as a skill considered challenging by the students. Educators are continuously striving to facilitate the learning process of programming. Teaching introductory programming at university level has been the basis for many lively discussions in computer science (Moström, 2011). Experts suggest that programming is a core subject within computer science curricula and is considered difficult to learn.

It has previously suggested that limited numbers of students find learning to program easy (Jenkins, 2002). This problem is particularly prevalent for students learning programming for the first time. A previous study also suggested that teaching beginners to program is challenging (Caspersen and Kolling, 2009). Studies also confirm that this problem is not only prevalent in a particular country but many countries where programming is taught encounter the same problems, irrespective of demographics, race and ethnicity.

McCracken et al. conducted a multi-national, multi-institutional study of assessment of programming skills of first-year Computer Science students and found that the problems they observed with programming skills seemed to be independent of country and educational system (McCracken et al., 2001). An entire volume of papers, called 'Studying the Novice Programmer', also documented the difficulties of learning to program (Collins et al., 1991) .

Programming courses are often required by Computer science students as a part of their degree and they often continue to struggle to complete the programming courses. Indeed, anecdotal evidence suggests that many students – not just Computer science students – struggle in programming principles courses (Woszczynski et al., 2005b). Academics in universities teaching programming courses believe that their introductory programming classes are not working as well as they should be and this belief leads to reluctance from academics to talk about the introductory programming class outside their own institution (Lister, 2005).

Students have been facing problems in learning programming since it started to be taught at Tertiary level. Studies on problems in learning programming were conducted as early as the 1970s, yet a large number of researchers are still trying to solve this problem (Hagan et al., 1997). A large number of research papers are still published in this area, which suggests that the problem has yet to be solved, although progress has been made.

Programming education is a great challenge, partly because programming seems to be intrinsically difficult. In spite of more than forty years of experience, teaching programming is still considered a major challenge. A countless number of sources in the literature affirmed the difficulties of teaching and learning programming (Caspersen and Kolling, 2009). A search in the ACM (Association of Computer Machinery) of the phrase "problems in learning programming" yielded 6000 results, although some may not be relevant, and it was considered one of the seven great challenges in computing education by the British Computer Society (BCS), which identified seven great challenges in computing education in 2005. The literature has reported this as a universal problem that has motivated many researchers to propose various methodologies, tools and pedagogies to help students (Seyal and Mey, 2015).

More official approaches to the design of computing curricula have also been taken, the most well-known and influential being the series of curricula recommendations made by the ACM and IEEE (Institute of Electrical and Electronics Engineers, Inc.) (Bennedsen et al., 2008).The dismay felt by the McCracken and Lister groups is also felt strongly today by educators faced with the task of teaching programming (Bartlett and Burt, 1933). What is especially worrying, though, is that the task of teaching programming has not become easier over the last few decades (Caspersen and Kolling, 2009). A negative attitude toward programming appears to be firmly established in many students' minds and over half of the undergraduates who have elected to take a computing degree course expressed anxiety about learning to program before their university studies commenced (Huggard, 2004).

Thus, after evaluating the past studies it is evident that learning to program is hard. It was found that students find it hard to persist with learning this topic and thus there are high failure, attrition and dropout rates. Thus, it was important to discover the factors that may help students persist with their studies. It was also important to study if the chosen factors have the same effect on persistence and student performance within two different learning cultures. The study design was based on Tinto's model of persistence discussed later in detail (Tinto, 1975). The purpose of studying the factors was to improve student performance, improve student persistence and reduce the high attrition and failure rates by identifying the factors that affect learning programming, irrespective of the educational culture. The study involved a comparison between universities with two different learning cultures i.e. Australian University, Australia, and Indian University, India, as the researcher wished to compare whether the factors have similar or dissimilar effects on students' performance in different learning cultures.

Although several Universities in Australia and India were contacted, no support was received from other universities. Therefore, this study was only conducted at Australian University, Australia, and Indian University, India.

Australian University, Australia, and Indian University, India, represent the student cohort in both countries as the course structure, teaching methodology, teaching approaches, as well as the study approaches used to teach programming in both of the countries are similar. Thus, this study represents a comparison of the effects of the chosen factors in these two learning cultures.

The second part of this study involves a comparison between the use of a Course Management System and social media to improve student engagement while learning programming, which may serve as an additional source of help. Some studies have discovered that a student’s negative attitude toward programming contributes to their lack of engagement in their learning (Hockings et al., 2008), which is a key determinant of their likely poor learning outcomes (Gomes et al., 2012). A number of studies clearly indicate that social engagement enhances retention (Greenhow et al., 2009) ; (Godwin-Jones, 2008); (Winke and Goertler, 2008); (Solomon and Schrum, 2007). Another study has also proved that social engagement can benefit students (Wankel and Blessinger, 2013). Therefore, it is beneficial to perceive a student’s lack of engagement ahead of time, so that appropriate actions can be taken to re-engage her/him before she/he decides to give up. However, first year topics, especially programming topics, usually have very large enrolments, making it hard for a lecturer to keep track of each individual student’s engagement level. Even though a lecturer endeavors to do that, it is often too late to take effective action after noticing a student has disengaged through their submitted works. As Course Management Systems (CMS) have been widely adopted by universities across the globe, this study proposes using a student’s voluntary participation in a programming topic’s discussion forum provided by the CMS as an indicator of their engagement in learning. The benefit of this solution is that the CMS can easily be extended with the function that automatically generates continuous up-to-date reports on each individual student’s engagement details. An early case study concluded that there was a positive correlation between students' results and their participation in the discussion forum (Xia et al., 2013). The study was supported by the students' comments on the forum but it was only a one-semester study on a non-programming topic, partly delivered through distance learning. Also, more importantly, incentives in the form of bonus points on their final scores were offered to encourage students to participate in the topic's discussion forum.

As Facebook has been widely adopted by students across the globe, this study also proposes using a student’s voluntary participation in a Facebook group as an indicator of their engagement in learning. It seems logical to conclude that the use of social media tools should also play a role in improving retention, since they provide a digital form of social engagement (Walsh, 2012).

“Social Networking Sites (SNS) used for academic purposes have shown positive results, as students interact outside of the classroom and therefore these SNSs assist in the learning process and building community” (Hung and Yuen, 2010). Furthermore, it has been shown that “Blending the real and virtual worlds inside and outside of the classroom has been shown to increase peer to peer and academic engagement, especially for the first year students” (McCarthy, 2010).

The additional benefit of using CMS and social media as an additional source of support for learning programming is that it may also provide a collaborative learning environment for the students. Most of the students at Australian University enrolled in the programming topic come from different areas of study, such as civil engineering, mechanical engineering or computer engineering. Usually they do not know each other well enough to ask for help or collaboration in their learning activities. Previous studies indicate that collaboration helps in learning programming (Teague and Roe, 2008). Collaboration is only possible if students know each other well enough to discuss their topic-related problems. Moreover, the coupling of smart devices with social media and easy accessibility of the internet at Universities has created isolation for students, despite their being physically surrounded by other students at the university. It is not uncommon to see students sitting next to each other and not communicating with each other at all. Most of them are so engrossed in their smartphones, communicating with their friends on social media, that they almost ignore the person sitting next to them. In this scenario, this setting can be used for the advantage of students by incorporating Facebook/social media and the discussion feature of CMS provided by the University into their studies. In this manner, their social engagement can be increased, as they can communicate with their peers and discuss questions of their programming topic on social media and CMS.

Thus, this study analyses the use of social media and CMS's discussion feature to enhance the engagement of students in the programming topic, as well as to provide an additional source of help while learning programming, thereby facilitating the process of learning programming.

1.3 Research Aims and Objectives

The main aim of this study was to investigate the factors that influence the learning of introductory programming at tertiary level and how these factors affect student learning. Another aim was to investigate if these factors have similar or dissimilar effects in two Universities that have different learning cultures. A further aim was to find out if the use of CMS provided by Australian University and social media (Facebook) may serve as an additional source of peer to peer interaction and social integration in the process of learning programming, by improving student engagement and by serving as an additional source to seek help while learning programming. A comparison between the two tools was also effected.

The objectives of this study are:

1) To find new factors that may affect learning of programming

2) To compare how learning programming is approached by an Australian University and by an Indian University,

3) To study and compare the effect of the chosen factors on student persistence and performance in learning programming in terms of the grades/scores obtained in the topic in the two universities,

4) To make recommendations on teaching programming in the relevant context, and

5) To study how the use of CMS and social media can improve the process of learning programming.

1.4 Research Questions

Given the scope of this study, the main research questions are as follows.

RQ1: What are the factors and attributes that affect the learning of introductory programming by students?

RQ2: Do the factors have the same or different effects in the two Universities?

RQ3: Can use of social media in comparison with the CMS provided by Flinders University, Australia serve as an additional source of peer to peer interaction and social integration to improve the process of learning programming?

1.5 Scope of the Study

It is necessary to highlight the scope of this study as a preparatory step for defining the area where the research problem exists. This study investigates the factors in terms of learning approaches used by students and their effect on performance for Australian University, Australia, and Indian University, India. The two Universities were chosen to study whether or not the factors have similar or dissimilar effects on students' performance in learning programming. This study also investigates whether the use of CMS and social media can improve students' performance.

For the first part of the study, a questionnaire was conducted with both sets of students, based on various parameters that may affect learning programming. The results for both Universities were compared on the basis of the students' responses. This study analysed how students approached the process of learning programming in both universities and how it affected their performance. The aim was to find the factors that have a positive effect on learning programming and thus to improve the process of learning programming. Another aim was to explore if the chosen factors have similar or different effects on student performance in the two universities.

This study concentrated on the learning approaches of students while learning elements of programming such as the preliminary preparation and the type of preliminary preparation, revision and the type of revision, attendance at lectures and laboratories and the kinds of work undertaken at lectures or laboratories, the use of online resources and the effects of these factors on the students' performance in terms of grades/scores. The effects can be studied in terms of the extent to which they have learnt programming or in terms of scores. To measure the degree to which they have learnt programming, a range of parameters can be studied, such as whether students can write simple or complex programs after passing the exams, whether they have passed the exams by learning theoretical concepts or practical concepts and so forth. These elements are beyond the scope of this study due to ethical factors, as some students may feel uncomfortable about writing programs after passing the examination or may not like to be evaluated for their performance by an external person who is not involved in their assessment. Therefore, their effects have been measured/studied only in terms of their performance in terms of grades/scores obtained in the examination. Although scores obtained in examinations may not necessarily be true indicators of the degree to which they have learnt programming, in a university, assessing the performance of students is mainly measured in terms of grades/scores in examinations/assignments.

This study also investigated the use of social media in comparison with using a CMS called Flinders Learning Online (FLO), a Moodle-based Course Management System provided by Australian University. Both social media and CMS were explored to study their usage in learning programming in terms of increasing student engagement and serving as an additional source for seeking help while learning programming through a collaborative learning environment. The group feature of Facebook was adopted in this study to find out if the use of social media can provide an additional source of help in learning programming and thus help improve the learning process, as compared with the discussion feature of FLO.

Adoption of a technology such as CMS and social media could be investigated from many angles. It could focus on the process of the adoption of social media as a peer-to-peer communication tool between students with the lecturer being the observer, or as a communication platform between the lecturer and students. This study concentrated on both peer-to-peer communications between students through restricted use of a Facebook group, as the lecturer was not involved in the communication, as well as communication between students and the lecturer through the use of FLO, where the lecturer was also involved in the communication. The Facebook group study focused on peer-to-peer adoption of technology, as the students voluntarily chose to be a member of the group initiated by the researcher. The lecturer was not involved in the interactions that took place in this study, thus it was termed peer-to-peer as a result of the absence of lecturer involvement. The FLO group study focused on student-to-lecturer adoption of technology, as the students could communicate with the lecturer through the messages posted in this group and the lecturer was involved in the communication that took place as part of the study.

The effects of technology adoption could be investigated from either the technology characteristics or the user's point of view. The level of analysis could be at the micro level (individual) or at the macro level (aggregate). This study concentrated on the technology adoption from the user's point of view with *voluntary* participation in the study and neither compulsory nor forced participation in either group. The use of social media and university CMS was studied as a tool to enhance student engagement, and as an additional source of help for students while learning programming at the *individual* level (students) with its effects on their performance in terms of grades/scores in the Australian University noted and analysed.

1.6 Methodology used

The methodology used to conduct this study involved the following:

The participants: The participants in this study were the students enrolled in the first year of tertiary level study at the University, focused on those who were studying programming as part of their degree.

The first instrument used to collect data: The instrument used to collect data was a questionnaire. In Australia, the participants had to complete the questionnaire either online or via paper. In India, the students had to complete the questionnaire online.

The online questionnaire was submitted directly to the researcher and the paper-based questionnaire was supervised either by the lecturer or the demonstrators teaching the topic. After the completion of the questionnaire, it was given to the researcher for data analysis.

The second instrument used to collect the data was data logging. The data from the usage of CMS and Facebook group was collected online and analysed.

The procedure used to analyse data: The data was securely stored on the University system and was analysed statistically by the researcher to answer the research questions.

1.7 Research Significance

The answers to the research questions are necessary to assist in the process of understanding the challenges in learning programming. This study was important as it investigated the factors that affect learning and thus assists in improving the process of learning programming. The comparison of the factors between Australian University in Australia and Indian University in India was completed to investigate further whether these factors are affected by the learning culture or are unvarying demographically. It was important to find out the factors that positively affect the learning process in terms of grades/scores so that in the future, teaching of programming can be devised according to the factors that positively affect the learning of programming. The factors that positively affect students' performance across the learning culture can be focused upon by the educators in different educational cultures. Thus, the course structure and teaching methodology may be formed accordingly. Social media and CMS can be used as tools to improve student engagement and also serve as an additional resource while learning programming. By comparing the two tools for their effectiveness, the tool that helped the students most may be incorporated into the improved teaching methodology and thus facilitate the process of learning programming.

1.8 Definition of Terms

The terms *topic*, *subject* or *course* have been used throughout this study and they all carry the same meaning, as some Universities refer to the *topic* as a *subject* or *course*.

The terms *lecturer* and *topic coordinator* have been used in this study and they refer to the *person teaching the topic in that semester*. In some semesters, the topic coordinator was the lecturer teaching the topic and in some semesters the lecturer was teaching the topic and the topic coordinator was coordinating the topic. All the permissions to conduct the study were provided by topic coordinator.

1.9 Thesis Outline

This section provides an overview of the contents of the thesis.

Chapter I presents an introduction to the problem, research aims and objectives, research questions, hypotheses to be tested, scope of the study, introduction to methodology, assumptions and significance of the research.

Chapter II presents a review of the literature, Bloom's Taxonomy in a programming context, the importance of revision in learning, approaches to teaching programming, the factors analysed in the past related to programming, social networking websites and social media, the most popular social media, the use of social media in education, why social media is good to use in education, and how educators are using social media in education.

Chapter III presents the research methodology used to conduct this study. This chapter also includes how this study was conducted, why Australian and Indian students were chosen for the study, who was recruited for this research and how, the research methods used, why this methodology was chosen, from where and how the data was collected, and the application of Bigg's 3P model and statistical techniques used to analyse the data generated.

Chapter IV presents the similarities and differences between Australian University, Australia, and Indian University, India. The similarities and differences are presented in terms of the choice of language and approach, the teaching structure involved, the cultural similarities and differences, which include the cultural background of the students and their universities, the study culture in lecture theatres and laboratories with a focus on attendance, the residences of students, that is, whether they resided on their own or within the University campus with their peers studying similar topics, the examination structure and the assessment structure.

Chapter V presents a comparison of the data analysis results between Australian University, Australia, and Indian University, India. This chapter describes the introduction and methods used to analyse the data, the sample description for both Universities, the sample size, average examination score, distribution of data, descriptive statistics of the average examination score, the findings using the analysis techniques and an analysis of the students' performance of Australian University, Australia, and Indian University, India, in terms of the scores derived from examinations and testing, based on the various factors studied.

Chapter VI analyses the interrelationships between various primary factors studied in the previous chapter. Fourteen research sub-questions are further explored in this chapter. The diverse analysis methods identified to find these interrelationships between factors are enumerated. The results of the analysis are presented for the individual Universities.

Chapter VII presents the use of the Facebook group and the Course Management System (FLO) as tools to enhance student engagement and additional resources for learning programming. The chapter further explores the reason to choose social media as an additional resource for learning programming in this study, and why Facebook was chosen out of the three most popular social media resources available. Also included in this chapter is the feature of Facebook that has been used in this study, along with a definition of Facebook groups and the role of social engagement in retention. The analysis section includes an analysis of the use of the Facebook group and FLO as tools to enhance student engagement and serve as an additional resource in learning programming. The assumptions made are explained, along with the preliminary work. Further on in the chapter, the use of Facebook when Universities have their own Course Management Systems like Moodle/Blackboard is explored. A statistical analysis of the results is undertaken after identifying the variables to be analysed and selecting the appropriate analysis methods. Analyses and interpretations are completed for both the original and modified data.

Chapter VIII concludes this thesis by outlining the findings drawn from this study, a summary of the hypotheses tested, along with the results obtained through data analysis and how they are related to the research questions. Finally, the chapter also discusses the research limitations and suggests future research directions.

1.10 Summary

This chapter introduces the research problem, why this study was conducted, the problems stated in the literature related to learning programming, research aims and objectives, research questions and the hypotheses to be tested, along with the scope of this study and the significance of the research.

The upcoming chapter reviews the literature and thus the studies conducted in the past form a basis for the subsequent chapter.

CHAPTER 2 : LITERATURE REVIEW

The focus of this chapter is to review those factors previously identified through research as affecting the learning of programming. Bloom's Taxonomy is also analysed in the context of programming. Furthermore, both the approaches used to teach programming and the methodologies used to conduct the studies are reviewed, in order to identify a suitable methodology for this study. Finally, this chapter reviews the forms and usage of social media in education to evaluate if they may be exploited as mechanisms to support peer group interaction and social integration, to assist the learning of programming. Thus, this chapter identifies where a new contribution can be made in this field of study. Fig 3.1 shows the structure of the Literature Review chapter:

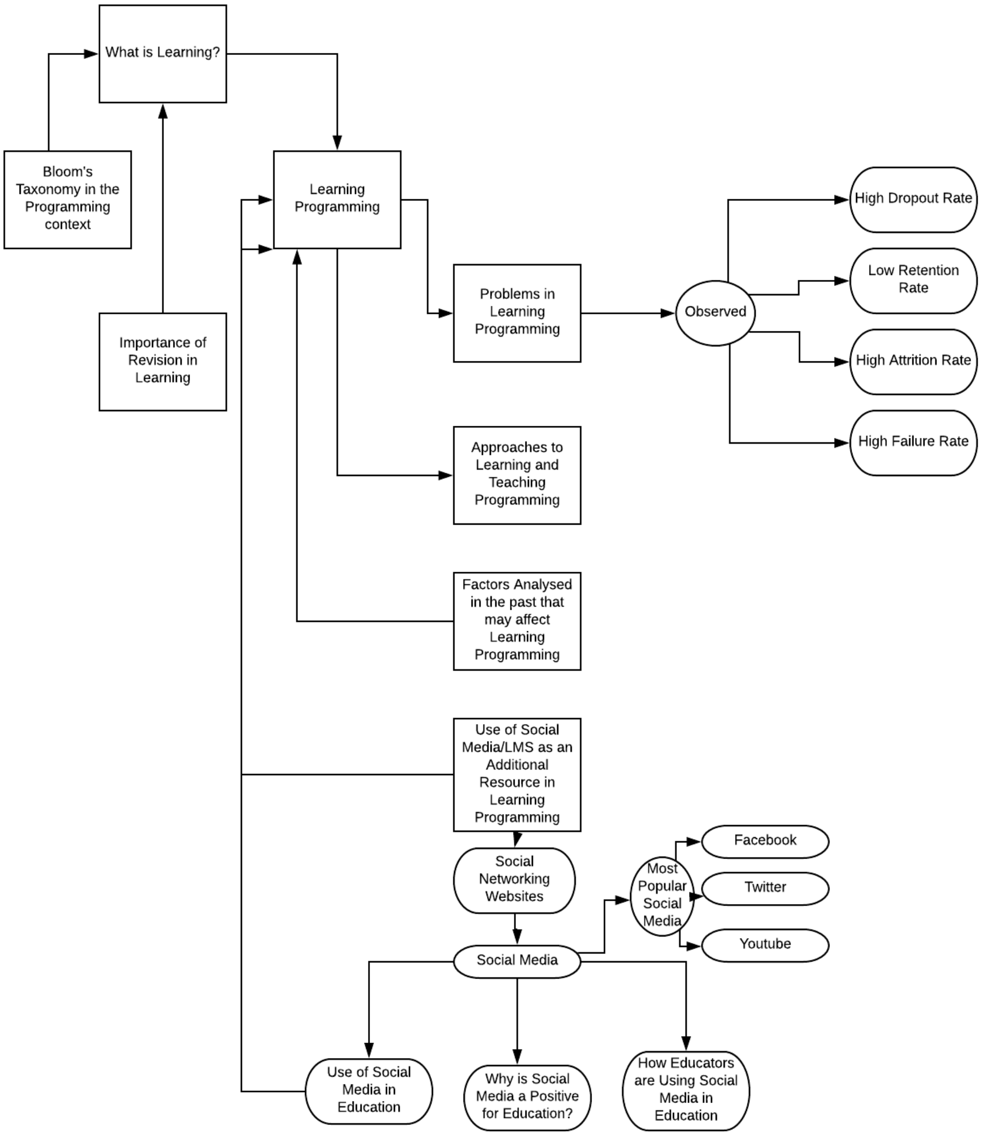


Figure 2-1: The structure of the chapter Literature Review

2.1 What is Learning?

“Learning refers to the process of increasing ones’ knowledge through the process of reading and the use of senses “(Dodero and Di Cerbo, 2012). “It [Learning] consists of a relationship between the learner and the environment, their present and past experience, a natural or innate curiosity to know and the social interaction between each of us” (Milanovic, 2015). So, the contribution of the learner in the process of learning, and in this case the student learning programming, cannot be bypassed. Identifying the factors which affect learning programming may help students understand programming more effectively.

In learning programming, like learning any other skill, the best kind of learning is effected by doing it (Hancock, 1999). This aspect of learning has been emphasized in this study. So, preliminary preparation itself, the nature of the preliminary preparation, revision and the nature of the revision, along with other factors that may affect learning programming, are explored in this study.

2.2 Problems in learning Programming observed in the literature

Learning programming has been a topic of discussion among Computer Science educators since 1970. Some students learn programming more easily than others and perform much better. They may be good performers right from the beginning when they start learning programming or become good performers over a period of time. However, study of the literature suggests that the challenges remain. The following challenges were found in the literature, which suggest that programming is difficult to learn and Universities worldwide have been facing these challenges over a significant period of time:

2.2.1 Observed: High dropout rate, high failure rate, high attrition rate and low retention rate.

Across the western world, enrolments in IT degrees have decreased dramatically in recent years (Clear et al., 2008). The declining number of students enrolling in computer science programs has become an issue (Denning, 2004). The students enrolled in computer science related courses in subsequent years of their tertiary education carefully choose options to minimise the risk of being asked to undertake any programming (Jenkins, 2002).

2.2.2 High Dropout Rates

Learning to program is generally considered hard, and programming courses often have high attrition rates (Carlson, 2010). A large number of universities have also observed that the attrition and failure rates of computing degrees are relatively high when compared with other university degree programs (de Raadt et al., 2005). Another study suggests that the relatively high dropout rates are due to poor performance in programming (Ma et al., 2007).

2.2.3 Low Retention Rates

Retention is also a major problem for IT education. In a report from the international 'Grand Challenges in Computing Education' conference, McGettrick notes that educators cite failure in introductory programming courses and/or disenchantment with programming as major factors underlying poor student retention in computing degree programs (McGettrick, 2004).

2.2.4 High Attrition Rates

A significant study has found that student attrition rates (*attrition rate:* is the number of students that move out of a course/topic in a semester/year). worldwide are high and suggests that such high numbers affect not only the individual, but also the institution, the education system, business and industry, and society as a whole (Roddan, 2002). Other studies have found that the attrition rate is around 30-50% (Newell and Simon, 1972, de Raadt et al., 2005). Not only are the departments that teach programming faced with high attrition rates, there exists a pressure to limit failure rates (Woszczynski et al., 2005a).

2.2.5 High Failure Rates

Failure rates in introductory programming are higher than for other topics. In a survey of failure rates for introductory programming courses, it was found that the average failure rate in the introductory programming course was 33% (Bennedsen and Caspersen, 2007).

Traditionally, first year introductory programming courses have a relatively high failure rate. A survey of universities and colleges worldwide was conducted to find out the failure rates. It proved that pass rates were on average around 67, giving failure rates of almost 40%% (Bennedsen and Caspersen, 2007). The causes of such high failure rates may be related to a number of factors (Butler and Morgan, 2007).

2.2.6 Unmatched Results with other topics

A study conducted by Byrne and Lyons suggests that students who are proficient in many other subjects sometimes fail to achieve success in programming and that some students who seem to perform well in early tutorials choose not to pursue the discipline(Byrne and Lyons, 2001).

All the above factors suggest that learning programming is different from learning other topics.

To understand why students face problems while learning programming, the nature of programming in a learning context needs to be explored. To achieve this, Bloom's Taxonomy is explored. The next section describes the use of Bloom's Taxonomy in a programming context both in the literature and in this study.

2.3 Bloom's Taxonomy in the programming context

In the 1950s, educational psychologist Benjamin Bloom developed a hierarchical classification of behavior that is important for learning. It is depicted using a pyramid shape. (Shih, 2011). There are six levels of abstraction and the level of complexity increases at each level. The bottom of the pyramid indicates simple cognitive behavior such as recall and fact recognition, leading on to more complex behavior, involving increasing mental abstraction (Shih, 2011). It is suggested that one cannot effectively — nor should one try to — address the higher levels until those below have been covered (Clarke and Clarke, 2009).

Table 2-1, below, describes Bloom's Taxonomy, with section 1 describing Skills observed and displayed, section 2 describing useful verbs along with the third section which describes the application of Bloom's Taxonomy in the context of learning programming.

The statements at each level describe a task at each level of the Taxonomy.

Table 2-1: Bloom's Taxonomy and its application in the programming context

	Skills Observed and Displayed	Useful Verbs	Application in context of learning Programming
Knowledge (Finding Out)	Observation & recall of information, facts, knowledge of dates, places, events, major ideas, concepts, terms, principles, mastery of subject matter,	Name, list, define, tell, describe, relate, select, identify, label, show, quote, name, find, write, locate, state, who, when, where, outline, match	State the types variables in Java or C. Write the syntax of FOR loop.
Comprehension (Understanding)	Understanding information, principles, grasp meaning, translate, knowledge into new contexts, interpret charts, facts, compare, contrast, order, group, infer,	Translate, explain, give examples, predict, rewrite, describe, outline, convert, summarise, interpret, discuss, predict, distinguish, restate,	Instantiate a character variable with 10 characters and assign value to them as 'a' 'b' 'c' 'd' 'e' 'f'' 'g' 'h' 'i' 'j'
Application (Making Use of Knowledge)	Use/apply information Use methods, concepts & theories in new situations, solve problems using required skills or knowledge, construct charts and graphs	Construct, complete, classify, solve, show, use, illustrate, apply, calculate, examine, demonstrate, modify, relate, change, predict, produce, compute,	Write a method/function that converts temperature from Celsius to Fahrenheit and
Analysis (Taking apart the known)	Seeing patterns, organisation & identification of components & parts, recognition of hidden meanings, distinguish between fact & inferences,	Compare, collect, select, explain, infer, analyse, distinguish, separate, investigate, contrast, connect, arrange, categorise, advertise,	Differentiate between 'while' and 'do while' loops.
Synthesis (putting things together differently)	Re-present old ideas to create new ones, relate/integrate knowledge from several areas, predict, create, draw conclusions, propose, produce original	Design, imagine, improve, create, plan, invent, devise, design,formulate, reconstruct, generate, modify, review, combine, integrate, compose,	Create a project that accepts the scores of students and generates grades on the basis of scores.

Evaluation (Judging outcomes)	Discriminate between ideas, assess value of theories, presentations, make choices based on reasoned argument, verify/appraise value of evidence/work, recognise subjectivity	Judge, interpret, grade, conclude, assess, rank, justify, debate, argue, assess, determine, rate, verify, recommend, select, discriminate, support, prioritise, appraise,	Evaluate the outcome of the project created by the student to find out if the output generated is the required output.

Bloom's Taxonomy is a cognitive skills taxonomy which has been applied in many fields of education, including computer science (Eldon, 2008). Lister and Leaney in their research on assessment of students studying programming, classify programming tasks on different levels of Bloom's Taxonomy. (Lister and Leaney, 2003)

The task of programming or software development is at the synthesis level: according to Bloom's Taxonomy, it involves relating/integrating knowledge from several areas. Also, in programming assignments, a programmer may be required to have not only knowledge about programming constructs and structures, but also some knowledge of the area of application. For example, to design a Banking System, a programmer needs to have some idea about the banking principles that need to be applied to design the software system.

This section explains the nature of programming tasks at different levels of Bloom's Taxonomy. The next section explores the importance of revision in learning to establish if revision plays a role in better learning of programming.

2.4 Importance of revision in learning

The research shows that revision plays a vital role in learning, and thus the relevance of revision cannot be ignored when learning programming. Learning programming requires cognitive skills and so the memory has to process much information simultaneously. For example, if a student is given a problem, and he/she has an idea that the problem can be solved in a particular manner, he/she also needs to know how to transfer his/her idea into a piece of code which will be executed to get the desired outcome. This means that the student needs to have a good understanding of programming concepts. This can be achieved if the student devotes time to learning the concepts presented in the lectures. For example, if loops are taught in the lecture, then the student should try and write new, similar programs to have a better understanding of loops. Some students grasp concepts very quickly, while others take longer, but if time is spent on revising and learning deeply, then an average student may be able to perform at a well above average student level, so revision may help in learning programming.

Figure 2-2 below represents the forgetting curve described by Ebbinghaus. The x-axis represents the length of time when information is remembered, measured by the number of days, and the y- axis represents memory. The forgetting curve is exponential, which means that memory loss is the greatest in the first few days, later (as evident from Fig. 2-1) forgetting still occurs, but the rate is slower than at the beginning. The forgetting curve clearly shows that in the first period after learning or reviewing material, we forget the most information.

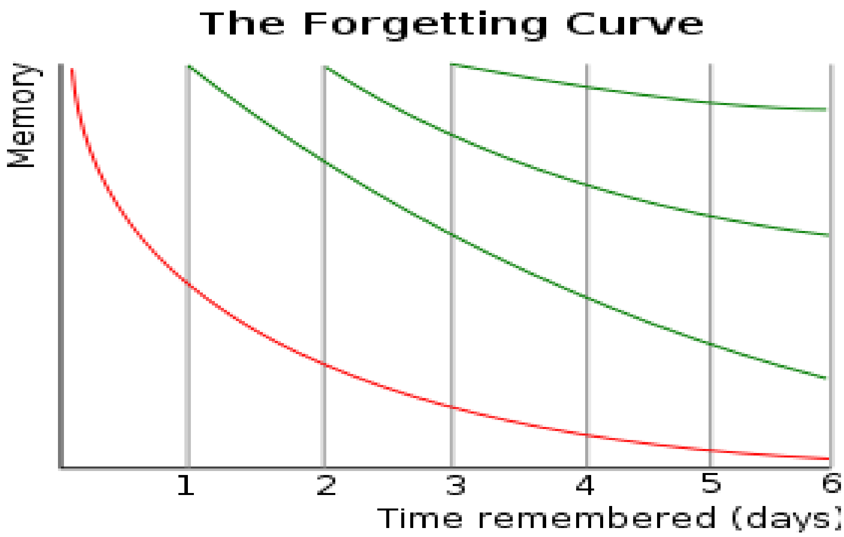


Figure 2-2: A typical representation of the forgetting curve by Hermann Ebbinghaus *(Psychestudy) (Ebbinghaus, 1985)*

Figure 2-3 shows a change in the learning curve after the information is revised numerous times. The x-axis represents the number of repetitions and the y-axis represents the percentage of data remembered. Every time information is revised, the level of knowledge reverts back to 100%. Furthermore, every time the information is reviewed, the memory is able to retain more information**.** Therefore, the learning curve and the corresponding retention rate for that piece of information become flatter (D'Monte, 2009).

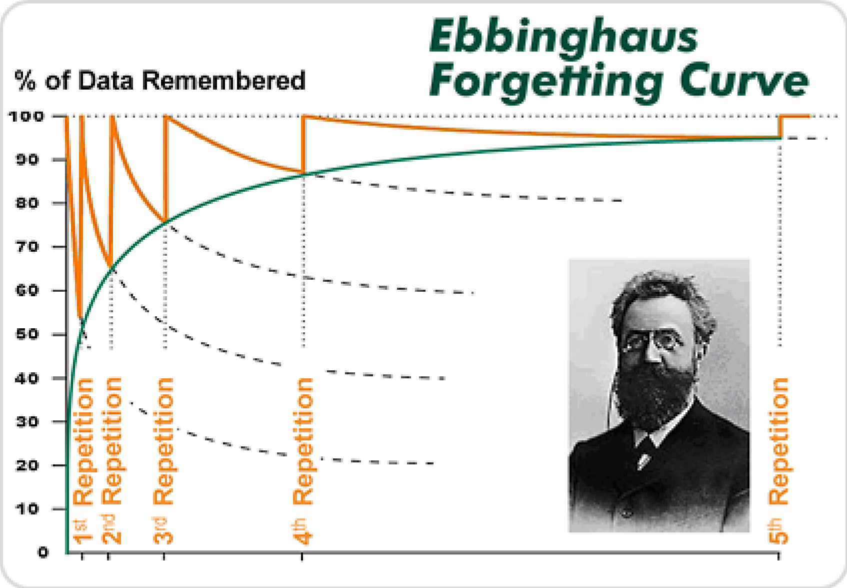


Figure 2-3: The forgetting curve after Repetition by Hermann Ebbinghaus *(Alexa.com, 2013)*

The concept of revision discussed in this study cannot be termed *rote learning* i.e. going through learning material repeatedly, without variation. Revision in the programming context refers to the practice of writing programs, beginning with simple programs already learnt and then moving on to complex structures of programs. Programming cannot be learnt by rote learning, as cognitive abilities are required to learn programming.

A study conducted on learning programming in a time span of four years concluded that turning a novice into an expert programmer is impractical in a four year program, however competence can be achieved (Robins et al., 2003). He stressed the importance of practice and suggested revision of old concepts in addition to learning new ones as a way of developing skills effectively. This section of the thesis suggests that revision promotes learning. Thus, revision and preliminary preparation need to be explored further to investigate if these factors affect students' learning of programming. These factors have not been explored in the past.

The next section explores the factors analysed in the past that may affect learning programming.

2.5 Factors analysed in the past

Finding the factors that affect students' learning of programming may help decrease the failure and attrition rates and improve such learning. If the factors that affect learning of programming can be identified, the students can focus on these factors to improve their learning and the educators can devise the teaching methodology/ies accordingly. Extant studies were explored before conducting this study, to locate factors worthy of further investigation or reinvestigation.

de Raadt studied the efficacy of a variety of approaches to learning amongst computer programming students and concluded that in computing, as for other disciplines, learning approaches were a powerful determinant of success (de Raadt et al., 2005).

Simon also conducted a multi-national, multi-institutional study that investigated introductory programming courses and concluded that there exists a positive correlation of scores with a deep learning approach and a negative correlation of scores with a surface learning approach (Simon et al., 2006).

Hagan and Markham analysed the effect of prior programming experience and the number of programming languages learnt, and concluded that a positive correlation existed between prior programming experience and student performance (Hagan and Markham, 2000).

Wilson and Shrock analysed twelve possible predictive factors, including mathematics background, attribution for success/failure (luck, effort, difficulty of task, and ability), domain specific self-efficacy, encouragement, comfort level in the course, work style preference, previous programming experience, previous non-programming computer experience, gender, spatial reasoning and mathematical ability, and concluded that comfort level and mathematics background seem to have a positive impact on success, whereas attribution to luck had a negative impact(Wilson and Shrock, 2001).

Mayer analysed measures of general intellectual ability and thinking skills and concluded that English and BASIC (Beginners All Purpose Symbolic Instruction Code) pre-training had a positive impact on student learning in terms of the speed and accuracy of learning (Mayer et al., 1986). Roddan, in a study concluded that a deep learning approach was found to have a positive correlation with scores, and a surface learning approach was found to have a negative correlation with scores(Roddan, 2002). Fincher et al. conducted a similar study and concluded that deep engagement of the students with the material tended to have a positive impact on performance(Fincher et al., 2006). Another study concluded that a student's perception of their understanding of the module had the strongest correlation with their programming performance (Bergin and Reilly, 2005). A further study indicated an association between programming ability and aptitude in mathematics and science subjects and also found that the performance of female students was on a par with that of male students; an outcome which contradicted some of the results achieved in previous studies (Byrne and Lyons, 2001). Yet another study showed that the influence of learning styles had a positive impact and found that sequential learners outperform random learners in computer-related courses (Lau and Yuen, 2011). Finally, Winslow gave an excellent overview of psychological studies into computer programming since the 1970s and concluded that, by the end of a four-year degree program, students should be proficient enough computer programmers; capable of consciously choosing an organized plan to achieve a desired goal (Winslow, 1996).

In reviewing the literature relating to predicting success in learning a first programming language, no clear result emerged as to the best approach to take to teaching and learning. It was found that even after 40 years of study, prediction of exact success factors continues to vary according to the circumstances and individual undertaking the learning (Woszczynski et al., 2005a). Despite extensive research on teaching methods and student responses, definitive predictors of success in learning to program could not be found (Bornat and Dehnadi, 2008); yet if some can program, and some cannot, there must be reasons why this is the case (Robins, 2010). Eminent researchers have suggested that even if the predictors of success cannot be found in computer programming, the research is still worthwhile, as it discourages the attribution of success in programming to innate factors, and hence encourages a more productive approach to learning" (Fincher et al., 2006).

A summary of these factors in the form of Table 2-2 as Factor (Publications) is presented below:

2.6 Summary of the Factors Analysed in the past:

Table 2-2: Summary of the factors analysed in the past

Factor:	**Publications**
Previous computing experience	(Wilson, 2002);(Bergin and Reilly, 2005)
Previous programming experience	n, 2002);(de Raadt et al., 2005)(Hagan and Markham)
Previous non-programming computer experience	n, 2002)
Attribution	n, 2002)
Self-efficacy	n, 2002)
Comfort	(Wilson, 2002)
Encouragement from others	(Wilson, 2002)
Work style preference	(Wilson, 2002)
Math background	(Wilson, 2002)
Midterm grade	(Wilson, 2002)
Learning approaches like deep learning approach and surface learning approach	(Fincher et al., 2006);(Robins et al., 2003)
Learning style	(Byrne and Lyons, 2001)
Standard paper-folding test	(de Raadt et al., 2005, Fincher et al., 2005, Fincher et al., 2006)
A cognitive task focusing on spatial visualisation and reasoning	(Fincher et al., 2006)
Map sketching	(Fincher et al., 2006, Fincher et al., 2005);(Tolhurst et al., 2006)
A behavioural task used to assess the ability to design and sketch a simple map and to articulate decisions based on that map	(Fincher et al., 2006)
Searching a phone book	(Fincher et al., 2006)
A behavioural task used to assess the ability to articulate a search strategy	(Fincher et al., 2006)

A standard study process questionnaire	(Fincher et al., 2006)
An attitudinal task focusing on approaches to learning and studying	(Fincher et al., 2006)
Bigg's instrument	(de Raadt et al., 2005)
Shortened Intellectual Development (ID) predictor	(Barker and Unger, 1983)
Cognitive	(Sheard et al., 2009, Bergin and Reilly, 2005)
Behavioural	(Sheard et al., 2009) attitudinal factors(Sheard et al., 2009, Robins et al., 2003)
Reading and Tracing Skills in Novice Programmers	(Lister et al., 2004)
The ability to articulate strategy	(Cutts et al., 2006)
Measures of general intellectual ability and thinking skills	(Mayer et al., 1986)
Self-predicted success	(Robins et al., 2003)
Keenness and general academic motivation	(Robins et al., 2003)
Previous academic experience	(Bergin and Reilly, 2005, Byrne and Lyons, 2001)
Personal information	(Bergin and Reilly, 2005)
Experience on the module	(Bergin and Reilly, 2005)
Gender	(Byrne and Lyons, 2001, Lau and Yuen, 2011)
Mental models	(Lau and Yuen, 2011)
Prior composite academic ability	(Lau and Yuen, 2011)
Medium of instruction	(Lau and Yuen, 2011)

After reviewing the literature, it was found that a number of factors in various categories have been studied repeatedly by researchers since the 1970s, suggesting that the success of students in learning programming depends upon several factors. It is therefore reasonable to search for explanations; that is, to search for pre-study elements, as well as in-study indicators of success for introductory programming. It can be argued that finding predictors may help decrease failure rates (de Raadt et al., 2005, McGettrick et al., 2005, Boyle et al., 2002, Fincher et al., 2006, Tolhurst et al., 2006). Thus, this section summarises the factors analysed in the area of learning programming. This forms the basis of the factors to be explored in this study. Later sections will explain the importance of these factors.

The following section explores the role and nature of social media as a resource to enhance student engagement. Diverse aspects of social media and their impact on learning are discussed below in terms of their impact on educational culture and learning

2.7 Social media explored

Various methods of exploring peer group interactions were investigated. Social media was found to be a promising channel and thus was explored in detail before using it as a medium of peer group interaction and social integration.

2.7.1 Social Networking Websites

Socialising via the Internet has become an increasingly important part of young adult life (Gemmill and Peterson, 2006). Social networking sites are the latest online communication tools that allow users to create a public or private profile to interact with those in their networks (Boyd and Ellison, 2007). Social networking websites are virtual communities which allow people to connect and interact with each other on a particular subject or to just ‘‘hang out” together online (Murray and Waller, 2007).

2.7.2 Social Media

Social media employ mobile and web-based technologies to create highly interactive platforms via which individuals and communities share, co-create, discuss, and modify user-generated content (Kietzmann et al., 2011). The concepts behind *social networking* are not new – ever since there have been humans, we have been looking for ways to connect and network with each other – but they have taken on an entirely new meaning (and momentum) in the digital age (Milanovic, 2015). Where we used to have handshakes, word-of-mouth referrals, and stamped letters, it is often argued that today's relationships are often begun and developed on LinkedIn, Google+, and Facebook (Milanovic, 2015).

A rich, diverse ecology of social media sites exists, which varies in terms of scope and functionality (Kietzmann et al., 2011). There are eight different types of social media as represented in Figure 2-4 below (Sorokina, 2015).

Figure 2-4: Types of Social Media
(Sorokina, 2015)

2.7.3 The Social Media Available

Table 2-3 shows a range of social media that applications dominate the market. The top 10 forms of social media as at April, 2016 were:

Table 2-3: Top 10 social media (eBizMba, 2016)

Type of Social Media	Number of monthly visitors

Facebook	1,100,000,000
Twitter	310,000,000
LinkedIn	255,000,000
Pinterest	250,000,000
Google+	120,000,000
Tumblr	110,000,000
Instagram	100,000,000
VK	80,000,000
Flickr	65,000,000
Vine	42,000,000

2.7.4 Use of Social Media in Education

According to a survey conducted by the Babson Survey Research Group, in collaboration with New Marketing Labs and Pearson, a global leader in education, technology and services, "More than four out of every five professors use social media", based on a (U.S.) national survey of nearly 1,000 faculty members. The survey also suggested that more than 30 percent use social networks to communicate with students; social media use is higher among the faculty in the Humanities and Social Sciences than those in Mathematics, Science, Business and Economics, (Susan, 2010). If social media is adopted for teaching programming, it may have a positive impact on the learning outcome of students. If social media is used in a positive manner, it may become the next best option after direct communication.

A 2011 study conducted on Facebook usage in education, participants described their Facebook group as "a pressure-free environment for English learning because it was a virtual community composed of closed group, which opens for limited members and makes them feel less stressful" (Wu and Hsu, 2011). Furthermore, a study on integrating Facebook with peer assessment with blended learning indicates that Facebook had a constructive impact in an ESL (English as a Second Language) writing course (Shih, 2011). This suggests that Facebook has helped students in their learning. A study of this, conducted at Australian University across two years, yielded varied outcomes (Sharma et al., 2016). Largely, the study suggests that if students accept Facebook as an additional educational tool, then it may prove to be beneficial. The challenge throughout this process is that students are often reluctant to use Facebook as an additional resource to the ones already available to them through the University (via the University Course Management System). The results are replicated by another, parallel study, which reports that using the Facebook tools increases students' motivation (Ross et al., 2009). Future studies should concentrate on integrating Facebook into education and teaching, as it is important in students' everyday working lives (Bicen and Cavus, 2011). A third study by Bugeja suggests that social networking offers the opportunity to re-engage individuals with learning and education, promoting a 'critical thinking in learners' about their learning, which is one of 'the traditional objectives of education'(Bugeja, 2006). Finally, a fourth study suggests that Facebook use in and of itself is not detrimental to academic outcomes, and can indeed be used in ways that are advantageous to students (Junco et al., 2011).

Another study conducted in 2017 to explore the education related use of social media for business students in India suggests that the approaches related to social media for learning positively enhanced the experiences of undergraduate and post-graduate business students(Bharucha, 2018).

A study conducted in 2018 on 723 students in Malaysia that analysed the use of social media for active collaborative learning and engagement concluded that student satisfaction of social media positively affected the learning performance of students(Al-Rahmi et al., 2018).

A study conducted in 2018 in Australia, to explore the educational use of social media and social connection for international students comparing Cooperative vs collaborative group work concluded that the students choosing to engage in collaborative interactions instead of cooperative interactions through social media were more likely to perceive a connection to their classmates

Another study conducted in 2017 that reviewed the effects and attitudes of social media in education concluded that the use of social media may have the potential to improve learning and positive effects may be yielded if the social media tools are combined with the real interaction in class(Zu et al., 2017).

2.8 Why Social Media a Positive for Education

Certain features of social media such as instant messaging makes it a probable choice to use in education as students may ask questions and get an instant reply.

JCR Licklider, an American psychologist and computer scientist remains one of the most important figures in computer science and general computing history. He was the first to foresee the future of computer networking, claiming “In a few years, men will be able to communicate more effectively through a machine than face to face. … we believe that we are entering a technological age in which we will be able to interact … as active participants in an ongoing process, bringing something to it through our interaction with it”(Licklider, 1965). Similarly, Grover and Stewart suggest that social media provide access to new sources of knowledge and new opportunities for learning both within the traditional model of learning and in new and evolving ways(Grover and Stewart, 2010). Another study emphasized the use of technology to address different student learning styles (Rodriguez, 2011). Thus, the potential of Social Media to enhance education is immense. Students do not need to be externally motivated to use Social Media, as they already spend significant parts of their quotidian life online. Educators need to supplement their teaching methodologies to communicate with students using this platform.

2.9 How Educators are using Social Media in Education

The rate of adoption of Social Media in the professional lives of teaching faculty was found to be over 90% (Moran et al., 2011). A systematic review of the published literature on Social Media use in Medical Education concludes that interventions using Social Media tools are associated with improved knowledge, attitudes and skills (Cheston et al., 2013). Junco et al., in their earlier study based on Twitter, conclude that their study provides the first piece of controlled experimental evidence that using Social Media in educationally relevant ways can increase student engagement and improve grades(Junco et al., 2011). This growing body of research, therefore suggests that Social Media should be explored for its ability to assist students in learning programming, as most of the students and educators are already engaged with this type of communication.

2.10 Summary

This chapter reviews the nature and importance of learning, the importance of revision and various approaches to teaching programming. There are two broad approaches to teaching programming: top-down and bottom-up. These approaches were explored to determine their effect on learning programming. Also, studies related to various factors that affect student learning of programming that had previously been evaluated were analysed. Given the importance of the twin skills of revision and learning, both need to be investigated further and such a study was undertaken as part of this research. It has been demonstrated that individual learning styles, approaches towards learning programming and key attributes of students play an active, critical role in students' learning. Thus, a gap was identified which formed the basis of this research. The gap focused on a study of peer group interactions through social media, an investigation of the most popular types of social media and the use of Social Media in education. It was then found that one of the three most popular types of Social Media can be used as a mechanism for both peer group interaction and social integration, thereby acting as a tool to enhance student engagement and serve as an additional resource in learning programming. The next chapter will outline the methodology used to achieve the aims and objectives of the research and answer the research questions directly. Furthermore, it will evaluate the research methods used, the ways in which the methodology is appropriate for this study, from where and how the data was collected, the statistical techniques used and the application of Bigg's 3P model for research purposes.

CHAPTER 3 RESEARCH MODEL/FRAMEWORK

The purpose of this study was to find ways to help student learn programming, it was found from the literature that the attrition rate is high and retention rate is low in programming topic. Thus it was important to search for models that help in the persistence at University level and then to adopt it to programming topic to find if it holds true for programming topic as well.

3.1 A few models were extracted from the literature

3.1.1 The Undergraduate Dropout Process Model (Spady, 1970) ,(Spady, 1971)

A theoretical model based on (Durkheim, 1951) concept of social integration. Two questionnaires from two diverse areas were used. The first questionnaire included items directed toward four general areas: the student's high school and family background, his expectations and motivations concerning life and performance at the University of Chicago, self-assessments of his intellectual capacities and personal relationships, and his social and cultural life.

The second questionnaire included the student's perceptions of environmental influences, friendship affiliations, reactions and behaviour toward both students and parents during the academic year, personal values, interests and attitudes, descriptions of the environment and other students, evaluations of courses, sense of intellectual development and social integration, expectations and satisfaction in diverse areas of life, and time spent in a host of activities.

3.1.2 A 10-variable causal model of the attrition process (Bean, 1980, Bean, 1982)

Bean studied the effect of 10 variables on student attrition rate at University. The 10 variables that are the predictors of the student attrition starting from most important to least important

1. Intent to leave

2. Grades

3. Opportunity to transfer

4. Practical value

5. Certainty of choice

6. Loyalty

7. Family approval

8. Courses

9. Student goals

10. Major and occupational certainty

3.1.3 The Student-Faculty Informal Contact Model (Pascarella and Terenzini, 1980)

This research focussed on four areas:

Student Background and characteristics

Institutional factors

Informal contact with faculty

Other college experiences

Educational outcomes

3.1.4 The Student Retention Integrated Model (Cabrera et al., 1993)

The variables used in this study were

Environmental variables, Endogenous variables, social integration, Institutional Commitment, Goal Commitment.

3.1.5 Tinto's model of persistence (Tinto, 1975)

Tinto's model was built on the basis of (Spady, 1970, Spady, 1971) model and based on the work of Emile Durkheim and Arnold Van Gennep

He stated that colleges system comprises of: academic and social systems. An integration is required into both systems by the student to persist at University.

The model also states that a student enters University with certain Goals as well as commitments which are shaped by student's attributes such as background of the family, his skills and abilities and the prior schooling. These goals and commitments affect student's level of goals and commitments which affect his persistence at University.

He also amended his model adding that the strength of a student's level of social and academic integration affects his or her persistence (Pascarella and Terenzini, 1980).

Tinto's model was found to be the most comprehensive model after the detailed study. Most of the above models except Spady's model were based on Tinto's models thus Tinto's model was chosen to be explored further in the area of learning programming. The other factors that led to the choice to study Tinto's model are:

1. It was first proposed in 1975 and was amended with Final version of the model in 1993 which suggests that it evolved with time and Tinto amended it with additional factors.
2. It is the most cited model for university persistence and has 14075 references on google scholar.
3. It has been examined and tested and thus validated by many studies ((Pascarella and Terenzini, 1980); (Berger and Braxton, 1998); (Elkins et al., 2000);(Pascarella and Chapman, 1983); (Halpin, 1990); (Murguia, 1991); (Sweet, 1986); (Brunsden et al., 2000); (Nora et al., 1990); (Getzlaf et al., 1984); (Baird, 2000); (Moström, 2011)
4. Most highly respected model (Kember et al., 1995)

3.1.6 The relationship of the components of the research model

The components of the research model suggested in Section 4.8 are primarily based on Tinto's conceptual model .There doesn't exist a direct relationship between the components as they were chosen based on the individual components of Tinto's model.

3.2 Application of Bigg's 3P Model

Bigg's 3P model was used to evaluate the success of students based on their attributes and how they approached learning programming (Biggs, 1987). Bigg's 3P model was used after its successful use in the literature to evaluate students' learning of programming. A study suggested that the models for explaining student success in programming should be augmented to include data from the Biggs questionnaire, or a similar instrument (de Raadt et al., 2005).The two models suggested by Biggs are the general model to evaluate student learning and a questionnaire to evaluate student learning. In this research, the general model was used rather than the questionnaire used to evaluate teaching. This study evaluated student learning in terms of their study habits in and outside the classroom. Greater depth of learning, knowledge and skills transfer is possible when students are suitably pre-prepared/pre-skilled for the subsequent learning experiences and variety of teaching/learning interactions to be encountered (Hamilton and Tee, 2010). This study also concentrated on the attributes of students in terms of prior knowledge of a programming language, knowledge of algorithms and flowcharts, sources sought to offer help for the topic, attendance in lectures and laboratories and learning approaches in terms of preliminary preparation, revision and their impact on student learning.

Biggs suggests that the resulting learning outcomes are complex and work in interaction with each other. He suggests the general direction of effects may be represented by heavy arrows, as shown in Figure 3-2, and that both student factors and the teaching context jointly drive the system towards a common set of learning outcomes.

3.2.1 Bigg's 3P: Presage-Process-Product.

Bigg's 3P model, as shown in Figure 3-2, was used in this study, as the 3P model has a feedback mechanism to inform the lecturer and students of changes that might have to be made to achieve desirable learning outcomes in any given educative process.

Presage: The Presage stage refers to individual (and institutional) states of *being* that foreshadow the educative process (Wikispaces, 1996).

The Student Presage state describes the learning-related characteristics of the student in terms of prior knowledge. In this study, this refers to knowledge of a programming language and skills in designing flowcharts and algorithms before learning programming at University, abilities, preferred approaches to learning which in this study refer to the learning approaches used by the students to learn programming, values, expectations, and competence in the language of instruction (Wikispaces, 1996).

The Teacher Presage state describes the lecturer's competence as a communicator and an educator, the use of curricula (including teaching methods) that underpin teaching and learning, the classroom climate, assessment practices, and the medium of instruction (Wikispaces, 1996).

In this study the course was taught by different lecturers in different semesters. The assessment method also varied across the semesters. This may help to determine individual effects on student learning i.e. the product stage in future studies.

The next is the Process state: that is, how the presage stage, in this case the student characteristics, affect the students. The Process stage refers to the manner in which students actually handle the task (Biggs, 1996). The process stage in this study refers to the way students handled the task of learning programming, like preliminary preparation before the lecture or laboratory, kinds of preliminary preparation made, the revision done by the students during the semester, kinds of revision undertaken, activities in the lecture theatre and some other approaches listed in detail in the study.

The Process stage leads to the Product stage of students' learning and describes low- and high-level cognitive outcomes. These range from quantitative recall in the case of low-level outcomes, which in this study were demonstrated by performance in the laboratory and attendance in lectures and laboratory, and activities performed in the lecture theatre, which can be a measure of effective involvement in learning the topic (Biggs, 1996).

High level outcomes were indicated by correct and relevant answers, abstract thinking, and elegant conceptualization of problems, which in this study were measured by the scores obtained by the students in the examination, scores obtained in the quiz, scores obtained in the project which demonstrate abstract thinking, and elegant conceptualization of problems (Biggs, 1996).

Biggs also indicates that each specific institution has an impact on the teaching and learning process. Thus, with many complex variables intertwining, any change in one area likely shows as an affect in another (Hamilton and Tee, 2009).This study involved student participants from two different universities in Australia and India. Thus, the data from different universities helped to analyse if the performance of students varied within Universities.

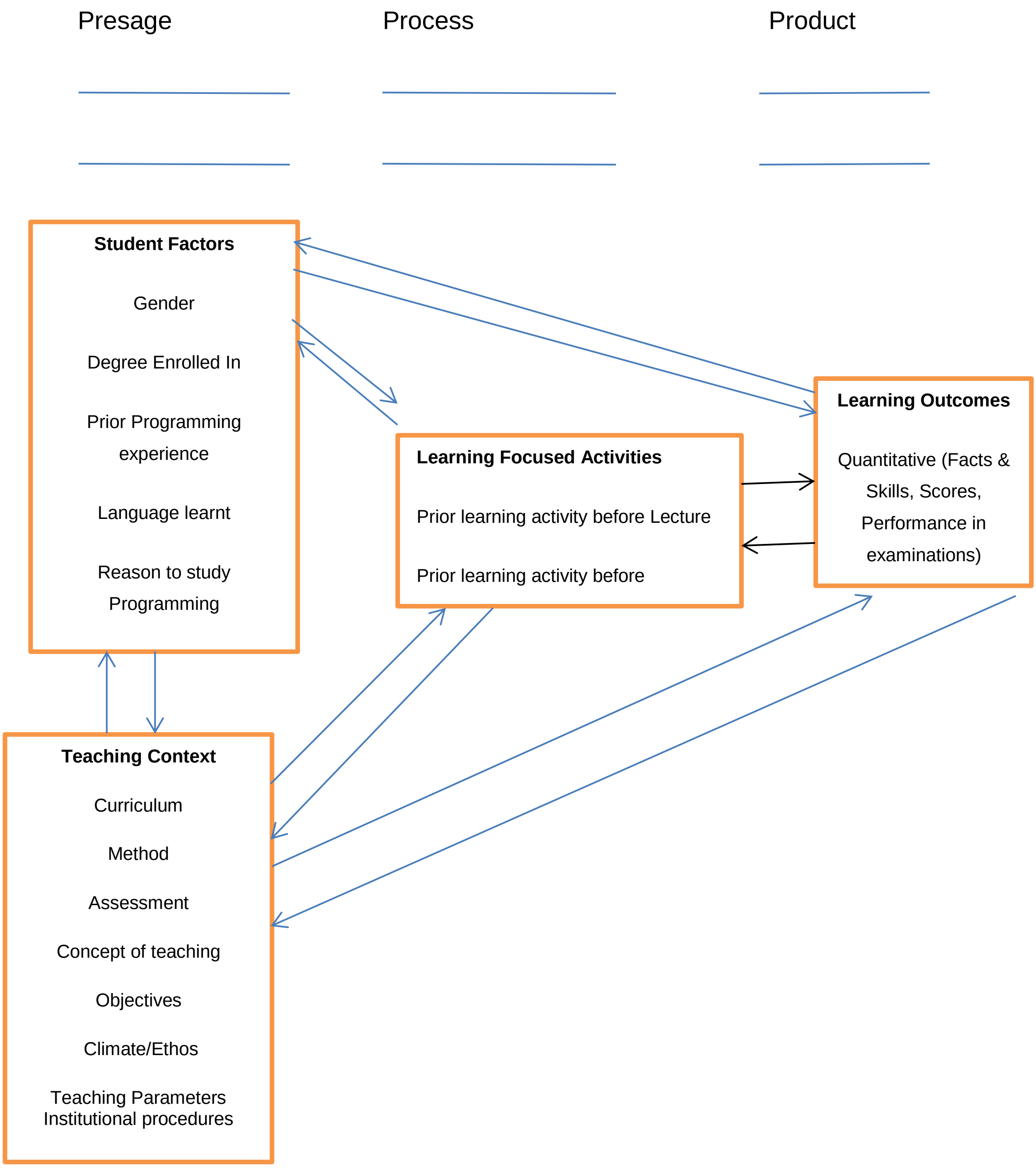


Figure 3-1: Application of Bigg's 3P model

(Bigg's 3P Model (Biggs, 1996) as referenced in (Wikispaces, 1996)

3.3 Why have learning approaches and factors that may affect learning of programming been analysed in this study?

Researchers have used different approaches to facilitate the process of learning programming. Many studies in the past support the fact that diverse student attributes contribute to success in learning programming. The results from an earlier, multi-national, multi-institutional study indicate that a deep approach to learning is positively correlated with scores for the topic, while a surface approach is negatively correlated (Fincher et al., 2006). Another study also suggests that the best indicators of success appear to be self-predicted success, attitude, keenness and general academic motivation (Robins et al., 2003). The results from a further, large-scale study undertaken across eleven institutions in three countries show that, like other disciplines in computing, learning approaches play a strong role in ultimate success (de Raadt et al., 2005). So, in this study, a comparison between two Universities, one in Australia and one in India, was undertaken on the basis of the factors that affect learning programming. The evaluation involved a comparison of the named factors and their effects on learning programming for the two chosen Universities. The main factors investigated in this study were learning approaches and student attributes. The aim was to improve persistence in the topic by finding a correlation between the learning approaches in terms of student attributes/habits and student performance, which was assessed by their scores in the examination. The use of social media, such as Facebook, to enhance student engagement and improve the learning of programming in comparison with the use of FLO alone was also a key element of this study. This study was conducted with tertiary level students learning programming as a part of their degree.

Programming is taught using different teaching styles and approaches. To improve the learning and teaching of programming, the best approaches may also be extracted as a result of this comparative study.

The factor analysis undertaken in this study is based on Tinto's conceptual model (Tinto, 1975), as represented in Figure 3-3 and Roddan's conceptual schema for university persistence (Roddan, 2002). Tinto suggests that grade performance is based on diverse factors, which include family background, individual attributes, pre- college schooling, goal and institutional commitment and, interestingly, peer group interactions and social integration. All these factors were taken into consideration when designing the factors to be studied. Peer group interaction was explored in detail in this research for Australian University by evaluating and comparing the use of CMS and social media.

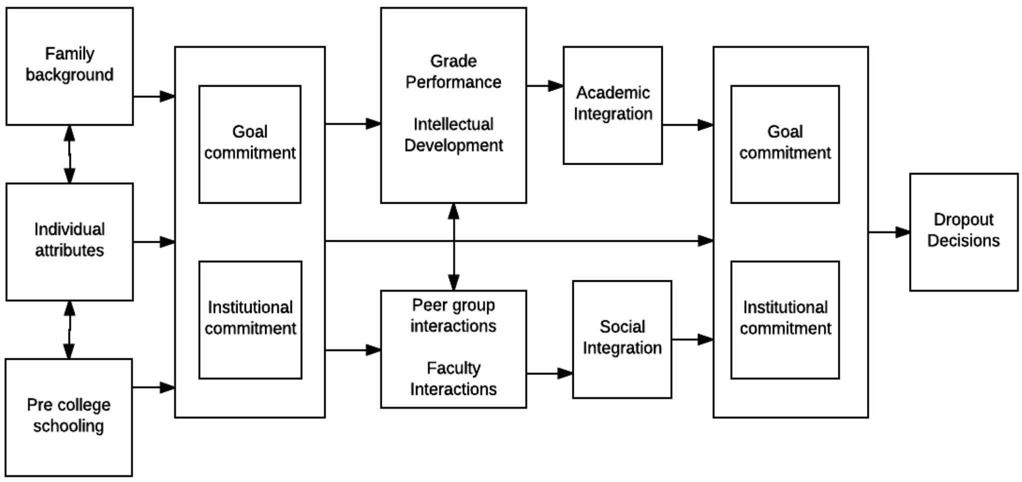


Figure 3-2: Tinto's conceptual schematic
(Tinto, 1975)

A model of the factors analysed in this study: The model in Figure 3-2 was created by combining the factors from 7 areas that may affect student learning of programming. Some of the factors have already been studied. These factors were included in this study to investigate how they affect learning in a new context. The factors that have already been studied elsewhere are gender, prior programming experience family background and study choices. The additional factors that are included in this study are preliminary preparation and revision, interest in programming, and the source(s) from which help is sought. The source(s) of help are further explored on the basis of their sub-factors.

To implement the central part of Tinto's model of learning, which describes peer group interaction and social integration as contributing factors for success, the use of social media was compared with the use of CMS to help improve student engagement and serve as an additional source of peer to peer interaction and social integration in the process of learning programming at Australian University. This comparative study (of social media with CMS) could not be conducted at Indian University due to insufficient institutional support, as students could not be asked to use social media on university premises.

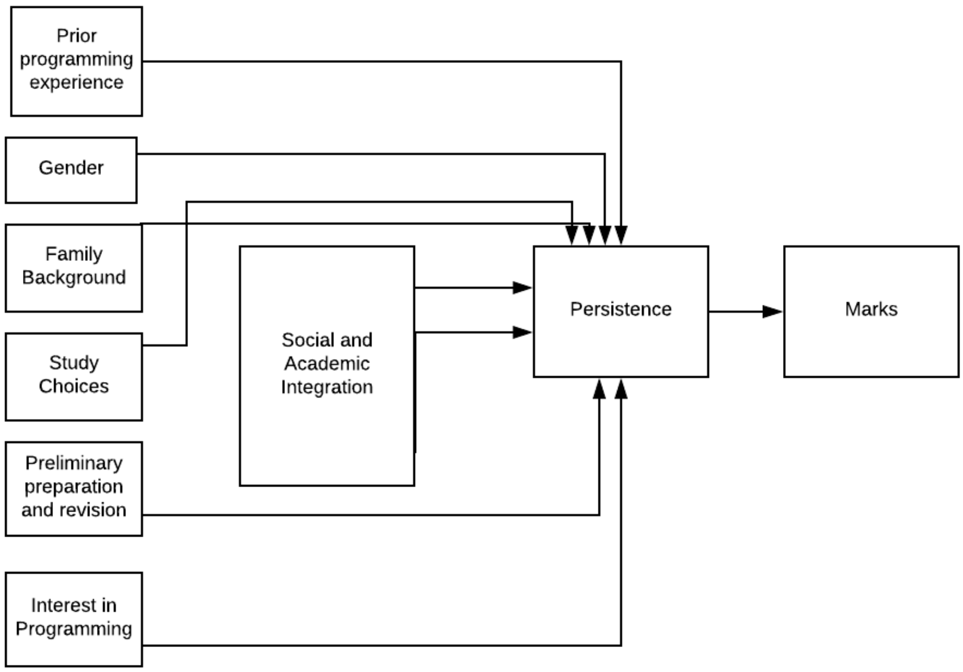


Figure 3-3 : Schematic of the factors analysed in this study

Detailed model of the factors analysed in this study: The detailed model represented in Figure 3-5 outlines the main factors for analysis, along with the subfactors that were investigated in this study. The factors that have already been studied have been denoted by hexagons and the additional factors are denoted by rectangles.The arrows pointing from the factors to the scores show that the impact of these factors on scores was investigated.The arrows pointing from the factors to the subfactors suggest that these factors have further subfactors which were also investigated.

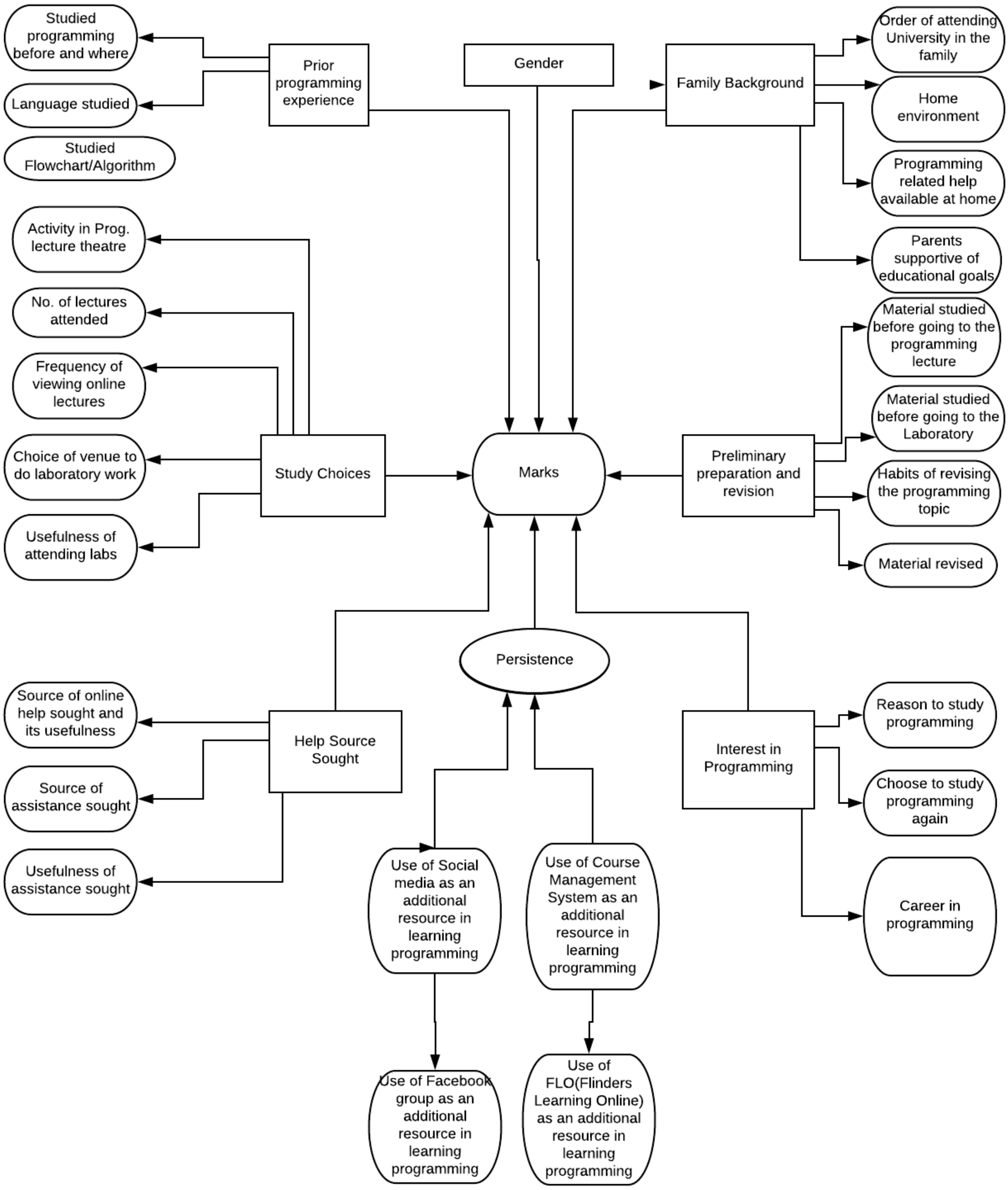


Figure 3-4: Detailed Schematic of the factors analysed in this study

3.4 Why have learning approaches related to preliminary preparation and revision been chosen for this study?

Learning programming is considered different from learning other subjects. So, this method was used to study whether these habits, such as doing preparation before the lectures or laboratory and revision during the semester, helped students to learn more effectively and thus obtain better grades in the programming examination. In particular, preliminary preparation before coming to lectures, practical classes and tutorials was reviewed. Revision completed during the semester was also investigated. Factors investigated for preliminary preparation before lectures included the completion of study notes related to the current lecture, studying notes from the previous lecture, reading lecture slides available on the university website/provided by the lecturer or another source, reading the textbook, and doing online tutorials/reading about the topic to be covered online before the lecture. Factors investigated for preliminary preparation before the laboratory included all the above, plus watching online tutorials/reading about the topic to be covered before the laboratory, reading and reviewing previous laboratory work, practicing previous laboratory work, reading new programs related to previous laboratory work, practicing new programs related to previous laboratory work, and reading and practicing new, similar programs related to the topic to be covered in the laboratory. The revision done during the semester, during mid-semester exams or during the mid-semester break was also investigated.

In most topics, preliminary preparation is needed before coming into the classroom. At Australian University, it is stated in the “Expectations from Students” section of the Introductory Lecture for Java Programming, that students should read the lecture slides before coming to the Lecture(University, 2017a). Other universities also support this flipped classroom approach, as universities state on their website that the actions like looking over lecture notes, knowing in advance what the lecture will be about, doing recommended reading, downloading lecture slides if available and reviewing notes from previous lectures can help students enhance their learning(UNSW Sydney, 2017). But most students are either unaware of this fact as they seldom read the pre-requisites for learning, or they are aware but for various reasons tend to come to the classroom unprepared and without having completed any revision. This study reports on the preliminary research which investigates whether actions taken by the students in the form of preliminary preparation and revision impact on their overall performance.

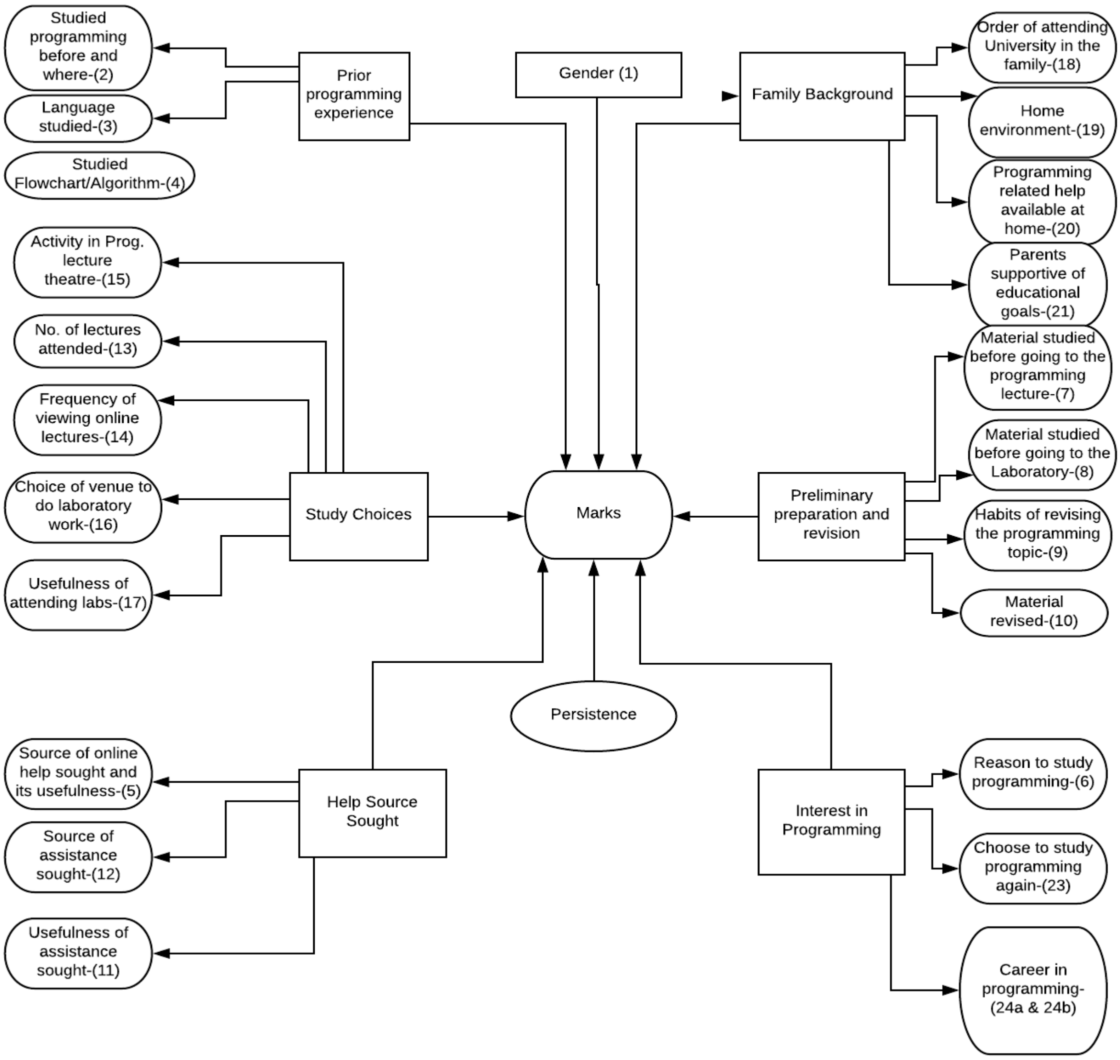
3.5 Why has having an interest in programming been further explored?

To understand the factors that may affect learning of programming, it was important to understand if the student's level of interest in programming has any effect on their learning. As Information Technology is a vast, growing field, the numbers of jobs in this area are increasing and the scope of these jobs is widening. So, the Universities have introduced a learning programming topic in almost all areas of engineering studies. Some students are not interested in learning programming, while others choose to learn. In some informal interactions with the students learning programming at Australian University while tutoring and instructing the topic, some students complained about the topic being mandatory in their field of study. They also complained that they will never study it again in the future. Clearly, they were not informed enough of the use of learning programming at a later stage in their careers. Thus, it was important to investigate if the level of interest in programming has an effect on student performance.

3.6 Benchmarks for assessing Students

In this study students were being assessed for their performance on the basis of their scores obtained in one semester. This is a limitation of the study as the scores in the examination may not truly measure the degree of programming learnt by the students. Significant research in this area suggests that as programming has no agreed, established 'core' list of essential programming concepts, let alone any robust multi-institutional instruments for assessing students' acquisition of programming concepts, a student's mark is the best performance indicator currently available (de Raadt et al., 2005). Thus, as the conclusions were made on the basis of scores obtained in the semester, this may not be a true measure of learning. As this is the only measure available within the scope of this study, it was used to form the conclusions in this study.

3.6.1 Relationships in the research model with the proposed hypotheses numbers



CHAPTER 4 : METHODOLOGY

This chapter explains the methodology used to answer the research questions formulated for this study. The reasons for studying the chosen factors has also been elucidated. The detailed model of the factors to be studied was also a part of this chapter. This chapter also explains how this study was conducted, why an Australian University and an Indian University were chosen for the study, the research participants, why this methodology was used, how the data was collected. Fig 3.1 shows the structure of the methodology chapter:

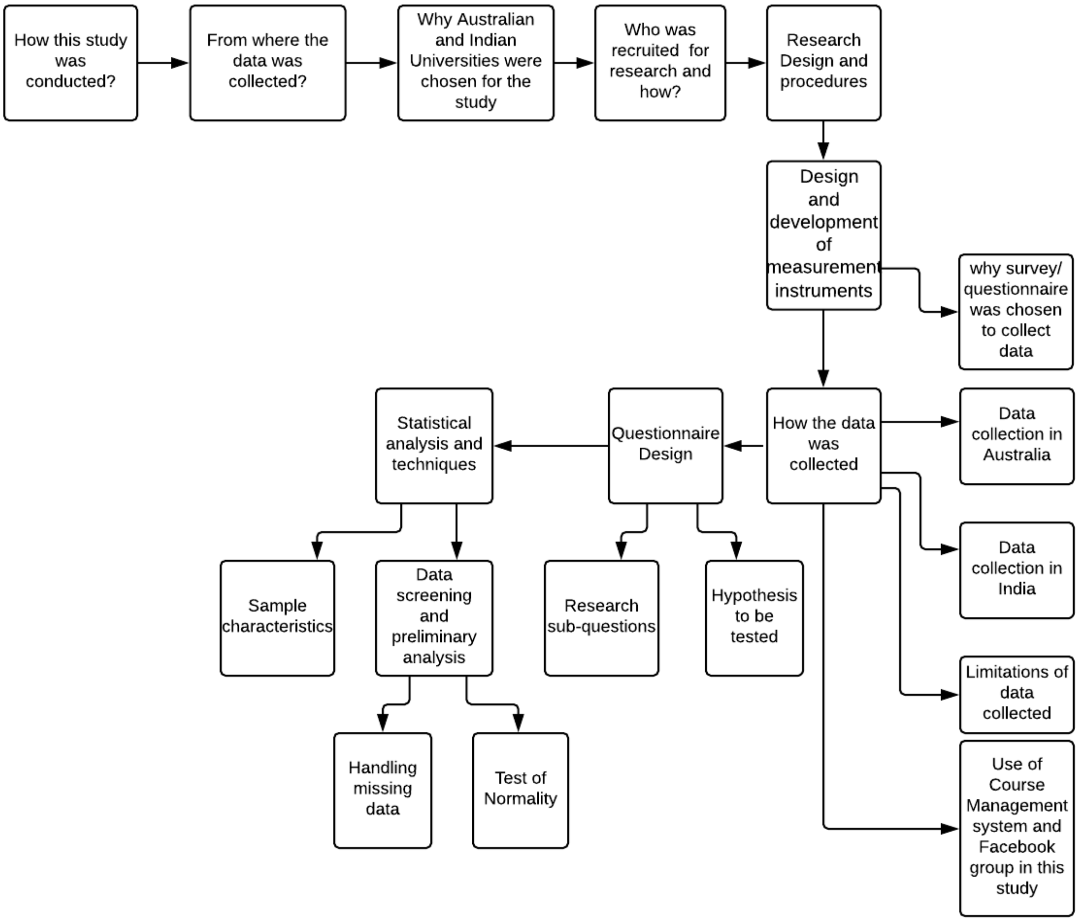


Figure 4-1: The structure of the Methodology chapter

Prior to conducting a study, it is vital to establish a rationale for inclusion and exclusion, along with the methodology. The following section explains why learning approaches and factors that may affect learning programming have been explored and why this comparison between a University in Australia and a University in India was conducted.

4.1 How was this study conducted?

A comparison of the effect of chosen factors from Tinto's model between the two Universities was conducted. The first tool used to conduct the comparison was a questionnaire. A questionnaire was conducted across different semesters over a period of 4 years and the results of the questionnaire were analysed to find the answers to the research questions. Another questionnaire was also used which was completed by the researcher on the basis of the answers provided by course coordinators in the two Universities, who were coordinating the course at the time the study was conducted. This questionnaire was used to find out the teaching methodology and course structure which was used at the time. To find the answer to Research Question 8, a Facebook group was created to investigate the use of Facebook as an engagement indicator and an additional source of peer to peer interaction and social integration in the process of learning programming in comparison with the CMS System (FLO) at Australian University, Australia.

4.2 From where was the data collected?

The data was collected from a reputable University in South Australia: Flinders University and a reputable University in India: Thapar University, Patiala.

4.3 Why were Australian and Indian Universities chosen for the Study?

Most of the research conducted to date has focused on a single learning culture, so it seemed important to explore the impact of chosen factors on different learning cultures. Both Universities are on a par in terms of reputation in their respective countries. Australian

University ranking is 25 in Australia(topuniversities.com, 2017) and Indian University ranking is 26 in India(University, 2017c).

After researching the structure of the tertiary level programming topic/course/subject in Australian, Indian and American Universities, it was found that the course structure of the universities investigated in Australia and America were similar. It was found that the course structure, teaching methodology and teaching approach were different at Australian University, Australia and Indian University, India.

The cultural background, educational background and the facilities available to the students in these two countries also differed in nature. Another difference was in discipline and attendance. For example, the students at Australian University, Australia, were not given compulsory attendance marks for lectures or laboratory classes, whereas at Indian University, India, the students were required to achieve a minimum of 70% attendance in lectures. So, it was important to find out if these differences in learning culture have an impact on student performance in learning programming.

In Indian University, the Topic Co-coordinator had agreed to provide support. He was contacted by telephone in the first instance and thereafter by email. After he agreed to provide assistance by asking the students to complete the questionnaires, the research process started. The researcher visited the Indian University Campus once to meet the Topic Co-coordinator to discuss the research with him and received complete support from him.

4.4 Population/Sampling

4.4.1 Participants:

Students studying programming at tertiary level at Australian University in Australia and Indian University in India were recruited to be candidates for the research.

Participation was voluntary and the students learning programming were introduced to the research in a lecture by the lecturer.

A preliminary questionnaire was conducted to test the questionnaire design. It was conducted in the second semester of 2011 at Australian University. After some revision, a final questionnaire was conducted at Australian University and at Indian University.

At Australian University, Australia, the questionnaire was conducted in three different semesters.

In Australia, an introduction to the research was given during the lecture and a link to the questionnaire was given to the students via the student learning system. The students were also sent emails with the questionnaire link, for easy accessibility. The students were also given hard copies of the questionnaire to complete during laboratory time; permission to do so was sought and given by the ethics committee (6126 SBREC). Participation was voluntary throughout the process. The whole process was conducted under the guidelines of the Australian University Ethics Committee.

For the second part of the study, the participants were recruited from Flinders University, Australia. The study was introduced in the class and the participation was voluntary. The permission was sought from ethics committee (6224 SBREC) before the process started.

In India, the research was conducted under the supervision of the Head of Computer Science and Information Systems. The students were introduced to the questionnaire through the questionnaire link in their laboratory by their course coordinator. Participation was voluntary.

4.5 Research Design and procedures

This research is *mixed-methods* research, in which both quantitative and qualitative research methods were used. Mixed-methods research was used because it is a flexible approach, so the design of the research was determined by what the researcher wanted to find out, rather than by any predetermined epistemological position. In this mixed-methods research, quantitative components predominated. The hypotheses to be tested were generated from the experience of the researcher as a lecturer in India and a tutor and demonstrator in Australia, using exploratory qualitative research along with a review of the literature. Most of the variables to be studied were defined by the researcher on the basis of Tinto's conceptual model and designed through experience, observation and the literature review. Some unexpected variables emerged due to the exploratory qualitative research method.

The variables to be studied changed across the semesters due to changes in the questionnaire design and content. The questionnaire design and content changed due to experience and the level of expertise the researcher accrued with time, after detailed observations, and suggestions from the supervisor and statistical consultant.

The quantitative research design used is descriptive, as the subjects were measured once and the associations between the variables were established(Hopkins, Hopkins, 2006).

The research questions were predominantly inferential, that is, trying to explain the phenomenon: some were descriptive. A co-relational investigation was undertaken to determine the extent of the relationships between two or more variables using statistical data (Shirish, 2013).

An empirical, systematic, investigative approach was used to collect the data to test hypotheses. The hypotheses were tested against the data collected from the study and was either proven correct or false. Deductive reasoning was used to analyse the data and the data collection was repeated for three semesters to ensure repeatability of the results.

The tools used to collect the data were survey questionnaires and data logging.

Intervention and *Observation* techniques were used in this study. For the first part of the research, to study the factors affecting student performance, an observation technique was used. Observational studies, also called epidemiological studies, are those where the investigator is observing natural relationships between factors and outcomes. To study the use of the Facebook group and CMS to improve student engagement and as an additional source of help in learning programming, an intervention technique was used. Interventional studies, also called experimental studies, are those where the researcher intercedes as part of the study design. The introduction of CMS was an intervention, as the students chose whether or not to participate in the discussion forum. Similarly, the introduction of a Facebook group was an intervention whereby students chose whether or not join the group. Their performance was compared with those students who chose not to participate in the discussion forum of CMS or join the Facebook group. The CMS and Facebook group research could not be conducted in India as the required support could not be obtained to conduct the study. Also, the students are not permitted to use any kind of mobile device in the academic area(University, 2018) as per the University policy.

4.5.1 Design and development of measurement instruments

The self-completed questionnaire was chosen as a method for data collection after analysing other methods for the following reasons:

1. It was convenient for the students to complete questionnaires, as they could do so online, in their own time.

2. A postal questionnaire was also used, as some students may not wish to complete the questionnaire online. They were handed the questionnaires in a stamped, addressed envelope for convenience.

3. The self-completed questionnaire took less student time as compared with interviews or observations.

4. Since the details were provided by the students themselves, no erroneous data was collected.

5. As the research was explained to the students by the lecturer, the students were aware of the context of the research; thus, the chances of obtaining reliable data were increased

4.5.2 Why was a questionnaire chosen as an instrument to collect data?

Existing studies on programming found in the literature have conducted research using a self-completed questionnaire as the tool for collecting data from the students. This suggests that the best way to get information from the students regarding their attributes was through questionnaires. In a study conducted at Monash University, Australia, a self-completed questionnaire was used as a tool to collect data from the students enrolled in the programming topic (Butler and Morgan, 2007). Another study to examine “Predictors of Success in a First Programming Course” also used a questionnaire as one of the tools to collect data from the students (Fincher et al., 2006)

Richard Light, a nationally recognized proponent of conducting research, especially in the area of student assessment, believes that good research is one of the most important bases for sound decision-making, including the wise use of questionnaire/survey research and can lead to improvements throughout an institution (Light, 1990).

Thus, a questionnaire was the tool chosen to collect data from the students for the first part of the study. The type of questions included was a combination of *closed* and *open-ended* questions. Closed questions provided respondents with a defined set of answers from which to choose. The response set included both *categorical* and *scaled* responses. Categories were created to cover all possible responses and were mutually exclusive. The questionnaire page also recorded the respondents' IP addresses to identify any respondents who answered the questionnaires more than once. All responses to the questionnaire were voluntary and all participants were encouraged to complete all questions in the questionnaire (Richard et al., 2009).

Another questionnaire was used in this study to get responses from the lecturers. The questionnaire was completed by the researcher on the basis of responses from the topic coordinators.

This questionnaire was used to collect data related to the course structure, teaching methodology and teaching approaches used in the teaching of programming. The best way to get information was through the questionnaire and thus it was used to gather data from course coordinators.

There are certain limitations of using a questionnaire as a tool for data collection as the students might furnish incorrect details about themselves. For example, the students may not be doing any preliminary preparation and still stating claiming to do so. However, the only way to collect data about student learning approaches is through asking students or their lecturer, who may ask the students questions about the topic before teaching each lecture. The process of asking the students questions individually would be very time-consuming and leave no time for the lecturer to teach, thus the only possible tool to collect the data was through questionnaires. The other methods were beyond the scope of this study.

A second tool, data logging, was used to conduct the second part of the study. Data logging was done using the Facebook group and the Australian University Course Management System. Data logging was used for this part of the study as data was automatically logged when students used the CMS or Facebook group, so it was readily available for use and analysis. Permission was sought to use the data from the ethics approval committee. The nature of the data was more reliable as it was collected directly from the source, so user entries could not be missed or recorded incorrectly. Only the initial effort of setting up the Facebook group and introducing it to the students were required for data collection, as compared with data collection using a questionnaire and interviews. The students were already aware of the CMS but they were reminded of the availability of its group discussion feature, which was later used as a data resource.

4.6 How was the data collected?

The data was collected across four semesters in Australia and three semesters in India. The web-based questionnaire used both *internal* and *external approaches* to obtain the respondents (Biffignandi and Toninell, 2005). The internal approach included publishing the questionnaire on the CMS, or FLO. The external approach sent invitation emails to three mailing lists across three semesters in Australia and three mailing lists across three semesters in India.

4.6.1 Data Collection in Australia

Permission for the data collection was sought from the ethics committee. Data collection started only after permission was granted.

The preliminary round of data collection was conducted with those students enrolled in the tertiary level programming topic at Australian University. It involved using the preliminary design of the questionnaire.

The students were given the paper-based questionnaire in the laboratory and participation was voluntary. In this phase of data collection, 33 students responded out of 35 students. An exploratory study was completed on the questionnaires and the questionnaire was altered in the light of the responses given. The initial design of questionnaire was objective and was later modified to use a Likert scale.

The questionnaire was collected over a span of three semesters.

The students were introduced to the research during the lecture and a link to the questionnaire was put on FLO: the student learning system at Australian University. Questionnaire collection started in 2011, when a preliminary questionnaire was prepared and around 50 students were approached to complete the questionnaire. Out of 50 potential respondents, 33 students completed the questionnaire and the data was analysed. In the first semester of 2013, a total of 180 students were approached: only 41 students responded by completing the questionnaires. The questionnaire was given to the students in the laboratories and a link to the questionnaire was sent to the students by the topic coordinator teaching the Java topic to the students. After receiving very few responses, a reminder email was sent to all the students enrolled in the Java topic. Despite the efforts made to collect the data, only 41 students in total completed the questionnaire. In the second semester of 2013 another effort was made to collect data from the students. A total of 205 students were enrolled in the topic and approached for questionnaire completion. Out of 205 students, 61 students dropped the topic and did not do any work related to the topic. Another 90 students failed the topic and did not appear for a supplementary examination. These were the students who did some work at the beginning of the topic but left after a few weeks. This time the students were introduced to the research at the beginning of the topic. The students were approached in the lecture theatre and were asked to voluntarily complete the questionnaire. The students were approached in tutorials, as well as during the period of their viva exams. In addition, the students were given the link to the online questionnaire, available on Survey monkey to complete in their own time. Finally, the students were approached immediately after their theory examination with hard copies of the questionnaires, along a stamped return envelope. The students were asked to complete the questionnaire and post it back. 48 out of 144 active students completed the questionnaire. Another attempt at data collection was made by sending reminder emails to the students, along with the questionnaire link. Thus, data collection remained an issue, as the students were not willing to participate and complete the questionnaire.

During the third round of data collection, the option of approaching the students in the laboratory was taken, along with the availability of the link on FLO. Permission to conduct the questionnaire on University premises had already been sought from the Flinders Ethics Committee and was used extensively during the last round of data collection. A total of 192 students were approached to answer the questionnaire. Again, both *internal* and *external* approaches were used to collect data. This time a total of 103 students completed the questionnaire.

4.6.2 Data Collection in India

The data collection in India was comparatively smooth as the culture at Indian Universities is different from the culture in Australia. The students generally consider requests made by their lecturer positively. So, during the first round of data collection a total of 70 students were approached to complete the questionnaire by introducing them to the questionnaire through the questionnaire link and all of them completed it. It was the same in the second and third rounds of data collection. During the second round, 46 students and during the third round 56 students completed the questionnaire. No hard copies of the questionnaires were handed out and the whole data collection process was delivered via an online link.

4.6.3 Limitations of the Data Collected

No data set can be perfect, and the data collected during this research was of no exception. There are certain limitations to the collected data. In Australia, during the first and second round of data collection, it was observed that most of the students who completed the questionnaire online were either high performing students or the students who passed the examination. So, there may be a bias in the results. To solve this issue, many students were approached in the laboratories at the time when it was expected that a large number of students would turn up to get their work marked. As a result, more responses could be collected in the final rounds of questionnaire completion. Also, the students who enrolled in the topic but dropped out after a few weeks were also approached to complete the questionnaire, but none of those who dropped out responded. Thus, the students who dropped the topic were excluded from this study. There is a gender bias amongst the respondents but the gender bias exists at the core level of participants, as the percentage of males opting to study Information Technology/ Engineering degrees is greater than for females in Australia, but not in India.

4.7 Use of the Facebook group and CMS

This section describes the use of the Facebook group in this study, that is, how the group was created, features of the Facebook group, how it was introduced to the students and the usage of CMS.

Method of Conducting the Facebook and CMS study. After the introduction of the Facebook group, the activity taking place within the group was recorded. The group activity was monitored by the researcher. Similarly, the students were introduced to the Course Management System (FLO) and told that they could ask topic related questions and the answers would be given by the lecturer, as well as demonstrators.

Creation of Facebook group: The Facebook group was created by the researcher.

Features of Facebook group: It was a private closed Facebook group, which means that it was not open to the public and only those students authorised by the administrator could become members of the group.

Introduction of the group to the students: The students were introduced to the Facebook group by the lecturer teaching and co-ordinating the topic. Though the group was introduced by the lecturer, the lecturer did not have access to the group content. The group was administered by the researcher. At the time of introduction of the group to the students, they were informed of the fact that the content would not be viewed by the lecturer. It was introduced as a peer to peer support and discussion group.

A link was provided on the student learning system (FLO) and the students could join either by clicking on the link or copying and pasting the link to the browser. The link was also shown to the students in lecture slides. A verbal script detailing the purpose of creating the group was explained to the students and an information sheet, along with verbal script and ethics approval document, was provided on FLO. Thus, the students were provided with complete information about the research, as well as its purpose. The students were also informed that their chat conversations would be monitored by the researcher and their names would not appear in the thesis or papers resulting from the experiment. Participation in the experiment was voluntary.

Specific instructions to the students: When the Facebook group was introduced to the students, some guidelines were given stating that they could not post anything that was not related to the topic being taught. They were reminded to be ethical in their conduct. Students were also told that they could only discuss content related to the topic and could not post answers to the assignments or quiz to avoid plagiarism. No personal discussions were allowed.

The students were asked only to help their peers in the way they would do in the classroom, by sharing their knowledge and ideas and not to provide complete solutions to the laboratory questions or major and minor projects.

Grades Classification

The terms Pass, Higher Distinction, Distinction, Credit and Fail have been used in this chapter. Table 3-1 below defines these grades.

Table 4-1: Grade Definitions

Grade	Scores
Fail	Less than 50
F/A	Fail but could reappear for Supplementary examination i.e. scores >44.25 but less than 50.

Pass	50<=Scores<65
Credit	65<=Scores<75
Distinction	75<=Scores<85
Higher Distinction	Greater than 85

4.8 Questionnaire Design

4.8.1 Research Sub Questions

RQ refers to Research Question

RSQ refers to research sub question

RQ1 Does gender affect students' performance?

RQ2: Do students' habits of preliminary preparation and revision affect students' performance in terms of scores?

RSQ: Please suggest the frequency if you study before going to the programming lecture

- Study lecture slides related to the current lecture available on FLO
- Study textbook slides related to the current lecture available on FLO
- Study lecture slides from the previous lecture available on FLO
- Study textbook slides from the previous lecture available on FLO
- Read paper-based textbook
- Do online tutorials /read about the topic to be covered online before the lecture
- What do you study before going to the Laboratory?

RSQ : Please tell us about your habits for revising the programming topic.

- During mid-semester break
- During mid-semester exams
- Both during mid-semester break and mid-semester exams
- Revised while the semester was in progress

RSQ: If you happen to revise the programming topic, please let us know if you revise the following?

- Theory from lecture slides available on FLO
- Textbook Slides available on FLO
- Laboratory Work
- View lectures online
- Revised on a website designed to revise the topic
- Revised previous week's laboratory work
- Revised New similar programs
- Read previously done laboratory work
- Redo previously done laboratory work
- Read new similar programs
- Redo new similar programs

RQ3 Do students' habits in terms of study choices affect student performance?

RSQ: What do you do in the programming Lecture theatre? (You can tick more than one box)

- Listen to the Lectures
- Listen and make notes
- Annotate if you have printed notes
- Play games on mobile phone/laptop
- Look up for terms being discussed in the lecture
- Use social media to socialize
- Browse the internet in general

RSQ: Please write the approximate number of programming lectures you have attended.

- 0%
- Upto 20%
- Upto 40%
- Upto 60%
- Upto 80%
- 100%

RSQ: How often do you view the programming lectures online?

- All
- Important topics
- The ones you find difficult to understand
- The ones suggested by your classmates

- The ones suggested by your Lecturer/Professor (important topics
- If I need to understand a concept again
- If I need to take a note of some key points that I missed during the Lecture

Where do you prefer to do the laboratory work?

- In the laboratory
- At home
- Library

Do you find attending laboratories useful?

- Not at all
- Slightly Useful
- Useful
- Very Useful
- Extremely Useful

RQ4 What kind and source of help is sought by students when required and what sources of help have proven to be useful?

RSQ: If you posted questions related to the topic/course/subject online on social networking or other websites to obtain help, was the help useful or not?

- Google
- Twitter
- Facebook

- email

From what sources do you try to seek help?

- Consult Classmates
- Consult Senior Students who have already passed the topic
- Consult Lecturer
- Read Textbook
- Read Lecture Notes/Slides
- Discuss the problem on discussion forums on FLO
- Discuss the problem on Facebook /twitter
- Discuss the problem on other socializing website
- Opt for private tuition outside University
- Attend help sessions at University

RSQ: Was seeking help from these sources useful?

RQ5 Does intrinsic interest in programming lead to better performance?

RSQ: Why did you choose to study programming? (You can tick more than one box)

- Interested to know about programming
- It is up-coming in the work market
- High paying work in the industry
- Mandatory in the degree

RSQ: If given the option, would you choose to study a topic/subject related to programming again? (i.e. a topic other than that which you have already studied.)

- Yes
- No

RSQ: Would you like to undertake a career or job related to programming?

- Yes
- No

RQ6 Does prior programming experience prove to be helpful in learning programming, leading to better performance?

RSQ: Did you study programming before attending this course at university?

- School (9th or 10th Grade)
- 11th and 12th grade
- At home (out of interest) /self-study
- At University as part of a different degree
- At University as part of the same degree you are currently studying

RSQ: Have you studied any of the languages below?

- C
- C++
- Visual Basic
- Java

- PHP
- Python
- Basic
- COBOL
- VC++
- PASCAL

RSQ: Have you studied designing Flowcharts/Algorithms?

- Yes
- No

RQ7: Does student family background affect student performance?

RSQ: Are you the first one from your family to attend university or do you have other members of your family who have attended university?

- First One
- Siblings
- Parents

RSQ: Is your home environment conducive to study?

- Yes
- No

RSQ: Can you get programming-related help at home from your parents/carers or siblings?

- Yes
- No

RSQ: Are your parents/grandparents/carers supportive of your educational goals?

- Yes
- No

RQ8: Can Social media, i.e. Facebook or the CMS system, help improve student engagement and serve as an additional source of peer to peer interaction and social integration in the process of learning programming?

This study analyses the implications of various factors, such as prior programming background, preliminary preparation, revision and the type of revision on student performance in terms of scores/grades. It compares the impact of the above-mentioned factors in two different countries, Australia and India, and investigates the factors that affect the learning programming process positively. The study also investigates if peer-to-peer interaction and social integration can be achieved through the use of social media tools such as Facebook and CMS, to help improve student engagement in the programming topic and provide a collaborative learning environment to serve as an additional resource for students learning programming.

To answer the research questions, a questionnaire was designed to gather information from the students.

P refers to page and Q refers to Question.

P1 Q2: Degree enrolled in at Tertiary level of education

This question was asked to find out if the degree a student has enrolled in has an impact on student performance. Programming is taught to students enrolled in engineering degrees and it was found from the initial analysis of the preliminary questionnaire that certain students were not interested in studying the programming topic as they thought that it would be of no use later in their career.

P1 Q3 Gender

Students were asked about their gender, as several studies have shown that the number of males who choose to study programming outnumbers the number of females. It was important to find out if there is uniformity in the pattern at both universities.

P1Q4

Did you study programming before attending this course?

This question sought to determine if prior knowledge of programming had any impact on learning of programming at tertiary level. There are a few studies that suggest that prior knowledge has a positive impact on student learning of programming, whereas other studies suggest that prior knowledge does not have any impact on learning programming.

P2 Q5 Have you studied any of the languages mentioned below?

To determine if knowledge of any particular language has any impact on student performance, students were given a list of languages they may have studied prior to learning programming at University. Students could tick more than one language/ packages if they had studied more than one language/package.

P2 Q6 Have you studied designing Flowcharts/Algorithms?

To determine if the knowledge of Flowcharts/Algorithms had any impact on student learning, the students were asked about their experience. Some schools teach flowcharts or algorithms during secondary school, so it was important to study its impact on student performance. Students could tick either or both of the options depending on their prior study.

P2 Q7

Why did you choose to study programming? (You can tick more than one box)

Students were asked about the reason they choose to study programming. This was asked to determine if the reason for studying programming had any impact on their performance. The students had the choice to tick more than one option. This question was asked to determine if the students who choose to study programming by choice perform any differently from those who are forced to study programming.

P2 Q8

If you posted questions related to the topic/course/subject online on social networking or other websites to obtain help, was the help useful or not?

This question was asked to determine if social media or other websites may be useful in learning programming. A lot of information related to programming is available online. Seeking an answer to this question may provide an insight into the websites or social media that prove to be most useful. The social media and websites considered were Google, Twitter, Facebook and email.

P3 Q9

Please suggest the frequency if you study before going to the programming Lecture?

This question was asked to determine if doing preliminary preparation before the programming lecture proved to be helpful in learning programming. Various choices of the type of preliminary preparation done by the students were given and they could choose to respond by answering *yes* or *no* to a series of options.

P3 Q10

What do you study before going to the Laboratory?

This question was asked to determine if doing preliminary preparation before the laboratory proved to be helpful in learning programming. Various choices of the type of preliminary preparation done by the students were given and they could choose to respond by answering *yes* or *no* to a series of options.

P4 Q11

Please tell us about your habits for revising the programming topic?

This question was asked to determine if doing revision during the mid-semester break, during mid-semester examinations, both during mid semester break and mid-semester examinations or revision while the semester was in progress proved to be helpful in learning programming. Various choices of the timing of revision done by the students were given and they could choose to respond by answering a 5-point Likert scale.

P4 Q12

If you happen to revise the programming topic, please let us know if you revise any of the following?

This question was asked to determine the most useful kind of material used during revision. The usefulness of the material was determined by its impact on student scores. The various options were given to the students are in a 5-point Likert scale format.

P5 Q13

From what sources do you try to seek help?

To determine the sources of help students seek the most, when required, this question was asked using a 5-point Likert scale format.

P5 Q14

Was seeking help from these sources useful?

This question was asked to find out if the source of help sought was useful to the student. This question was asked using a 4-point Likert scale format.

P6 Q15

Please give the approximate number of programming lectures you have attended.

The number or percentage of programming lectures attended may help to determine if attending lectures was helpful to the students in learning programming.

P6 Q16

How often do you view the programming lectures online?

This question was asked to determine the effect of viewing recorded lectures online. The question was asked using a 5-pt Likert scale format and the type of lectures viewed/reasons to view lectures were given as options to choose from.

P6 Q17

17. What do you do in the programming Lecture theatre?

(You can tick more than one box)

This question was asked to determine the kind of activities performed in the lecture theatre and whether they have any impact on student learning of programming. The question was asked using a 4-point Likert scale format and the kinds of activities were given as options to choose from.

P6 Q18

Where do you prefer to do the laboratory work?

This question was asked to study the impact of the place where the students choose to do the laboratory work. It was asked in objective form and the students could reply as *yes* or *no* to *laboratory, library* or *home*. It was asked as studying in a standalone environment like home may impact student learning. The availability of demonstrators in the laboratory and the availability of peers in the library may prove to be an additional support to the students.

P6 Q19

Do you find attending laboratories useful?

This question was asked to find out if the students find attending laboratories useful at both universities. Its impact on student performance was also needed evaluation. The question was asked in a 5-point Likert scale format and the students had to respond to the degree of usefulness of attending laboratory sessions.

P6 Q20

Please give reasons why you find attending laboratory sessions useful.

The students were asked to explain why they found attending laboratory sessions useful, as this may give an insight into the reasons why some students find attending laboratory sessions useful, while others may not.

P7 Q21

Are you the first one from your family to attend University, or have other members of your family attended University?

This question was asked to determine if being a part of a family where parents/grandparents/carers or siblings attended university proved to be useful to students.

P7 Q22

Is your home environment conducive to study?

This question was asked to determine if the home environment had any effect on student performance.

4.8.2 Hypotheses to be tested

Main Null Hypothesis:

Factors and attributes that affect the learning of programming in Australia and India are not same.

The Social media, i.e. Facebook or the CMS system, help improve student engagement and serve as an additional source of peer to peer interaction and social integration in the process of learning programming

Research Sub-Questions

These include sub Null Hypothesis

- Does Gender affect the performance of students?

Hypothesis1

$H_{0:}$ Gender of students does not affect the performance of students.

$H_{1:}$ Gender of students affects the performance of students.

- Does prior programming experience affect the performance of students in terms of scores?

Hypothesis 2

H_0: Studying programming anywhere or anyhow does not affect the scores of the students.

H_1**:** Studying programming anywhere or anyhow affects the scores of the students

- Does studying any of the aforementioned languages affect the performance/scores of the students?

Hypothesis 3

H_0: Studying any of the aforementioned languages does not affect the performance/scores of the students

H_1: Studying any of the aforementioned languages affects their performance/scores

- Do studying Flowcharts and algorithms affect student performance in terms of scores?

  Hypothesis 4

  H_0: Studying flowcharts or algorithms does not affect the performance/scores of the students

  H_1: Studying flowcharts or algorithms affects the performance/scores of the students

- Is posting questions/asking for help on Google, Twitter, Facebook, and by email helpful to the students?

  Hypothesis 5

  H_0: Posting questions/asking for help on Google, Twitter, Facebook, and Email is not helpful to the students

  H_1: Posting questions/asking for help on Google, Twitter, Facebook, and Email is helpful to the students

- Does the reason for studying programming affect the scores of students?

  Hypothesis 6

  H_0: Reasons to study programming do not affect the scores of students

  H_1: Reasons to study programming affect the scores of students

- Do the students who do any kind of preliminary preparation before going to the lecture perform better in terms of scores than those who do not? What kinds of the above-mentioned preliminary preparation are most helpful in terms of scores?

  Hypothesis 7

  H_0: Preliminary preparation is not helpful in improving the performance of the students

  H_1: Preliminary preparation is helpful in improving the performance of the students

- Is preliminary preparation before the laboratory helpful in getting better scores?
- What kind of preliminary preparation before laboratory is helpful in getting better scores?

Hypothesis 8:

H_0: The students who do preliminary preparation before the laboratory do not perform better than those who do not.

H_1: The students who do preliminary preparation before the laboratory do not perform better than those who do not.

- Does revision help support better performance of students in terms of scores?
- At what time in the semester does revision prove to be the most beneficial in terms of obtaining good scores?

Hypothesis 9

H_0: Revision does not help generate better performance and thus obtaining good scores

H_1: Revision does not help generate better performance and thus obtaining good scores

- What kind of revision is the most beneficial in terms of obtaining good scores?

Hypothesis 10

H_0: One kind of revision is not helpful over the other kinds of revision in terms of scores obtained by the students

H_1: One kind of revision is not helpful over the other kinds of revision in terms of scores obtained by the students

- Does seeking help from various sources proves to be helpful in terms of scores obtained?

Hypothesis 11

H_0: Seeking help from various sources does not prove to be helpful in terms of scores obtained

H_1: Seeking help from various sources proves to be helpful in terms of scores obtained

- Does seeking help from any source proves to be helpful in terms of scores obtained?
- What kind of help proves to be most beneficial in terms of scores?

Hypothesis 12

H_0: Seeking help from given source does not prove to be helpful in terms of scores obtained

H_1: Seeking help from given source proves to be helpful in terms of scores obtained

- Does attending a higher percentage of lectures lead to better performance in terms of scores obtained?

Hypothesis 13

H_0: Attending a higher percentage of lectures does not lead to better performance in terms of scores obtained.

H_1: Attending a higher percentage of lectures leads to better performance in terms of scores obtained.

- Does viewing lectures online affects student performance in terms of scores obtained?
- What kinds of lectures are viewed online the most?

Hypothesis 14

H_0: The students who view lectures online do not obtain better scores than those who do not

H_1: The students who view lectures online obtain better scores than those who do not

- Do students who perform a particular activity in the lecture theatre get better scores?
- What kind of activity is most prevalent in the lecture theatre?

Hypothesis 15

H_0: The students who perform a particular activity in the lecture theatre do not get better scores than those who do not

H_1: The students who perform a particular activity in the lecture theatre do not get better scores than those who do not

- Does practising laboratory work at a particular venue help to improve the performance of students in terms of scores?

Hypothesis 16

H_0: Practicing laboratory work at a particular venue does not help to improve the performance of students in terms of scores

H_1: Practicing laboratory work at a particular venue does not help to improve the performance of students in terms of scores

- Do the students who find attending laboratories useful perform better than those who do not?

Hypothesis 17

H_0: The students who find attending laboratories useful do not perform better than those who do not.

H_1: The students who find attending laboratories useful perform better than those who do not.

- Do the students whose parents/carers or siblings attended University perform better than those who did not?

Hypothesis 18

H_0: The students whose parents/carers or siblings have attended University do not perform better than those who did not.

H_1: The students whose parents/carers or siblings have attended University perform better than those who did not.

- Do the students whose home environment is conducive to study perform better than those whose do not?

Hypothesis 19

H_0: The students whose home environment is conducive to study do not perform better than those whose do not

H_1: The students whose home environment is conducive to study perform better

- Do the students who can get programming-related help at home from parents/carers or siblings perform better than those who do not?

Hypothesis 20

H_0: The students who can get programming related help at home from their parents/carers or siblings do not perform better than those who do not

H_1: The students who can get programming related help at home from their parents/carers or siblings perform better

- Do the students whose parents/grandparents/carers are supportive of their educational goals perform better than those who do not?

Hypothesis 21

H_0: The students whose parents/carers are supportive of their educational goals do not perform better than those who do not.

H_1: The students whose parents/carers are supportive of their educational goals perform better

- Do the students who are studying this topic for the second or more times perform better than those who are studying it for the first time?

Hypothesis 22

H_0: The students who are studying this topic for the second or more times do not perform better than those who are studying it for the first time

H_1: The students who are studying this topic for the second or more times perform better than those who are studying it for the first time

- Do the students choose to study a topic related to programming again perform better than the students who do not?

Hypothesis 23

H_0: The students who choose to study a topic related to programming again are not better performers

H_1: The students who choose to study a topic related to programming again are better performers

- Do the students who would like to take up a career in programming perform better than those who wo not? *(for Australian University)*
- Would you like to undertake a career or a job related to programming, testing, being a technical writer, or graphic designer? *(for Indian University)*

Hypothesis 24

H_0: The students who would like to take up a career in programming do not perform better than those who would not

H_1: The students who would like to take up a career in programming perform better than those who would not

4.9 Statistical Analysis and Techniques

A range of statistical analysis techniques were applied to examine the research data. IBM SPSS Statistics version 23 (IBM, 2015) and Microsoft Excel were used to analyse the data. During the first phase, the sample characteristics were studied. The statistical analysis began with some preliminary analysis of data collected at the initial stage of the study in 2011. In the first paper that was produced, most of the initial analysis stage was undertaken using Microsoft Excel: percentages and averages were calculated for the data collected. At the later stage, when the final questionnaire design was complete and the data was collected, IBM SPSS Statistics versions 22 and 23 were used to analyse the data. After the collection of data at each stage, data screening, missing data handling, tests of normality and assumption checking were undertaken. The Skewness, Kurtosis, Shapiro-Wilk normality test(Shapiro and Wilk, 1965), and QQ plots(Wilk and Gnanadesikan, 1968) were used to determine if the dependent variable (examination score) were normally distributed.

To determine if there were a relationship between the examination score and variables of interest, Wilcoxon ranked-sum tests (Hollander et al., 2013) (used when the variable of interest had 2 levels) and Kruskal–Wallis tests(Hollander et al., 2013) (used when the variable of interest had more than 2 levels) were conducted. Wilcoxon ranked-sum tests determine if there were a statistically significant difference in examination score between the two groups of interest. Kruskal–Wallis tests determine if there were a statistically significant difference in examination score among the groups of interest. If the results of the Kruskal–Wallis test were significant, Dunn’s procedure (Dunn, 1964) for pairwise comparisons was performed to investigate which two levels of the variable were statistically significantly different in the examination score. A p-value of less than 0.05 indicated significance.

4.9.1 Sample Characteristics:

This study collected data from a total of 205+33 (who were not included in the final analysis as it was a preliminary questionnaire) students from the Australian University and 172 students from the Indian University. The characteristics of the sampled individuals were studied.

Australia: In Australia, the students enter the degree course after completing Year 12 and obtaining a minimum ATAR score.

India: In India, the students enter the degree course after completing their Class 12 study. It is mandatory to have studied Maths, Physics and Chemistry as their core subjects in Classes 11 and 12 to get admission to the degree course at Indian University. Entry is through a national level competitive examination based on the subjects studied in Classes 11 and 12. The admission is made on the basis of merit of score in JEE (Main). In 2015, 1.3 million students took the JEE (Main) examination.

4.9.2 Data Screening and Preliminary Analysis

4.9.2.1 Handling Missing Data

Missing data is one of the most common problems in data analysis of practical research. The questionnaire was carefully re-designed and modified after the initial design of the questionnaire to avoid missing data. Since it was a Likert scale questionnaire, only a few open-ended questions were included in the questionnaire where necessary, so that respondents were only able to provide complete and accurate data. Missing data can affect the data analysis and, if not dealt with carefully, may lead to inaccurate results. The paramount question concerning the issue of missing data is whether these missing values are a function of a random or a systematic process (sagepub.com, Sagepub.com, 2005). The severity of missing data depends on the pattern of missing data, how much is missing, and why it is missing. According to Tabachnick and Fidell, the pattern of missing data is more important than the amount that is missing(Tabachnick and Fidell, 2007). Missing values that occur randomly through a data matrix create less severe problems than non-randomly missing values which are severe, no matter how few they are, because they affect the results. There are a number of methods used to handle missing data, such as deleting cases, using mean substitution, using a missing data correlation matrix, and treating missing data as data. After the data collection process was complete, the first step in the data analysis was to screen the data. As a general rule, variables containing missing data on 5% or fewer of the cases can be ignored (Tabachnick and Fidell, 2007).

Australian University: The data collected had variables exceeding this 5% mark which was small enough to ignore. Thus 13 cases could not be analysed for further statistical analysis, as some of them had missing values in their response. The missing value in these 13 cases was the main variable. Emphasis was given to the completeness of the data on the main variables relating to the investigation of research questions. This reduced the sample size from 211 to 198.Out of these 198, 4 cases were excluded as the value of the dependent variable to be studied was missing. Students who had answered over 90% of the questionnaire items (i.e., with 9 or fewer missing responses) were kept in the data analysis for this study. The final sample size for Australian University was 184. Students who had answered over 90% of the questionnaire items (i.e., with 9 or fewer missing responses) were kept in the data analysis for this study. Out of the 198 questionnaires collected, the final sample size for Australian University was 184.

Indian University: Students who had answered over 90% of the questionnaire items (i.e., with 6 or fewer missing responses) were kept in the data analysis for this study. The total number of questionnaires collected was 172. Out of 172 questionnaires, 78 questionnaires were rejected as they were completed by Masters Students who were studying programming for the first time. The reason was miscommunication, as the researcher had explained that questionnaires from first year students were required, so the first year Masters students were also asked to complete questionnaires. Thus, the final sample size reduced to 94. Out of 94 questionnaires collected, the final sample size for Indian University was 79.

4.9.2.2 Test of Normality:

Statistical errors are common in scientific literature and about 50% of the published articles have at least one error (Ghasemi and Zahediasl, 2012).To ensure that the statistical analysis is accurate and without errors, the collected data needs to be checked for normality. The tests to be performed on the data are chosen based on the distribution of data.

The Skewness, Kurtosis, Shapiro-Wilk normality test(Shapiro and Wilk, 1965), and QQ plots(Wilk and Gnanadesikan, 1968) were used to determine if the dependent variable (examination score) were normally distributed.

For both universities, as the data were not normally distributed, Wilcoxon ranked-sum tests and Kruskal–Wallis tests were used to determine which factors were statistically significantly related to the three examination scores.

4.10 Summary

This chapter describes the reasons for investigating the learning approaches and factors in this study. A detailed model of the factors to be analysed is also presented. This chapter also describes the research methods used and why these research methods were chosen for the study, from where and how the data was collected. Also, it describes the statistical techniques used and the kinds of statistical analysis performed. Further details will be provided in the subsequent chapters. The next chapter studies the similarities and differences between Australian and Indian universities chosen for this study.

CHAPTER 5 : SIMILARITIES AND DIFFERENCES BETWEEN THE LEARNING CULTURES IN THE TWO UNIVERSITIES

This chapter introduces the similarities and differences between the learning culture at Australian University, Australia, and Indian University, India, in terms of teaching approaches, study culture, examination structure and assessment structure. To conduct this study, it is important to find out the similarities and differences between the two universities chosen for this study, so that further conclusions based on the results can be made.

5.1 The study culture in Lecture theatre and Laboratory

5.1.1 Attendance:

At Australian University, the students are not required to attend lectures or laboratories. It is the students’ choice to attend or miss the lectures and they may choose to do the laboratory work in a laboratory session at the scheduled time or at any other venue of their choice. A lecture recording is available to the students within one to two hours of its delivery and students may watch the lecture recording at any time. The lecture recording is available to the students until the end of the semester. There is no limit to either the number of lectures they watch online or the number of times they may watch the lectures.

At Indian University, attendance at lectures and laboratories is compulsory. The students must attend a minimum of 75% of lectures and have to be present in at least 75% of laboratory sessions. No lecture recording is available to watch later and the laboratory work is assigned each week, to be completed in that week.

Use of mobile devices: At Australian University, the use of mobile devices and laptops was permitted in the lecture theatre, whereas the use of mobile devices and laptops was not permitted at Indian University. Thus, the students at Australian University were free to use their laptops, tablets or even mobile phones, which may be useful at times and a source of distraction at other times. At Indian University, the students could only listen to the lecture or make notes by hand, as the use of laptops, tablets and mobile phones may distract the students from the content of lecture.

5.1.2 Residence:

Australia: At Australian University, there is very limited provision of residence at the campus. Most students reside off campus and meet each other only during the lectures, laboratory sessions or workshops, if and when they attended. Thus, the opportunities to collaborate with each other for study purposes were minimal. The students only had formal meeting times. The Literature suggests that collaborative learning enhances the learning experience (Teague and Roe, 2008). The students at Australian University may collaborate informally if they join study groups where the other students studying the same topic collaborate to support each other in learning, clearing up doubts and even motivating each other.

India: At Indian University there is paid provision of hostels and most of the students stay in the hostels. So, while this study was conducted, the students were living in a hostel provided by the University and were surrounded by other students almost all the time, who were studying the same topic. The grouping of the students in the hostels (residences are called hostels in India) was also according to the course they were enrolled in. So, generally, there were two students studying the same course in a room. The adjoining rooms also generally contained students belonging to the same degree course. Thus, the students studying the same topic were surrounded by other students studying the same topic almost 24 hours a day. It is common for students to support each other when asked for help. They could walk up to a classmate/course mate to ask for help if required. So, the students not only had formal but informal meeting times as well, which improved their chances of collaboration for learning purposes.

5.2 Examination Structure:

Australia: The examination structure at Australian University involves an aggregate of work done during the semester which includes a Quiz, Laboratory assignments or projects and a practical or theory examination at the end.

Quiz: The quiz involves objective type questions based on the course material taught in the lectures each week in a progressive manner.

Laboratory assignments: The laboratory assignments involve practical exercises on the basis of course material taught in the lectures each week in a progressive manner.

Theory examination: The one-hour theory examination involves objective type questions.

India: The examination at Indian University also involves an aggregate of work done during the semester.

Mid-Semester Examinations: The mid-semester examination includes a theory examination, which covers the course material taught till the examination is taken. The structure of the theory examination involves subjective, practical and objective questions but primarily is a subjective examination along with practical problems that must be solved on paper. The students have to answer the examination in two hours.

Practical Examinations: The practical examinations include practical problems to be solved by students, assigned to them at the time of examination. The examination duration is 1.5 hours.

A *Quiz and VIVA* was conducted after the examination and the students are asked questions based on the practical examination taken, as well as the course material studied throughout the semester. Its duration is 15 minutes.

Along with the formal examinations conducted by the University, the lecturer sometimes conducts informal class tests to judge their understanding of the topic informally. This gives students feedback about their learning and the key areas they need to work upon before the formal assessment. Such tests are usually conducted throughout the semester.

5.3 The Assessment Practice

5.3.1 Assessment Practice at Australian University:

Though the course objective, course structure and course material were same for each semester, the assessment structure and structure of the laboratory work changed across the three semesters when this study was conducted. It was the same in Semester 1, 2014 and Semester 2, 2014 but it was different in Semester 2, 2013.

2013 Semester 2

Laboratory Work: The Laboratory work involved programming on two projects categorised as MINOR and MAJOR projects. For the MINOR project, the students had to submit their responses in three different parts. The first part was MINOR A, and then they had to submit MINOR B and then MINOR C. For the MAJOR project the students had to submit it in two different parts. The first part was MAJOR A and the second part was MAJOR B. The scores were assigned for the total of MINOR and MAJOR projects. The students had to take a Quiz in weeks 3, 4, and 5 and at Mid-Semester. At the end of the semester, the students had the option to take a theory examination. The final score was an aggregate of their scores in the Quiz and the best two out of the MINOR, MAJOR and Theory examinations.

Final Score= Scores in Quiz + Best 2 of (Scores in MINOR, Scores in MAJOR, Scores in Theory Examination)

2014 Semester 1

In Semester 1, 2014 the laboratory structure changed. The students had to complete 10 laboratory exercises and take 10 quiz sessions during the semester. The students also had to complete two assignments and take a practical examination. An optional bonus task was also given to the students to help them pass the examination.

Final Score= Laboratory Exercise (30%)+ Quiz (20%)+ 30% of (Assignment 01

Assignment 02) + Practical Examination (20%)+ Bonus Task (10%)

2014 Semester 2

The laboratory structure in Semester 2, 2014 was same as the laboratory structure in Semester 1, 2014. The bonus task was not given in this semester.

Final Score= Laboratory Exercise (30%)+ Quiz (20%)+ 30% of (Assignment 01

Assignment 02) + Practical Examination (20%)

5.3.2 Assessment practice at Indian University, India:

At Indian University the students are assessed based on work done during the semester. This involved three examinations that were taken by students at regular intervals throughout the semester. The final score was an aggregate of the score obtained in the three exams. In addition to the work done in the laboratory, a practical test taken at the end of the semester is also added to the final score.

Final Score in the examination: Scores in Mid-Semester Examination 1 (12.5) + Scores in Mid-Semester Examination 2 (12.5) + Scores in Mid-Semester Examination 3(45) +Laboratory Work (Two laboratory work evaluations (20)) + Quiz & VIVA (10)

Duration of Mid-Semester Examination: 2 Hours

Duration of Quiz: 15 Minutes

Duration of Laboratory examination: 1.5 Hours

5.4 Summary:

This chapter analyses the similarities and differences in the educational cultures at Australian University, Australia, and Indian University, India, in terms of the approaches used to teach programming, the teaching structure involved, the assessment structure and the examination structure, alongside a number of cultural similarities and differences between the two universities. It was found that there are many differences in terms of the teaching of programming, the University culture, examination structures and assessment structures. The next chapter compares the data of both universities through the application of statistical tests to find out the factors that may affect the learning of programming in terms of scores.

CHAPTER 6 : ANALYSIS OF COLLECTED DATA

This chapter presents a statistical analysis of various factors that may affect the performance of students in terms of scores for the two universities. The statistical analysis of the data is important to make conclusions based on the results. The discussion for each hypothesis has also been done in this chapter which was related to the findings back to the literature and to the results reported by other researchers. The comparison seeks to find out if the factors affecting the learning of programming in Australian University and Indian University, are same or different, despite the differences in educational culture.

6.1 Introduction and methods

Questionnaire responses regarding various aspects of programming skills were recorded for students at the two universities. Examination scores for the students were recorded. The purpose of this study was to determine those learning and teaching factors that are statistically significantly related to examination scores for each university.

Frequency tables were used to summarize the questionnaire responses. Descriptive statistics were used to summarize the examination scores. The Skewness, Kurtosis, Shapiro-Wilk normality test(Shapiro and Wilk, 1965), and QQ plots(Wilk and Gnanadesikan, 1968) were used to determine if the examination scores were normally distributed. To determine if there were a relationship between the examination scores and variables of interest, Wilcoxon ranked-sum tests (used when the variable of interest had 2 levels) and Kruskal–Wallis tests (used when the variable of interest had more than 2 levels) were conducted. Wilcoxon ranked-sum tests determine if there were a statistically significant difference in examination score between the two groups of interest. Kruskal–Wallis tests determine if there were a statistically significant difference in examination score among the groups of interest. If the results of the Kruskal–Wallis test were significant, Dunn's procedure for pairwise comparisons was performed to investigate which two levels of the variable were statistically significantly different in the examination score. A p-value less of than 0.05 indicated significance.

6.2 Sample description for Australian University

Data from 198 students (48 from 2013 semester 2, 54 from 2014 semester 1, and 96 from 2014 semester 2) were collected for Australian University. 4 students with no examination scores were excluded. Table 5-1 shows the frequency counts of missing responses for the 106 questionnaire items (not including the open-ended questions) for the 194 students. Students who had answered over 90% of the questionnaire items (i.e., with 9 or fewer missing responses) were kept in the data analysis for this study. The final sample size for Australian University was 184.

Table 6-1: Frequency counts of missing responses

Number of missing responses	Frequency (%)
0	178 (91.8)
1	5 (2.6)
9	1 (0.5)
11	1 (0.5)
28	1 (0.5)
63	2 (1.0)
73	1 (0.5)
79	2 (1.0)
89	1 (0.5)
99	2 (1.0)

(Note: N = 194)

6.3 Sample description for Indian University

Data of 94 students (36 from year 2011, 20 from year 2012, and 38 from year 2013) were provided for Indian University. Table 5-2 shows the frequency counts of missing responses for the 115 questionnaire items (not including the open-ended questions) for the 94 students. Students who had answered over 90% of the questionnaire items (i.e., with 6 or fewer missing responses) were kept in the data analysis for this study. The final sample size for Indian University was 79.

Table 6-2: Frequency counts of missing responses

Number of missing responses	Frequency (%)
0	34 (36.2)
1	20 (21.3)
2	7 (7.4)
3	7 (7.4)
4	5 (5.3)
5	5 (5.3)
6	1 (1.1)
13	3 (3.2)
16	1 (1.1)
22	1 (1.1)
54	1 (1.1)
80	1 (1.1)
94	3 (3.2)
98	1 (1.1)
113	2 (2.1)
114	2 (2.1)

(Note: N = 94)

6.3.1 Sample Size

Data for 198 students were recorded at Australian University. After the missing data analysis, the final sample size was 184. Data for 94 students were recorded at Indian University and after the missing data analysis, the final sample size was 79.

6.3.2 Distribution of Data

For Australian University students, the negative skewness (-0.81) suggested that more data points lay to the right of the mean and the positive kurtosis (0.18) suggested that the distribution was taller (more peaked) than the normal distribution. For Indian University, the negative skewness (-0.23) suggested that more data points lay to the right of the mean and the negative kurtosis (-0.92) suggested that the distribution was flatter than the normal distribution.

For both Universities, as the data were not normally distributed, Wilcoxon ranked-sum tests and Kruskal–Wallis tests were used to determine which factors were statistically significantly in terms of the three examination scores.

6.4 Findings

6.4.1 Gender Comparison

Table 5-3 represents the gender comparison of the students studying programming. The percentage of males and females studying programming at both the universities was as following: 85.3% males and 14.7% females at the Australian University and 64.6% males and 35.4% females at Indian University. Thus, it can be concluded that the distribution of students by gender was somewhat better at Indian University than Australian University and more female students opted to study programming at Indian University, India, than at Australian University, Australia.

Table 6-3: Comparison of the students studying programming by gender

Gender		
	Australian University	Indian University
Male	157 (85.3)	51 (64.6)
Female	27 (14.7)	28 (35.4)

6.4.2 Attendance in Lectures

Table 5-4 represents a comparison of attendance at lectures for Australian University and Indian University. At Australian University, 62.5% had attended equal to or more than 80% of the programming lectures and at Indian University; 68.4% of the participants had attended equal to or more than 80% of the programming lectures.

So, the number of students attending equal to or more than 80% of the lectures was greater at Indian University than at Australian University. This result shows that the students at Indian University preferred to attend lectures, although the mandatory attendance in lectures was 70%, there was no student at Indian University who did not attend a lecture, as compared with 3.3% of students at Australian University who did not attend any of the lectures. There may be various reasons for this inclination towards attending lectures at Indian University, such as interesting teaching strategies or course material. Another reason may be the fact that at Indian University, there was no provision for watching the recorded lectures online. Further study needs to be conducted to find out the exact reasons.

Table 6-4 : A comparison of attendance in lectures for Australian University and Indian University

Q16. Number of programming lectures attended		
	Australian University	Indian University

0%	6 (3.3)	0
Up to 20%	17 (9.2)	1 (1.3)
Up to 40%	22 (12.0)	5 (6.3)
Up to 60%	24 (13.0)	19 (24.1)
Up to 80%	66 (35.9)	48 (60.8)
100%	49 (26.6)	6 (7.6)

6.4.3 Frequency of taking the course:

Table 5-5 represents the frequency of taking the programming course by the students at Australian University. At Australian University, 88% of the students took the programming course for the first time. 10.9% of students for the second time and 1.1% of students took it for the third time. These may be the students who could not pass the course first or second time or who dropped out due to some other reason. At Indian University, this question wasn't asked, as the criteria for retaking the examination were different. At Australian University, if a student fails the course, the student must take the course again, do the laboratory' work again and then re-take the examination, whereas at Indian University, if a student fails the examination, the student re-takes the examination during the next semester. The student may not attend the lectures and perform the laboratory work again.

Table 6-5: Frequency of studying the topic for Australian University

Q26. Are you studying this topic for the first time?	
Frequency	Number(Percentage)
1st time	161 (88.0)
2nd time	20 (10.9)
3rd time	2 (1.1)

6.4.4 Experience with programming

The Table 5-6 represents a comparison of students' experience with programming for both Universities. The analysis of the data collected depicts that in general a larger percentage of students had studied programming before attending the course at Indian University, India, than at Australian University, Australia.

"In 9th or 10th grade", 13% of the students had studying programming at Australian University 15.2% of students did the same at Indian University. In "11th or 12th" grade. 20.1% had studied programming at Australian University whereas 22.8% had studied programming at Indian University. "At home/self-study" 26.6% had studied programming at Australian University and 30.4% did the same at Indian University. "At University as a part of a different degree", 8.7% of students had studied programming at Australian University and 16.5% students did the same at Indian University. "At University as part of the same degree they were studying", 17.9% of students had studied programming at Australian University whereas 67.1% of students did the same at Indian University.

Table 6-6: A comparison of Experience with programming for both Universities

Q5. Did you study programming before attending this course?				
	Australian University		**Indian University**	
	Yes	No	Yes	No
9th or 10th grade	24 (13.0)	160 (87.0)	12 (15.2)	67 (84.8)
11th and 12th grade	37 (20.1)	147 (79.9)	18 (22.8)	61 (77.2)
At home/self-study	49 (26.6)	135 (73.4)	24 (30.4)	55 (69.6)
At university as part of a different degree	16 (8.7)	168 (91.3)	13 (16.5)	66 (83.5)
At university as part of the same degree you are currently studying	33 (17.9)	151 (82.1)	53 (67.1)	26 (32.9)

6.4.5 Prior Programming Language studied

The Table 5-7 represents a comparison of the Prior Programming Language studied factor for both universities. At Australian University, the most common prior programming language learnt was JAVA (36.4%), followed by Visual Basic (17.9%) and C++ (12.5%), whereas at Indian University, C appeared to the most common programming language (83.5%), and followed by C++ (55.7%) and Java (54.4%), before students were enrolled in the university. Thus, students at Australian University had learnt an object-oriented language prior to studying programming, whereas students at Indian University had studied a procedural language before studying programming at University.

Table 6-7: A comparison of Prior Programming Language studied for Australian University and Indian University

Q6. Have you studied any of the below mentioned languages?				
	Australian University		Indian University	
	Yes	No	Yes	No
C	11 (6.0)	173 (94.0)	66 (83.5)	13 (16.5)
C++	23 (12.5)	161 (87.5)	44 (55.7)	35 (44.3)
Visual Basic	33 (17.9)	151 (82.1)	7 (8.9)	72 (91.1)
Java	67 (36.4)	117 .(63.6)	43 (54.4)	36 (45.6)
PHP	15 (8.2)	169 (91.8)	28 (35.4)	51 (64.6)
Python	17 (9.2)	167 (90.8)	12 (15.2)	67 (84.8)
Basic	17 (9.2)	167 (90.8)	7 (8.9)	72 (91.1)
COBOL	2 (1.1)	182 (98.9)	0	79 (100.00)
VC++	1 (0.5)	183 (99.5)	0	79 (100.00)
PASCAL	6 (3.3)	178 (96.7)	1 (1.3)	78 (98.7)

6.4.6 Knowledge of Flowcharts and Algorithms

The Table 5-8 represents a comparison of Knowledge of Flowcharts and Algorithms for both universities. At Australian University around 1/3 of the participants had studied flowcharts (33.7%) and algorithms (27.7%), whereas at Indian University just under half of the participants had studied flowcharts (45.6%) and over 70% of the participants had studied algorithms (73.4%). In comparison, a greater percentage of students had studied algorithms and flowcharts at Indian University than at Australian University. This may be due to the inclusion of the study of flowcharts and algorithms at school level, or individual interest in understanding flowcharts and algorithms. To study the underlying reasons for this difference, further study may be conducted if the knowledge of flowcharts and algorithms proves to have a positive impact on student performance in terms of scores.

Table 6-8: A comparison of Knowledge of Flowcharts and Algorithms for Australian University and Indian University

Q7. Have you studied Flowcharts/Algorithms?				
	Australian University		Indian University	
	Yes	No	Yes	No
Flowcharts	62 (33.7)	122 (66.3)	36 (45.6)	43 (54.4)
Algorithms	51 (27.7)	133 (72.3)	58 (73.4)	21 (26.6)

6.4.7 Material studied before attending laboratory sessions

Table 5-9 represents a comparison of the material studied before attending laboratory sessions for both the universities. It appears that the students at Australian University liked to:

1) Study lecture slides related to the laboratory work (62.5%),

2) Read previous laboratory work (47.3%), and

3) Study textbook slides related to the laboratory work (46.7%) before going to the laboratory.

Whereas the students at Indian University students liked to:

1) Read previous laboratory work (63.6%),

2) Study textbook chapter related to the laboratory (50.6%), and

3) Read new programs related to previous laboratory work (50.6%) before going to the laboratory.

This suggests that students at both the Universities preferred to read previous laboratory work and study textbook slides/chapters related to laboratory work prior to attendance at laboratories. The students at Australian University also studied lecture slides related to laboratory work, which suggests that the students found those lecture slides useful. The students at Indian University seemed to explore the topic further, as they chose to read new programs related to previous laboratory work. These findings suggest that students at both the Universities found previous laboratory work and textbook slides/chapter useful. It may be beneficial for the lecturers to know if the students find textbooks useful as a resource and hence, may be able to upload the lecture slides and text book material on the course management system in advance. If the lecturer does not wish to upload the slides on the course management system or the facility to do so is not available, the students may be given printed material to study.

Table 6-9: A comparison of Material studied before attending laboratory sessions for both universities

Q11. What do you study before going to the laboratory?				
	Australian University		Indian University	
	Yes	No	Yes	No
Study lecture slides related to the laboratory	115 (62.5)	69 (37.5)	29 (36.7)	50 (63.3)
Study textbook slides related to the laboratory	86 (46.7)	98 (53.3)	40 (50.6)	39 (49.4)
Read paper-based textbook	77 (41.8)	107 (58.2)	24 (30.4)	55 (69.6)
Do online tutorials	74 (40.2)	110 (59.8)	50 (63.3)	29 (36.7)
Read previous laboratory work	87 (47.3)	97 (52.7)	38 (48.1)	41 (51.9)
Practice previous laboratory work	54 (29.3)	130 (70.7)	40 (50.6)	39 (49.4)
Read new programs related to previous laboratory work	46 (25.0)	138 (75.0)	38 (48.1)	41 (51.9)
Practice new programs related to previous laboratory work	54 (29.3)	130 (70.7)	38 (48.1)	41 (51.9)

Read new similar programs related to the laboratory	65 (35.3)	119 (64.7)	24 (30.4)	55 (69.6)
Practice new similar programs related to the laboratory	61 (33.2)	123 (66.8)	28 (35.4)	51 (64.6)

6.4.8 Preference of doing laboratory work

Table 5-10 presents a comparison of the preference for doing laboratory work for both universities. At Australian University, students liked to do laboratory work in the laboratory (78.8%) or at home (68.5%), but not in the library (25.0%) At Indian University, students liked to do laboratory work in the laboratory (75.9%) or at home (75.9%), but not in the library (20.3%). So, at both Universities the students preferred to do the laboratory work in the laboratory. The results are similar to the results obtained by a previously conducted study, which suggested that laboratory classes were considered the most important study activity; ranked first by 31% of participants as > 75% of students at both the Universities preferred to do the laboratory work in the laboratory (Butler and Morgan, 2007).

Table 6-10: A comparison of the preference for doing laboratory work for both universities

Q19. Preference of doing laboratory work				
	Australian University		Indian University	
	Yes	No	Yes	No
In the laboratory	145 (78.8)	39 (21.2)	60 (75.9)	19 (24.1)
At home	126 (68.5)	58 (31.5)	60 (75.9)	19 (24.1)
In the library	46 (25.0)	138 (75.0)	16 (20.3)	63 (79.7)

6.4.9 Family history of attending the University

Table 5-11 presents a comparison of the family history of attending University for both universities. At Australian University, 38.3% of the students were the first one in their family to attend a university and at Indian University, 34.2% of the students were the first one in the family to attend a university. For both the Universities, the percentage of students whose parents/carers attended University was approximately the same i.e. 50.3% of parents/carers of Australian University students had attended University and 50.6% of parents/carers of

Indian University students had attended University. This indicates that the children of the parents/carers who attended university are more likely to attend university. A greater percentage of siblings (50.6%) of students at Indian University attended University, as compared with Australian University students (44.8%).

Table 6-11: A comparison of family history of attending University for both Universities

Q22. First one attending University or other members in the family have attended University?				
	Australian University		Indian University	
	Yes	No	Yes	No
First one	70 (38.3)	113 (61.7)	27 (34.2)	52 (65.8)
Siblings	82 (44.8)	101 (55.2)	40 (50.6)	39 (49.4)
Parents/Carers	92 (50.3)	91 (49.5)	40 (50.6)	39 (49.4)

6.4.10 Is your home environment conducive to study?

Table 5-12 presents a comparison of whether or not the students' home environment is conducive to study for both universities. At Australian University, the majority of the students believed that their home environment was conducive to study (80.3%) and similarly at Indian University, the majority of the students believed that their home environment was conducive to study (89.9%). A slightly higher percentage of students at Indian University believed that their home environment was conducive to study.

Table 6-12: A comparison of whether or not students' home environment is conducive to study for Australian University and Indian University

Q23. Is your home environment conducive to study?			
Australian University		Indian University	
Yes	No	Yes	No
147 (80.3)	36 (19.7)	71 (89.9)	8 (10.1)

6.4.11 Availability of programming related help at home

Table 5-13 presents a comparison of the availability of programming related help is available at home for both universities**.** At Australian University, students did not get programming related help at home from their parents/grandparents/carers or siblings (86.9%) and similarly at Indian University students did not get programming related help at home from their parents/grandparents/carers or siblings (77.2%). A slightly greater percentage of students from Indian University could get programming related help at home, as compared with students at Australian University.

Table 6-13: A comparison of the availability of programming related help at home for both Universities

Q24. Can you get programming related help at home from your parents/carers or siblings?			
Australian University		Indian University	
Yes	No	Yes	No
24 (13.1)	159 (86.4)	18 (22.8)	61 (77.2)

6.4.12 Parents/Carers supportive of educational goals

Table 5-14 presents a comparison of those whose parents/carers are supportive of educational goals for both universities**.** At Australian University, it appeared that the parents/carers were supportive of the students' educational goals (90.2%) and similarly at Indian University it appeared that the parents/carers were supportive of the students' educational goals (97.5%). A slightly greater percentage of students from Indian University had parents/carers supportive of their educational goals, as compared with students at Australian University but since, at both universities, the percentage was higher than 90%, it may be concluded that both sets of parents/carers were supportive of the educational goals of the students.

Table 6-14: A comparison of the number of parents/carers supportive of students' educational goals for both universities

Q25. Are your parents/carers supportive of your educational goals?	
Australian University	Indian University

Yes	No	Yes	No
165 (90.2)	18 (9.8)	77 (97.5)	2 (2.5)

6.4.13 Studying a programming related topic again

Table 5-15 presents a comparison of whether or not students will choose to study a programming related topic again for both Universities. At Australian University, 55.2% the students would study a topic/subject related to programming again and at Indian University, 92.4% would study a topic/subject related to programming again. The percentage of students who would choose to study a topic related to programming again was higher at Indian University than at Australian University. To understand the reasons for this difference, further research needs to be conducted.

Table 6-15: A comparison of whether or not students would choose to study a programming related topic again for both Universities

Q27. If given the option, would you choose to study a topic/subject related to programming again?			
Australian University		Indian University	
Yes	No	Yes	No
101 (55.2)	82 (44.8)	73 (92.4)	6 (7.6)

6.4.14 Reasons of studying programming

Table 5-16(a) summarizes the reasons for studying programming at Australian University. For students at Australian University “Mandatory in the degree” and “Interested to know about programming” seemed to be the most important reasons for the participants to study programming.

Table 5-16(b) summarizes the reasons for studying programming at Indian University. For students at Indian University, “Interested to know about programming” and “High paying work in the industry” seemed to be the most important reasons for the participants to study programming.

So "Interested to know about programming" was found to be the common reason for students to study programming at both Universities.

Table 6-16(a): Reasons for studying programming at Australian University

Q8. Why did you choose to study programming?				
	Frequency (%) of responses			
	1	2	3	Mean (SD)
Interested to know about programming	45 (24.5)	74 (40.2)	65 (35.3)	2.11 (0.77)
It's up-coming in the job market	86 (46.7)	63 (34.2)	35 (19.0)	1.72 (0.76)
High paying work in the industry	105 (57.1)	61 (33.2)	18 (9.8)	1.53 (0.67)
Mandatory in the degree	31 (16.8)	34 (18.5)	119 (64.7)	2.48 (0.77)

(Note: N = 184. 1 = not important, 2 = somewhat important, 3 = most important. SD = standard deviation)

Table 6-16(b): Reasons for studying programming at Indian University

Q8. Why did you choose to study programming?				
	Frequency (%) of responses			
	1	2	3	Mean (SD)
Interested to know about programming	5 (6.3)	33 (41.8)	41 (51.9)	2.46 (0.62)
It's up-coming in the work market	11 (13.9)	33 (41.8)	35 (44.3)	2.30 (0.70)
High paying work in the industry	8 (10.1)	33 (41.8)	38 (48.1)	2.38 (0.67)
Mandatory in the degree	17 (21.5)	27 (34.2)	35 (44.3)	2.23 (0.78)

(Note: N = 79. 1 = not important, 2 = somewhat important, 3 = most important. SD = standard deviation)

6.4.15 Posting Questions Online

Table 5-17(a) summarizes the participants' sources of help at Australian University. Students thought that posting questions related to the topic/course/subject online on social networking or other websites, such as Google, Twitter, Facebook, and email was very helpful (mean responses > 3).

Table 5-17(b) summarizes the participants' sources of help and activities in lectures at Indian University. Students thought that posting questions related to the topic/course/subject online on social networking or other websites, such as Google, Twitter, and Facebook, and email was very helpful (mean responses > 2.5).

The analysis shows that students at both Universities found that posting questions related to the topic/course/subject online on social networking or other websites, such as Google, Twitter, Facebook, and email was very helpful. The mean score for both the Universities suggests that Twitter was the most preferred platform to post questions online, followed by email, Facebook and Google. The order of preference of students was uniform for both Universities. Based on these results, social media such as twitter or Facebook may potentially be used to provide additional support to students while learning programming.

Table 6-17(a): Posting questions online by students at Australian University

Q9. If you posted questions related to the topic/course/subject online on social networking or other websites to obtain help, was the help useful or not?					
	Frequency (%) of responses				
	1	2	3	4	Mean (SD)
Google	3 (1.6)	44 (23.9)	58 (31.5)	79 (42.9)	3.16 (0.84)
Twitter	31 (16.8)	3 (1.6)	1 (0.5)	149 (81.0)	3.46 (1.14)
Facebook	27 (14.7)	19 (10.3)	6 (3.3)	132 (71.7)	3.32 (1.15)
Email	21 (11.4)	13 (7.1)	19 (10.3)	131 (71.2)	3.41 (1.04)

Table 6-17(b): Posting questions online by students at Indian University

Q9. If you posted questions related to the topic/course/subject online on social networking or other websites to obtain help, was the help useful or not?					
	Frequency (%) of responses				
	1	2	3	4	Mean (SD)
Google	2 (2.5)	16 (20.3)	44 (55.7)	17 (21.5)	2.96 (0.72)
Twitter	8 (10.1)	3 (3.8)	2 (2.5)	66 (83.5)	3.59 (0.97)
Facebook	8 (10.1)	24 (30.4)	6 (7.6)	41 (51.9)	3.01 (1.12)
Email	4 (5.1)	21 (26.6)	9 (11.4)	45 (57.0)	3.20 (1.00)

6.4.16 Sources of help sought by students

Table 5-18(a) summarizes the sources of help sought at Australian University. In general, students at Australian University thought that lecture notes/slides, textbooks, classmates, lecturers, senior students who passed the topic (mean responses > 2), in that order, were more helpful than discussion forums, Facebook/Twitter, other social websites, private tuition outside university, and help sessions at university (mean responses < 2).

Table 5-18(b) summarizes the sources of help sought at Indian University. At Indian University, lecture notes/slides, classmates, textbooks, lecturers, senior students who passed the topic (mean responses > 3), in that order, seemed to be the most popular sources for help, as compared with discussion forums, Facebook/Twitter, other social websites, private tuition outside university, and help sessions at university (mean responses < 2).

Interestingly, at both the universities, students found lecture notes/slides, textbook/classmates, lecturers and senior students who passed the topic to be the most popular source for help. This suggests the importance of lecture notes/slides, classmates' help, textbooks, lecturers' help and senior students' help, as they were regarded as the preferred sources of help by students at both Universities.

Table 6-18(a): Sources of help sought by students at Australian University

Q15. Which sources of help were most useful?					
	Frequency (%) of responses				
	1	2	3	4	Mean (SD)
Classmates	37 (20.1)	52 (28.3)	62 (33.7)	33 (17.9)	2.49 (1.01)
Senior students who had passed the topic	91 (49.5)	25 (13.6)	44 (23.9)	24 (13.0)	2.01 (1.12)
Lecturers	69 (37.5)	30 (16.3)	47 (25.5)	38 (20.7)	2.29 (1.17)
Textbooks	26 (14.1)	52 (28.3)	62 (33.7)	44 (23.9)	2.67 (0.99)
Lecture notes/slides	20 (10.9)	54 (29.3)	75 (40.8)	35 (19.0)	2.68 (0.91)
Discussion forums	128 (69.6)	37 (20.1)	14 (7.6)	5 (2.7)	1.43 (0.75)
Facebook/Twitter	146 (79.3)	25 (13.6)	13 (7.1)	0	1.28 (0.59)
Other social websites	155 (84.2)	16 (8.7)	10 (5.4)	3 (1.6)	1.24 (0.63)
Opt for private tuition outside university	156 (84.8)	13 (7.1)	12 (6.5)	3 (1.6)	1.25 (0.65)
Help sessions at university	141 (76.6)	24 (13.0)	15 (8.2)	4 (2.2)	1.36 (0.73)

(Note: N = 184. For Q15, 1 = not useful, 2 = useful sometimes, 3 = useful most of the time, and 4 = always useful)

Table 6-18(b): Sources of help sought by students at Indian University

Q15. Which sources of help were most useful?						
	Frequency (%) of responses					
	1	2	3	4	5	Mean (SD)
Lecturers	11 (13.9)	25 (31.6)	24 (30.4)	16 (20.3)	3 (3.8)	2.68 (1.07)
Classmates	4 (5.1)	7 (8.9)	27 (34.2)	28 (35.4)	13 (16.5)	3.49 (1.04)
Senior students who had passed the topic	15 (19.0)	34 (43.0)	16 (20.3)	11 (13.9)	3 (3.8)	2.41 (1.07)
Textbooks	3 (3.8)	10 (12.7)	28 (35.4)	24 (30.4)	14 (17.7)	3.46 (1.05)
Lecture notes/slides	5 (6.3)	11 (13.9)	25 (31.6)	15 (19.0)	23 (29.1)	3.51 (1.23)
Facebook/twitter	48 (60.8)	14 (17.7)	15 (19.0)	2 (2.5)	0	1.63 (0.88)

Other social websites	43 (54.4)	17 (21.5)	15 (19.0)	2 (2.5)	2 (2.5)	1.77 (1.01)
Opt for private tuition outside university	63 (79.7)	4 (5.1)	8 (10.1)	3 (3.8)	1 (1.3)	1.42 (0.91)
Opt to study the topics at training institutes teaching similar courses	55 (69.6)	8 (10.1)	9 (11.4)	4 (5.1)	3 (3.8)	1.63 (1.11)
Watch related content on YouTube	20 (25.3)	19 (24.1)	18 (22.8)	17 (21.5)	5 (6.3)	2.59 (1.26)

(Note: N = 79. For Q14, 1 = never, 2 = sometimes, 3 = often, 4 = very often, and 5 = always. SD = standard deviation)

6.4.17 Activities in lecture theatres

Table 5-19(a) summarizes the activities in lecture theatres at Australian University and Table 5-19(b) summarizes the activities in lecture theatres at Indian University. The data analysis suggests that students at Australian University preferred to listen to the lecture (M = 3.04) or listen and make notes (M = 2.47) in the programming lecture theatre and similarly at Indian University students preferred to listen to the lecture (M = 2.89) or listen and make notes (M = 2.81) in the programming lecture theatre. Thus, there was uniformity in the results in the activities in the lecture theatre for both Universities. This suggests that students give importance to the lecture delivered and they also actively participate by taking notes, which is indicative of student engagement in learning.

Another noteworthy observation was that at both Universities, a similar percentage of students performed activities that were not related to the lecture. “Playing games on mobile phones/laptop” had a Mean (SD) score of 1.47(0.67) for Australian University and 1.54(0.71) for Indian University. Similarly, “Use social media to socialize” had a Mean(SD) score of 1.53(0.69) for Australian University and 1.48(0.68) for Indian University and “Browse internet in general” had a Mean(SD) score of 1.69(0.70) for Australian University and 1.69(0.89) for Indian University. This seems to be an interesting observation, as in both universities a similar number of students engaged in activities that were not related to learning programming. More interestingly, this was reported even though the students at Indian University were not allowed to use mobile devices in lectures, which suggests that they used their devices without the knowledge of the lecturer. There may be some reasons as to why the students engaged in these activities instead of concentrating on the lecture or performing activities related to programming. Further research needs to be conducted to investigate the reasons and re-engage the students in learning programming rather than wasting their precious time in engaging in other activities.

This also uncovered another interesting observation that at Australian University attendance is not compulsory, whereas at Indian University it is, which suggests that at Australian University the students tried to learn programming by being present in the lectures but still they performed other activities which suggests that they could not engage in learning despite their attendance. This, then, suggests that programming needs to be taught in a manner that engages students throughout the lecture or at least in a large part of the lecture.

Table 6-19(a): Activities in lecture theatres at Australian University

Q18. What do you do in the programming lecture theatre?					
	Frequency (%) of responses				
	1	2	3	4	Mean (SD)

Listen to the lecture	3 (1.6)	39 (21.2)	90 (48.9)	52 (28.3)	3.04 (0.75)
Listen and make notes	33 (17.9)	64 (34.8)	54 (29.3)	33 (17.9)	2.47 (0.99)
Annotate if you have printed notes	114 (62.0)	41 (22.3)	17 (9.2)	12 (6.5)	1.60 (0.91)
Play games on mobile phone/laptop	112 (60.9)	62 (33.7)	6 (3.3)	4 (2.2)	1.47 (0.67)
Look up technical terms discussed in the lecture	85 (46.2)	78 (42.4)	16 (8.7)	5 (2.7)	1.68 (0.75)
Use social media to socialize	103 (56.0)	68 (37.0)	9 (4.9)	4 (2.2)	1.53 (0.69)
Browse the internet in general	79 (42.9)	86 (46.7)	16 (8.7)	3 (1.6)	1.69 (0.70)

(Note: N = 184, For Q18, 1 = never, 2 = sometimes, 3 = large part of lecture, 4 = whole lecture. SD = standard deviation)

Table 6-19(b): Activities in lecture theatres at Indian University

Q18. What do you do in the programming lecture theatre?					
	Frequency (%) of responses				
	1	2	3	4	Mean (SD)
Listen to the lectures	3 (3.8)	13 (16.5)	53 (67.1)	10 (12.7)	2.89 (0.66)
Listen and make notes	2 (2.5)	22 (27.8)	44 (55.7)	11 (13.9)	2.81 (0.70)
Annotate if you have printed notes	15 (19.0)	47 (59.5)	13 (16.5)	4 (5.1)	2.08 (0.75)
Play games on mobile phone/laptop	45 (57.0)	26 (32.9)	7 (8.9)	1 (1.3)	1.54 (0.71)
Look up technical terms discussed in the lecture	14 (17.7)	43 (54.4)	18 (22.8)	4 (5.1)	2.15 (0.77)
Use social media to socialize	48 (60.8)	25 (31.6)	5 (6.3)	1 (1.3)	1.48 (0.68)

Browse the internet in general	36 (45.6)	31 (39.2)	6 (7.6)	6 (7.6)	1.77 (0.89)

(Note: N = 79. For Q19, 1 = never, 2 = sometimes, 3 = large part of lecture, 4 = whole lecture. SD = standard deviation)

6.4.18 Preliminary preparation before lectures

Table 5-20 (a) summarises "Preliminary preparation before lecture" at Australian University and Table 5-20 (b) summarises "Preliminary preparation before lecture" at Indian University. At Australian University, students preferred to study lecture slides related to the current lecture (M = 2.47) or lecture slides from previous lectures (M = 2.45) before going to the programming lecture. At Indian University, students preferred to watch content related to lectures on YouTube (M = 2.67) before going to the programming lecture. This suggests that students at Australian University considered lecture slides as important, be they the lecture slides related to the current lecture or lecture slides related to the previous lecture, whereas at Indian University, students preferred online modes of preliminary preparation. The mean(SD) number of students who opted for "online tutorials" was similar for both Universities, 2.18(1.28) for Australian University and 2.24(1.25) for Indian University.

Table 6-20(a): Preliminary preparation before lectures at Australian University

Q10. Frequency of studying before going to the programming lecture						
	Frequency (%) of responses					
	1	2	3	4	5	Mean (SD)
Lecture slides related to current lecture	45 (24.5)	58 (31.5)	46 (25.0)	19 (10.3)	16 (8.7)	2.47 (1.21)
Textbook slides related to current lecture	59 (32.1)	57 (31.0)	40 (21.7)	21 (11.4)	7 (3.8)	2.24 (1.13)
Lecture slides from previous lectures	42 (22.8)	68 (37.0)	34 (18.5)	30 (16.3)	10 (5.4)	2.45 (1.17)

Textbook slides from previous lectures	62 (33.7)	66 (35.9)	31 (16.8)	19 (10.3)	6 (3.3)	2.14 (1.10)
Read paper-based textbook	68 (37.0)	48 (26.1)	26 (14.1)	22 (12.0)	20 (10.9)	2.34 (1.37)
Online tutorials	75 (40.8)	46 (25.0)	31 (16.8)	18 (9.8)	14 (7.6)	2.18 (1.28)

Table 6-20(b): Preliminary preparation before lectures at Indian University

Q10. Frequency of studying before going to the programming lecture						
	Frequency (%) of responses					
	1	2	3	4	5	Mean (SD)
Textbook chapter related to current lecture	19 (24.1)	40 (50.6)	14 (17.7)	4 (7.6)	0	2.09 (0.85)
Lecture slides from previous lectures	20 (25.3)	28 (35.4)	20 (25.3)	9 (11.4)	2 (2.5)	2.30 (1.05)
Textbook chapter related to the previous lectures	16 (20.3)	30 (38.0)	22 (27.8)	9 (11.4)	2 (2.5)	2.38 (1.02)
Online tutorials	31 (39.2)	16 (20.3)	19 (24.1)	8 (10.1)	5 (6.3)	2.24 (1.25)
Watch content related to lecture on YouTube	18 (22.8)	20 (25.3)	16 (20.3)	18 (22.8)	7 (8.9)	2.67 (1.29)

6.4.19 Revision habits

Table 5-21(a) summarises the "Habits of revision "at Australian University and Table 5-21(b) summarises the "Habits of revision "at Indian University. At Australian University, students preferred to revise the programming topic while the semester was in progress (*M* = 2.91) or during mid-semester exams (*M* = 2.81). At Indian University, students preferred to revise the programming topic during mid-semester examinations (*M* = 3.63). Thus, at both Universities the common preference of students was to revise the topic during mid-semester examinations. This also suggests that it may help to include mid-semester examinations in the assessment structure of the topic, further motivating the students to revise.

Table 6-21(a): Habits of revision at Australian University

Q12. Habits of revising the programming topic						
	Frequency (%) of responses					
	1	2	3	4	5	Mean (SD)
During mid-semester break	33 (17.9)	71 (38.6)	42 (22.8)	26 (14.1)	12 (6.5)	2.53 (1.14)
During mid-semester examinations	36 (19.7)	34 (18.6)	60 (32.8)	34 (18.6)	19 (10.4)	2.81 (1.24)
Both during mid-semester break and mid-semester examinations	41 (22.4)	54 (29.5)	51 (27.9)	25 (13.7)	12 (6.6)	2.52 (1.17)
Revised while the semester was in progress	12 (6.5)	69 (37.5)	47 (25.5)	35 (19.0)	21 (11.4)	2.91 (1.13)

(N = 183 for Q12 (During mid-semester exams, both during mid-semester break and mid-semester exams. For, Q12 1 = never, 2 = sometimes, 3 = often, 4 = very often, and 5 = always. SD = standard deviation)

Table 6-21(b): Habits of revision at Indian University

Q12. Habits of revising the programming topic						
	Frequency (%) of responses					
	1	2	3	4	5	Mean (SD)
During mid-semester break	10 (12.7)	25 (31.6)	21 (26.6)	13 (16.5)	10 (12.7)	2.85 (1.22)
During mid-semester examinations	7 (8.9)	9 (11.4)	19 (24.1)	15 (19.0)	29 (36.7)	3.63 (1.32)
Both during mid-semester break and mid-semester examinations	7 (8.9)	21 (26.6)	29 (36.7)	14 (17.7)	8 (10.1)	2.94 (1.10)
Revised while the semester was in progress	4 (5.1)	30 (38.0)	26 (32.9)	14 (17.7)	5 (6.3)	2.82 (1.00)

(Note: N = 79. For, Q12, 1 = never, 2 = sometimes, 3 = often, 4 = very often, and 5 = always. SD = standard deviation.)

6.4.20 Kinds of revision undertaken by students

Table 5-22(a) summarises the "Habits of studying before lecture, revising programming topics, and viewing the programming lectures online, sources of help, and attitude regarding the laboratory" at Australian University. At Australian University when revising programming topics, students liked to use theory from lecture slides available on FLO, textbook slides available on FLO and laboratory work, view lectures online, revise on a website designed to revise the topic, and revise the previous week's laboratory work (mean responses > 2). Table 5-22(b) summarises the "Habits of studying before lecture, revising programming topics, and viewing the programming lectures online, sources of help, and attitude regarding the laboratory" at Indian University. At Indian University, when revising programming topics, students liked to use textbook chapters, laboratory work, and theory from lecture slides given by the lecture (mean responses > 2.7). Revision of laboratory work was a common choice for students at both Universities.

Table 6-22(a): Habits of studying before lectures, revising programming topics, and viewing the programming lectures online, sources of help, and attitudes regarding the laboratory at Australian University.

Q13. Programming topic revision						
	Frequency (%) of responses					
	1	2	3	4	5	Mean (SD)
Theory from lecture slides available on FLO	23 (12.6)	53 (29.0)	38 (20.8)	39 (21.3)	30 (16.4)	3.00 (1.29)
Textbook slides available on FLO	39 (21.2)	64 (34.8)	28 (15.2)	26 (14.1)	27 (14.7)	2.66 (1.35)

Laboratory work	27 (14.7)	64 (34.8)	54 (29.3)	22 (12.0)	17 (9.2)	2.66 (1.15)
View lectures online	40 (21.7)	57 (31.0)	41 (22.3)	28 (15.2)	18 (9.8)	2.60 (1.25)
Revised on a website designed to revise the topic	84 (45.7)	44 (23.9)	27 (14.7)	23 (12.5)	6 (3.3)	2.04 (1.18)
Revised previous week's laboratory work	48 (26.1)	76 (41.3)	40 (21.7)	13 (7.1)	7 (3.8)	2.21 (1.03)
Revised new similar programs	82 (44.6)	54 (29.3)	35 (19.0)	7 (3.8)	6 (3.3)	1.92 (1.04)
Read previously done laboratory work	47 (25.5)	73 (39.7)	39 (21.2)	18 (9.8)	7 (3.8)	2.27 (1.07)
Redo previously done laboratory work	95 (51.9)	53 (29.0)	23 (12.6)	9 (4.9)	3 (1.6)	1.75 (0.97)
Read new similar programs	86 (45.1)	52 (28.3)	35 (19.0)	9 (4.9)	5 (2.7)	1.92 (1.04)
Redo new similar programs	96 (52.5)	49 (26.8)	26 (14.2)	7 (3.8)	5 (2.7)	1.78 (1.01)

(N = 183. Q13 (Theory from lecture slides available on FLO, Redo previously done laboratory work, Redo new similar programs). For Q13 1 = never, 2 = sometimes, 3 = often, 4 = very often, and 5 = always. SD = standard deviation)

Table 6-22(b): Habits of studying before lectures, revising programming topics, and viewing the programming lectures online, sources of help, concentration status during lecture, attitude regarding the laboratory, attitude regarding future careers at Indian University.

Q13. Programming topic revision						
	Frequency (%) of responses					
	1	2	3	4	5	Mean (SD)

Theory from lecture slides given by the lecturer	14 (17.7)	21 (26.6)	25 (31.6)	10 (12.7)	9 (11.4)	2.73 (1.23)
Textbook chapters	9 (11.5)	24 (30.8)	21 (26.9)	14 (17.9)	10 (12.8)	2.90 (1.21)
Laboratory work	8 (10.1)	30 (38.0)	19 (24.1)	16 (20.3)	6 (7.6)	2.77 (1.12)
Watch subject related content on YouTube	16 (20.3)	25 (31.6)	16 (20.3)	17 (21.5)	5 (6.3)	2.62 (1.21)
Revised on a website designed to revise the topic	22 (28.2)	22 (28.2)	18 (23.1)	13 (16.7)	3 (3.8)	2.40 (1.18)
Revised previous week's laboratory work	14 (17.7)	37 (46.8)	15 (19.0)	9 (11.4)	4 (5.1)	2.39 (1.07)
Revised new similar programs	11 (14.1)	37 (47.4)	20 (25.6)	7 (9.0)	3 (3.8)	2.41 (0.97)
Read previously done laboratory work	10 (12.8)	30 (38.5)	23 (29.5)	9 (11.5)	6 (7.7)	2.63 (1.09)
Redo previously done laboratory work	19 (24.4)	31 (39.7)	18 (23.1)	7 (9.0)	3 (3.8)	2.28 (1.06)
Read new similar programs	9 (11.4)	32 (40.5)	23 (29.1)	10 (12.7)	5 (6.3)	2.62 (1.05)
Redo new similar programs	11 (14.1)	33 (42.3)	22 (28.2)	7 (9.0)	5 (6.4)	2.51 (1.05)

(Note: N = 78 for Q13 (Textbook chapters, Revised on a website designed to revise the topic, Revised new similar programs, Read previously done laboratory work, Redo previously done laboratory work, Redo new similar programs. For Q13 1 = never, 2 = sometimes, 3 = often, 4 = very often, and 5 = always. SD = standard deviation.)

6.4.21 Usefulness of attending laboratory sessions

Table 5-23 summarises the "Usefulness of attending laboratory sessions" for both Australian University and Indian University.

At Australian University, attending labs was regarded as most useful by the students (*M* = 3.55) and similarly attending labs was also regarded as most useful by the students (*M* = 3.14) at Indian University. This suggests that students at both universities gave priority to laboratory work. Thus, it must be ensured that adequate support is provided to the students in the laboratory.

Table 6-23: Usefulness of attending laboratory sessions

Q20. Do you find attending labs useful?						
	1	2	3	4	5	p
Australian University	6 (3.3)	24 (13.0)	57 (31.0)	56 (30.4)	41 (22.3)	3.55 (1.07)
Indian University	5 (6.3)	17 (21.5)	30 (38.0)	16 (20.3)	11 (13.9)	3.14 (1.11)

For Q20, 1 = not at all, 2 = slightly useful, 3 = useful, 4 = very useful, and 5 = extremely useful. SD = standard deviation.

6.5 Analysis of the students' performance in terms of scores based on the various factors studied

6.5.1 Gender: Hypothesis 1:

The Table 5-24 summarises the "Mean (SD) examination scores by gender" for the two universities. From the analysis of data of Australian University, the results of the Wilcoxon ranked-sum test suggested that there were no statistically significant differences in examination scores between male and female students (p = 0.828) and from the data analysis of Indian University, the analysis results of Wilcoxon ranked–sum tests suggested that female students had statistically significantly higher examination scores than male students (M = 65.29, SD = 12.30 for female; M = 55.29, SD = 14.82 for male; p = 0.003).

Table 6-24: Mean (SD) examination scores by gender *(Note: SD = standard deviation. P = p-value of the Wilcoxon ranked-sum tests for gender)*

Gender				
	Australian University		Indian University	
	Mean (SD)	p	Mean (SD)	p
Male	63.17 (22.62)	0.828	55.29 (14.82)	0.003*
Female	64.44 (20.71)		65.29 (12.30)	

This result is consistent with the dominant pattern of male vs female performance in academia in general in India(Sharma, 2016). A study concluded that "In those more competitive academic sectors with entrance quotas, being female (gender) increases the probability of persisting at the university" (Montmarquette et al., 2001). Another study conducted by Arulampalam concluded that "There are significant differences by gender, with males more likely to drop out" (Arulampalam et al., 2004). For the past few years, the females have been performing better than males academically in India. The person who came first in the civil services examination in India for year 2016 was a female. The class 12 results witnessed a similar trend, with girls scoring better than boys: "Girls have performed better than boys this time as well, with 88.58 per cent girls clearing the exams compared to 78.85 per cent of boys" (Sharma, 2016). Another similar study conducted by Wilson suggested that females reported having more encouragement to study computer science than the males in the sample(Wilson, 2002). This result suggests that learning culture may affect performance based on gender, with females tending to perform better than boys in an eastern learning culture. The cause of this variation needs further investigation.

Thus, hypothesis H_0 was accepted for Australian University and hypothesis H_1 was accepted for Indian University.

6.5.2 Prior Programming experience: Hypothesis 2

Table 5-25 shows the mean examination scores by experience with programming. For Australian University, the results of the Wilcoxon ranked-sum tests suggested that students who had studied programming before attending the course at home (p = 0.012), would have statistically significantly higher examination scores than students who had studied programming at other levels. There was no statistically significant difference in examination scores across other categories of experience with programming. This suggests that the students who study programming at home for interest before attending the University perform better.

The results obtained from the analysis of Australian University data were consistent with the results obtained by Hagan and Markham ,who analysed the effect of prior programming experience and the number of programming languages learnt and concluded that students who had experience in at least one programming language at the beginning of an introductory programming course performed significantly better in the assessment than those with none(Hagan and Markham, 2000).

For Indian University, there was no statistically significant difference in examination scores across other categories of experience with programming.

Further study needs to be conducted to investigate the reasons for this inconsistency wherein the effect of prior-programming language on procedural programming language may be studied in detail. The results obtained at Indian University were consistent with the results obtained by de Raadt who concluded from their study that "while previous programming experience contributing to better scores is logical this was not found to happen in all cases and somewhat surprisingly not for all programming languages" (de Raadt et al., 2005).

So, hypothesis H_1 was accepted for Australian University and hypothesis H_0 was accepted for Indian University.

Table 6-25: Mean (SD) examination scores by experience with programming

Q5. Did you study programming before attending this course at						
	Australian University			Indian University		
	Yes	No	p	Yes	No	p

9th or 10th grade	61.17 (29.48)	63.69 (21.12)	0.957	59.96 (17.10)	58.63 (14.38)	0.667
11th and 12th grade	60.89 (27.05)	63.98 (21.00)	0.926	61.89 (14.12)	57.93 (14.87)	0.332
At home/ individual study	68.10 (26.37)	61.64 (20.47)	0.012*	62.39 (15.15)	57.29 (14.38)	0.136
At university as part of a different degree	55.31 (30.44)	64.13 (21.33)	0.304	57.12 (14.12)	59.17 (14.90)	0.611
At university as part of the same degree you are currently studying	59.97 (20.81)	64.10 (22.61)	0.216	59.26 (15.37)	57.97 (13.50)	0.684

(*Note: SD = standard deviation. P = p-value of the Wilcoxon ranked-sum tests. * indicates significance at the 0.05 level.)*

6.5.3 Prior knowledge of programming language: Hypothesis 3

Table 5-26 summarises “Mean (SD) examination scores by knowledge of programming language” for both universities. There was no statistically significant difference in examination scores across the different categories of languages learnt in the past at either Australian University or Indian University.

The results of this study were inconsistent with the results obtained by Hagan and Markham, as their study concluded that the more the languages with which a student has experience, the better their performance tends to be(Hagan and Markham, 2000). The data collected from both universities had a few students who had studied one or more programming languages prior to studying the topic at university but this study could not establish a relationship between the languages already learnt and student performance for both universities. Further study needs to be conducted to find out the reason.

The H_0 hypothesis is accepted for both universities.

Table 6-26: Mean (SD) examination scores by knowledge of programming language

Q6. Have you studied any of the below mentioned languages?		
	Australian University	Indian University

	Yes	No	p	Yes	No	p
C	72.36 (19.14)	62.79 (22.42)	0.172	59.81 (13.71)	53.90 (18.88)	0.318
C++	60.83 (25.30)	63.72 (21.91)	0.630	58.91 (13.80)	58.74 (15.98)	0.902
Visual Basic	61.88 (28.51)	63.68 (20.81)	0.865	65.04 (10.83)	58.23 (14.95)	0.251
Java	66.90 (20.80)	61.33 (22.96)	0.170	60.29 (13.26)	57.10 (16.29)	0.485
PHP	56.93 (26.63)	63.93 (21.88)	0.265	60.48 (13.48)	57.93 (15.40)	0.552
Python	64.06 (27.18)	63.29 (21.84)	0.690	63.15 (12.42)	58.06 (15.03)	0.287
Basic	64.76 (26.53)	63.22 (21.92)	0.667	61.89 (10.79)	58.54 (15.06)	0.654
COBOL	32.50 (33.23)	63.70 (22.05)	0.096	NA	58.84 (14.71)	NA
VC++	66.00 (NA)	63.34 (22.36)	0.989	NA	58.84 (14.71)	NA
PASCAL	73.67 (34.33)	63.01 (21.85)	0.124	37.50 (NA)	59.11 (14.60)	0.177

(Note: SD = standard deviation. P = p-value of the Wilcoxon ranked-sum tests. * indicates significance at the 0.05 level.)

6.5.4 Effect of studying flowcharts or algorithms on the performance/scores of the students: Hypothesis 4

Table 5-27 summarises "Mean (SD) examination scores by knowledge of Flowcharts/Algorithms" for both universities. There was no statistically significant difference in examination scores whether students had or had not learnt Flowcharts or Algorithms at both the universities.

After this study it may be concluded that studying flowcharts and algorithms had no effect on student performance, or further study may be conducted with a different design model to explore the effect of prior knowledge of flowcharts and algorithms on student performance in learning programming.

Hypothesis H_0 is accepted for both Universities.

Table 6-27: Mean (SD) examination scores by knowledge of Flowcharts/Algorithms

Q7. Have you studied designing Flowcharts/Algorithms?						
	Australian University			Indian University		
	Yes	No	p	Yes	No	p
Flowcharts	63.81 (24.05)	63.13 (21.46)	0.587	59.33 (13.46)	58.42 (15.82)	0.906
Algorithms	65.82 (23.01)	62.41 (22.04)	0.275	58.43 (15.32)	59.95 (13.16)	0.694

(*Note: SD = standard deviation. P = p-value of the Wilcoxon ranked-sum tests. * indicates significance at the 0.05 level)*

6.5.5 Posting questions/asking for help on Google, Twitter, Facebook, and Email and its effect on scores: Hypothesis 5

Table 5-28(a) summarises the Mean (SD) examination scores by sources of help and activities in lecture at Australian University. The results of the Kruskal–Wallis tests were significant for “Twitter” as an option, but the results of the pairwise comparisons were not significant at the 0.05 level. Thus, these factors were concluded as not statistically significant to the examination scores. There was no statistically significant difference in examination scores across other categories of sources of help.

Table 5-28(b) summarises the mean (SD) examination scores by sources of help and activities in lectures at Indian University. For Indian University, these factors were concluded to be not statistically significant to the examination scores as there was no statistically significant difference in examination scores across any categories of sources of help.

This question was asked to determine the sources of help sought by students when required and whether the availability of online resources proves to be useful to students. The usage of Twitter showed some interesting results for students at Australian University, so it should be evaluated in further studies.

So, hypothesis H_0 is accepted for both Universities.

Table 6-28(a): Mean (SD) examination scores by sources of help and activities in lectures at Australian University

Q9. If you posted questions related to the topic/course/subject online on social networking or other websites to obtain help was the help useful or not?					
	1	2	3	4	p
Google	61.67 (22.85)	61.61 (19.17)	63.07 (21.86)	64.61 (24.47)	0.590
Twitter	56.65 (21.02)	37.67 (24.17)	71.00 (NA)	65.22 (22.16)	0.038**
Facebook	59.70 (23.06)	59.42 (16.13)	52.00 (28.76)	65.19 (22.53)	0.201
Email	61.48 (22.62)	64.31 (18.37)	61.79 (23.54)	63.79 (22.64)	0.947

(Note: For Q9, 1 = not helpful, 2 = somewhat helpful, 3 = very helpful, and 4 = did not post. SD = standard deviation. NA = not available. P = p-value of the Kruskal–Wallis tests. *indicates significance at the 0.05 level. ** indicates significant at the 0.05 level for the Kruskal–Wallis tests, but not significant at the 0.05 level for the pairwise comparisons)

Table 5-28 (b): Mean (SD) examination scores by sources of help and activities in lectures at Indian University

Q9. If you posted questions related to the topic/course/subject online on social networking or other websites to obtain help was the help useful or not?					
	1	2	3	4	p
Google	68.88 (2.65)	56.61 (14.86)	59.01 (14.73)	59.31 (15.70)	0.680
Twitter	49.16 (11.01)	61.00 (12.77)	77.00 (3.18)	59.36 (14.81)	0.066
Facebook	60.34 (10.65)	57.84 (15.99)	63.88 (15.78)	60.34 (14.24)	0.225
Email	51.81 (10.39)	57.99 (15.22)	57.50 (20.86)	60.12 (13.60)	0.656

(Note: For Q9, 1 = not helpful, 2 = somewhat helpful, 3 = very helpful, and 4 = did not post. SD = standard deviation. NA = not available. P = p-value of the Kruskal–Wallis tests. *indicates significance at the 0.05 level. ** indicates significant at the 0.05 level for the Kruskal–Wallis tests, but not significant at the 0.05 level for the pairwise comparisons)

6.5.6 Reasons to study programming and its effect on the scores of students: Hypothesis 6

Table 5-29(a) shows the mean examination scores by reasons of studying programming for Australian University students. The various parameters evaluated included “interested to know about programming”, “It is upcoming in the job market”, and “High paying work is available in the industry”, “Mandatory in the degree. The results of the Kruskal–Wallis tests suggested that there was a statistically significant difference in examination scores among students viewing the importance of “Interested to know about programming” differently (p = 0.006). The results of pairwise comparisons suggested that the students who regarded “Interested to know about programming” as not important had statistically significantly lower examination scores (M = 55.91, SD = 20.86) than the students who regarded “Interested to know about programming” as most important (M = 66.20, SD = 25.76) (p = 0.005). There was no statistically significant difference in examination scores across other categories of reasons for studying programming (p > 0.05).

Table 5-29(b) shows the mean examination scores by reasons of studying programming for Indian University students. The analysis results of the Kruskal–Wallis tests suggested that there was no statistically significant difference in examination scores across categories of reasons of studying programming (p > 0.05).

For Australian University, the results suggested that the students who studied programming as they were interested to know about programming scored better. This result indicates that students generally score better if they study programming by choice and not because for any external factor. The results of this study for Australian University were in close conjunction to the results of the study conducted by Carter which concluded that the students’ understanding of the amount of money to be made in the field was not a significant influence in the choice not to study Computer Science(Carter, 2006).

So, hypothesis H_1 is accepted for Australian University and hypothesis H_0 is accepted for Indian University.

Table 6-29(a): Mean (SD) examination scores by reasons of studying programming for Australian University.

Q8. Why did you choose to study programming?				
	1	2	3	p
Interested to know about programming	55.91 (20.86)	65.39 (18.91)	66.20 (25.76)	0.006*
It's up-coming in the job market	60.95 (21.68)	68.78 (20.08)	59.51 (26.12)	0.055
High paying work in the industry	63.30 (21.68)	63.31 (22.01)	63.83 (27.74)	0.976
Mandatory in the degree	66.97 (24.17)	59.59 (23.01)	63.50 (21.62)	0.273

(*Note: 1 = not important, 2 = somewhat important, 3 = most important. SD = standard deviation. P = p-value of the Kruskal–Wallis tests. * indicates significance at the 0.05 level)*

Table 5-29(b): Mean (SD) examination scores by reasons of studying programming for Indian University

Q8. Why did you choose to study programming?				
	1	2	3	p
Interested to know about programming	54.90 (12.10)	55.64 (16.24)	61.88 (13.29)	0.182
It's up-coming in the job market	62.09 (15.59)	60.34 (15.02)	56.39 (14.17)	0.367
High paying work in the industry	59.13 (16.75)	57.58 (15.20)	59.87 (14.17)	0.831
Mandatory in the degree	62.09 (13.35)	55.69 (16.66)	59.68 (13.66)	0.360

(*Note: 1 = not important, 2 = somewhat important, 3 = most important. SD = standard deviation. P = p-value of the Kruskal–Wallis tests. * indicates significance at the 0.05 level)*

6.5.7 Effect of Preliminary preparation in improving performance of the students: Hypothesis 7

Table 5-30 (a) summarises the mean (SD) examination scores by habits of studying before lectures for Australian University. For "Lecture slides related to current lecture", "Read paper-based textbook", and "Online tutorials", the results of the Kruskal–Wallis tests were significant, but the results of the pairwise comparisons were not significant at the 0.05 level. Thus, these factors were concluded as not statistically significant to the examination scores. There was no statistically significant difference in examination scores across other categories of habits of studying before lectures ($p > 0.05$).

The various parameters studied at Australian University included "Study lecture slides related to the current lecture available on FLO", "Study textbook slides related to the current lecture available on FLO", "Study lecture slides related to the previous lecture available on FLO", "Study textbook slides related to the previous lecture available on FLO", "Read paper-based textbook", "Do online tutorials/read about the topic to be covered online before lecture". For the students at Australian University, "Lecture slides related to current lecture", "Read paper-based textbook", and "Online tutorials", the results of the Kruskal–Wallis tests were significant, but it was concluded that the results of the pairwise comparisons were not significant at the 0.05 level. Thus, it was concluded that these factors were not statistically significant to the examination scores.

Table 5-30 (b) summarises mean (SD) examination scores by habits of studying before lectures for Indian University. The various parameters studied at Indian University included "Study textbook chapter related to the current lecture", "Study lecture slides from the previous lecture given by the lecturer", "Study textbook chapter related to the previous lecture", "Do online tutorials/read about the topic to be covered online before lecture", "Watch content related to lecture on YouTube". For Indian University, there was no statistically significant difference in examination scores across any of the categories of habits of studying before lectures.

The results achieved through this study were in contrast with the results achieved by Chen and Lin who studied the effects of downloading PowerPoint slides before lectures and concluded that downloading lecture slides before a class improved students' examination performance by 3.48 percent suggested that instructors could help students improve their academic performance by supplying PowerPoint slides(Chen and Lin, 2008). Thus, the study concluded that downloading PowerPoint slides before a lecture might enhance students' comprehension of class materials, thereby enhancing or improving their learning.

Another study conducted by Moravec et al. to study the effects of learning before lectures in Biology investigated the influence of studying before lectures on student performance(Moravec et al., 2010). The study concluded that learning undertaken before lectures combined with interactive exercises can be implemented incrementally and result in significant increases in learning gains in large introductory biology classes.

There may be a few reasons for the contrasting results obtained through this study. As learning programming is considered different from learning other topics/subjects, it may be possible that the learning before lectures may not prove to be as effective in learning programming as it may be in learning other topics like Biology, where it is more important to learn and understand facts. In learning programming, the facts learnt have to be implemented through abstract reasoning. A preliminary study conducted at Australian University showed some positive co-relations between preliminary preparation and scores obtained but the sample size was only 33. The results could not be replicated for a large sample size at both Australian University and Indian University. Further studies need to be conducted to analyse the effect of preliminary preparation on students' performance. Some other tools may be used to access the preliminary preparation done by students, such as an optional exercise based on an upcoming lecture may be given to them before the lecture. Finally, the students who complete the exercise may be compared with the students who did not complete. This may be a part of the further study to be conducted.

Some parameters proved to be somewhat effective for Australian University, such as "Lecture slides related to current lecture", "Read paper-based textbook", and "Online tutorials", thus these factors may be explored in further studies with larger sample sizes to confirm the results.

So, hypothesis H_0 is accepted for both Universities.

Table 6-30(a): Mean (SD) examination scores by habits of studying before lectures for Australian University

Q10. Frequency of studying before going to the programming lecture

	1	2	3	4	5	p
Lecture slides related to the current lecture	67.69 (20.38)	56.52 (22.78)	63.02 (22.32)	72.21 (14.50)	66.44 (28.01)	0.024**
Textbook slides related to the current lecture	62.58 (22.94)	61.33 (21.32)	64.13 (22.28)	66.33 (21.48)	73.14 (29.81)	0.325
Lecture slides from the previous lecture	65.36 (21.27)	58.50 (23.66)	63.38 (22.08)	69.63 (18.26)	69.10 (25.72)	0.178
Textbook slides from the previous lecture	62.90 (22.38)	59.58 (23.22)	70.10 (18.01)	67.84 (21.42)	60.67 (30.52)	0.320
Read paper-based textbook	56.96 (24.39)	70.19 (15.61)	60.00 (24.10)	68.41 (19.11)	67.55 (24.37)	0.025**
Online tutorials	61.57 (22.11)	59.11 (22.70)	62.87 (22.79)	75.06 (15.34)	72.93 (23.63)	0.021**

(*Note: N = 184. 1 = never, 2 = sometimes, 3 = often, 4 = very often, and 5 = always. For Q20, 1 = not at all, 2 = slightly useful, 3 = useful, 4 = very useful, and 5 = extremely useful. SD = standard deviation. P = p-value of the Kruskal–Wallis tests. * indicates significance at the 0.05 level. ** indicates significant at the 0.05 level for the Kruskal–Wallis tests, but not significant at the 0.05 level for the pairwise comparisons)*

At Indian University, there was no statistically significant difference in examination scores across any of the categories of habits of studying before lectures.

Table 5-30(b): Mean (SD) examination scores by habits of studying before lectures for Indian University

Q10. Frequency of studying before going to the programming lecture						
	1	2	3	4	5	p
Lecture slides related to the current lecture	59.66 (14.09)	58.51 (14.90)	57.93 (14.57)	60.50 (19.17)	NA	0.948
Textbook slides related to the current lecture	61.20 (15.77)	54.29 (14.55)	63.33 (12.78)	55.64 (15.07)	68.25 (12.37)	0.194
Lecture slides from the previous lecture	63.23 (15.24)	57.97 (13.97)	54.09 (13.97)	67.06 (10.06)	51.88 (35.89)	0.136
Textbook slides from the previous lecture	58.75 (14.64)	58.47 (17.21)	55.89 (14.40)	64.97 (7.77)	61.90 (18.23)	0.685
Read paper-based textbook	60.63 (14.31)	61.89 (13.59)	56.06 (15.07)	54.78 (14.59)	62.29 (18.76)	0.491
Online tutorials	59.66 (14.09)	58.51 (14.90)	57.93 (14.57)	60.50 (19.17)	NA	0.948

(Note: For and Q10, 1 = never, 2 = sometimes, 3 = often, 4 = very often, and 5 = always SD = standard deviation. SD = standard deviation. NA = not available. P = p-value of the Kruskal–Wallis tests. *indicates significance at the 0.05 level. ** indicates significant at the 0.05 level for the Kruskal–Wallis tests, but not significant at the 0.05 level for the pairwise comparisons)

6.5.8 Effect of preliminary preparation before laboratories on student performance: Hypothesis 8

Table 5-31 (a) and Table 5-31 (b) show the mean examination scores by study habits before going to the laboratory for Australian University and Indian University respectively. There was no statistically significant difference in examination scores based on study habits before going to the laboratory ($p > 0.05$) for both Australian University and Indian University.

Table 6-31(a): Mean (SD) examination scores by study habits before going to the laboratory for Australian University

Q11. What do you study before going to the laboratory?			
	Yes	No	p
Study lecture slides related to the laboratory	63.12 (21.13)	63.75 (24.29)	0.617
Study textbook slides related to the laboratory	62.69 (24.03)	63.95 (20.78)	0.920
Read paper-based textbook	66.56 (21.76)	61.06 (22.51)	0.092
Do online tutorials	65.41 (22.42)	61.98 (22.22)	0.261
Read previous laboratory work	62.85 (22.56)	63.81 (22.18)	0.784
Practice previous laboratory work	65.30 (21.42)	62.55 (22.69)	0.409

Read new programs related to previous laboratory work	62.28 (22.86)	63.72 (22.19)	0.750
Practice new programs related to previous laboratory work	68.30 (19.89)	31.31 (22.99)	0.052
Read new similar programs related to the laboratory	64.32 (22.15)	62.83 (22.46)	0.632
Practice new similar programs related to the laboratory	64.56 (23.75)	62.76 (21.63)	0.468

(Note: SD = standard deviation. P = p-value of the Wilcoxon ranked-sum tests. * indicates significance at the 0.05 level)

Table 5-31(b): Mean (SD) examination scores by study habits before going to the laboratory for Indian University

Q11. What do you study before going to the laboratory?			
	Yes	No	p
Study lecture slides related to the laboratory	55.97 (15.50)	60.50 (14.13)	0.218
Study textbook slides related to the laboratory	57.94 (15.90)	59.76 (13.53)	0.698
Read paper-based textbook	59.66 (15.98)	58.48 (14.26)	0.701

Do online tutorials	56.89 (14.81)	62.20 (14.15)	0.132
Read previous laboratory work	55.63 (15.82)	61.81 (13.09)	0.090
Practice previous laboratory work	58.40 (14.09)	59.28 (15.49)	0.784
Read new programs related to previous laboratory work	59.45 (14.51)	58.26 (15.05)	0.695
Practice new programs related to previous laboratory work	60.33 (14.89)	57.45 (14.58)	0.336
Read new similar programs related to the laboratory	56.02 (16.42)	60.06 (13.88)	0.365
Practice new similar programs related to the laboratory	55.65 (15.52)	60.59 (14.09)	0.154

(*Note: SD = standard deviation. P = p-value of the Wilcoxon ranked-sum tests. * indicates significance at the 0.05 level)*

6.5.9 Effect of revision on performance of students: Hypothesis 9

Table 5-32 (a) shows the mean examination scores by habits of revising programming topics for Australian University. The results of the Kruskal–Wallis tests suggested that there was a statistically significant difference in examination scores among students with habits of revising programming topics while the semester was in progress (p = 0.005). In particular, the results of pairwise comparisons suggested that the students consistently revising while the semester was in progress had statistically significantly higher examination scores (M = 74.76, SD = 12.85) than the students never (M = 55.50, SD = 25.84; p = 0.019) or sometimes (M = 55.50, SD = 25.84; p = 0.045) revising while the semester was in progress. There was no statistically significant difference in examination scores across other categories of habits of revising programming topics (p > 0.05).

The results obtained at Australian University were close to the expected results: the students who revise throughout the semester perform better. The results of a study conducted by Roddan concluded that statistical evidence from the post examination questionnaire demonstrated that students who state they have kept up with the course perform better in the exams, and these two findings combine to reinforce the view that students should make every effort to stay on top of the course requirements(Roddan, 2002). Ebbinghaus proved the fact that revision improves learning in general. When a new piece of information is learned, at the beginning the rate of retention is 100% but the retention drops unto 40% in the first few days(Ebbinghaus, 1985). The fact which was proved to be true in general also proved to be true for Australian University results but not for Indian University.

Table 5-32 (b) shows the mean examination scores by habits of revising programming topics for Indian University. There was no statistically significant difference in examination scores across any categories of habits of revising programming topics ($p > 0.05$).

Further research needs to be conducted to find out the reason for "revision throughout the semester" being ineffective for Indian University students and effective for Australian University students.

Hypothesis H_1 is accepted for Australian University and Hypothesis H_0 is accepted for Indian University.

Table 6-32(a): Mean (SD) examination scores by habits of revising programming topics for Australian University

Q12. Habits of revising the programming topic						
	1	2	3	4	5	p
During mid-semester break	62.12 (20.96)	60.46 (23.61)	65.50 (18.68)	70.62 (20.23)	60.67 (31.83)	0.411
During mid-semester exams	61.47 (23.64)	64.38 (22.83)	61.25 (22.22)	66.32 (20.59)	69.16 (19.91)	0.609
Both during mid-semester break and mid-semester exams	60.71 (22.43)	63.67 (22.38)	63.10 (21.68)	62.04 (22.88)	71.92 (22.97)	0.485
Revised while the semester was in progress	46.17 (29.98)	60.10 (20.34)	68.47 (20.22)	61.97 (25.78)	74.76 (12.85)	0.005*

(N = 183 for Q12 (During mid-semester exams, Both during mid-semester break and mid-semester exams). 1 = never, 2 = sometimes, 3 = often, 4 = very often, and 5 = always. For Q20, 1 = not at all, 2 = slightly useful, 3 = useful, 4 = very useful, and 5 = extremely useful. SD = standard deviation. P = p-value of the Kruskal–Wallis tests. * indicates significance at the 0.05 level)

Table 5-32 (b): Mean (SD) examination scores by habits of revising programming topics for Indian University

Q12. Habits of revising the programming topic						
	1	2	3	4	5	p
During mid-semester break	61.08 (16.05)	60.47 (15.23)	62.87 (13.81)	50.25 (14.91)	55.20 (10.34)	0.116
During mid-semester exams	54.57 (12.52)	52.67 (15.02)	54.75 (17.86)	60.27 (13.89)	63.72 (12.22)	0.120
During both mid-semester break and mid-semester exams	59.89 (19.03)	57.80 (14.77)	59.10 (15.16)	61.77 (14.93)	54.53 (14.93)	0.836
Revised while the semester was in progress	53.25 (14.78)	58.48 (15.38)	59.17 (13.92)	57.98 (16.45)	66.10 (12.33)	0.734

(*Note: For and Q12, 1 = never, 2 = sometimes, 3 = often, 4 = very often, and 5 = always SD = standard deviation. SD = standard deviation. NA = not available. P = p-value of the Kruskal–Wallis tests. *indicates significance at the 0.05 level. ** indicates significant at the 0.05 level for the Kruskal–Wallis tests, but not significant at the 0.05 level for the pairwise comparisons*)

6.5.10 Effect of the kinds of revision undertaken in terms of scores obtained by the students: Hypothesis 10

Table 5-33(a) summarises the mean (SD) examination scores for the kinds of revision done for Australian University. The results of the Kruskal–Wallis tests suggested that there was a statistically significant difference in examination scores among students with habits of revising on a website designed to revise the topic (p = 0.025). In particular, the results of pairwise comparisons suggested that students who are very often revising on a website designed to revise the topic had statistically significantly higher examination scores (M = 76.65, SD = 10.65) than students who never did so (M = 61.81, SD = 24.19; p = 0.037) or sometimes (M = 59.41, SD = 22.48; p = 0.015) revising on a website designed to revise the topic.

Table 5-33(b) summarises the mean (SD) examination scores for kind of revision done for Indian University. For Indian University, the analysis results suggested that there was a statistically significant difference in examination scores among students with different habits of revising programming topics using theory from lecture slides given by the lecturer (p = 0.019). In particular, the results of pairwise comparisons suggested that students who never revised programming topics using “theory from lecture slides given by the lecturer” had statistically significantly higher examination scores than students sometimes revised programming topics using theory from lecture slides given by the lecturer (*M* = 68.34, *SD* = 10.37 for "never" vs. *M* = 52.73, *SD* = 14.17 for "sometimes"; *p* = 0.021). This result is the opposite of the expected result, so the reason needs to be further investigated.

There was no statistically significant difference in examination scores across other categories of kind of revision done ($p > 0.05$).

So, Hypothesis H_1 was accepted for both universities.

Table 6-33(a): Mean (SD) examination scores kind of revision done for Australian University

Q13. Programming topics revised						
	1	2	3	4	5	p
Theory from lecture slides available on FLO	63.70 (23.18)	59.42 (25.23)	61.74 (20.02)	68.67 (19.88)	64.80 (21.98)	0.455
Textbook slides available on FLO	64.15 (22.88)	60.97 (23.85)	66.07 (17.66)	69.62 (20.78)	59.04 (23.20)	0.402
Laboratory work	67.70 (21.94)	59.48 (23.78)	65.56 (19.08)	63.36 (23.58)	64.06 (25.23)	0.502
View lectures online	62.23 (25.62)	65.75 (22.75)	61.76 (20.68)	67.29 (14.49)	55.83 (26.30)	0.562
Revised on a website designed to revise the topic	61.81 (24.19)	59.41 (22.48)	63.33 (18.13)	76.65 (10.65)	63.17 (30.84)	0.025*
Revised previous week's laboratory work	61.13 (25.05)	60.59 (22.74)	67.40 (16.55)	74.23 (18.21)	65.43 (29.02)	0.198
Revised new similar programs	62.01 (22.74)	59.80 (22.11)	70.34 (20.73)	74.71 (13.57)	59.83 (28.35)	0.112
Read previously done laboratory work	61.04 (25.19)	61.40 (22.84)	63.87 (18.60)	77.78 (12.56)	59.43 (26.03)	0.050
Redo previously done laboratory work	62.03 (23.59)	64.51 (22.20)	65.26 (18.50)	59.78 (14.80)	50.33 (36.14)	0.851
Read new similar programs	62.69 (21.68)	59.02 (22.65)	68.31 (19.99)	73.78 (25.65)	66.20 (33.80)	0.099

Redo new similar programs	61.43 (22.69)	66.27 (20.92)	65.77 (21.67)	69.57 (15.37)	49.80 (38.13)	0.727

(Note N = 183 for Q13 (Theory from lecture slides available on FLO, Redo previously done laboratory work, Redo new similar programs). 1 = never, 2 = sometimes, 3 = often, 4 = very often, and 5 = always. For Q20, 1 = not at all, 2 = slightly useful, 3 = useful, 4 = very useful, and 5 = extremely useful. SD = standard deviation. P = p-value of the Kruskal–Wallis tests. * indicates significance at the 0.05 level)

Table 5-33 (b): Mean (SD) examination scores kind of revision done for Indian University

Q13. Programming topics revised						
	1	2	3	4	5	p
Theory from lecture slides given by the lecturer	68.34 (10.37)	52.73 (14.17)	61.63 (14.15)	53.25 (15.54)	56.75 (15.37)	0.019*
Textbook chapters	62.89 (15.35)	56.81 (14.72)	60.49 (12.95)	57.11 (15.85)	58.53 (18.32)	0.781
Laboratory work	65.16 (14.84)	57.93 (14.01)	52.53 (15.69)	64.41 (14.14)	60.04 (10.97)	0.154
Watch subject related content on YouTube	57.83 (11.46)	56.72 (17.09)	61.55 (13.51)	57.68 (14.64)	67.90 (16.27)	0.503
Revised on a website designed to revise the topic	57.70 (13.83)	62.94 (14.12)	52.60 (15.41)	58.83 (15.27)	67.67 (6.79)	0.248
Revised previous week's laboratory work	62.46 (13.17)	58.74 (15.51)	57.27 (15.55)	59.03 (14.06)	52.44 (14.28)	0.817
Revised new similar programs	59.66 (13.83)	57.77 (14.04)	60.01 (16.80)	62.61 (10.94)	61.83 (19.62)	0.890
Read previously done laboratory work	61.68 (13.95)	55.38 (15.15)	60.24 (15.13)	59.39 (16.15)	64.38 (11.41)	0.539

Redo previously done laboratory work	65.07 (12.04)	59.02 (14.92)	52.33 (16.12)	60.68 (13.71)	58.58 (8.91)	0.166
Read new similar programs	61.03 (15.05)	58.20 (15.32)	57.34 (14.52)	62.65 (15.92)	58.25 (12.38)	0.823
Redo new similar programs	63.05 (14.22)	58.13 (16.01)	57.23 (14.36)	58.93 (13.98)	58.55 (14.22)	0.876

(*Note: For and Q13, 1 = never, 2 = sometimes, 3 = often, 4 = very often, and 5 = always SD = standard deviation. SD = standard deviation. NA = not available. P = p-value of the Kruskal–Wallis tests. *indicates significance at the 0.05 level. ** indicates significant at the 0.05 level for the Kruskal–Wallis tests, but not significant at the 0.05 level for the pairwise comparisons)*

6.5.11 Effect of seeking help through different sources in terms of scores obtained: Hypothesis 11

Table 5-34 (a) summarises the mean (SD) examination scores by sources of help for Australian University. There was a statistically significant difference in examination scores among students with different opinions regarding the importance of classmates (p = 0.029). In particular, students who thought classmates were never helpful had statistically significantly higher examination scores (M = 71.08, SD = 26.07) than students who thought classmates were sometimes useful (M = 58.27, SD = 23.11) (p = 0.020).

For the “Lecture notes/slides” option, the results of the Kruskal–Wallis tests were significant, but the results of the pairwise comparisons were not significant at the 0.05 level. Thus, these factors were concluded as not statistically significant to the examination scores.

The results are consistent with the results obtained in a study conducted by Butler, M. & Morgan, M. (2007) which suggested that study with peers was rated most important by only 3% of respondents(Butler and Morgan, 2007). Note for “Lecture notes/slides” for Q14, the results of the Kruskal–Wallis tests were significant, but the results of the pairwise comparisons were not significant at the 0.05 level. Thus, these factors were concluded as not statistically significant to the examination scores.

Table 5-34 (b) summarises the mean (SD) examination scores by sources of help for Indian University. For “Watch related content on YouTube” of Q14, the results of the Kruskal–Wallis tests were significant, but the results of the pairwise comparisons were not significant at the

0.05 level. Thus, this factor was concluded to be not statistically significant to the examination scores.

There was no statistically significant difference in examination scores across other categories of sources of help ($p > 0.05$) for both universities.

So, hypothesis H_1 is accepted for Australian University and hypothesis H_0 is accepted for Indian University.

Table 6-34(a): Mean (SD) examination scores by sources of help for Australian University

Q14. Sources of help						
	1	2	3	4	5	p
Classmates	71.08 (26.07)	58.27 (23.11)	59.95 (22.36)	64.71 (17.24)	65.65 (19.47)	0.029*
Senior students who have passed the topic	64.54 (25.04)	64.72 (19.38)	58.22 (21.08)	59.50 (16.47)	65.60 (6.54)	0.316
Lecturers	65.46 (22.02)	59.81 (20.83)	65.60 (25.40)	66.09 (20.46)	62.33 (30.55)	0.373

Textbooks	58.61 (26.39)	62.51 (21.06)	64.18 (23.48)	59.97 (20.91)	69.80 (19.92)	0.219
Lecture notes/slides	74.36 (18.93)	57.97 (21.25)	56.67 (25.54)	65.30 (21.47)	70.53 (17.37)	0.005**
Discussion forums	62.41 (22.90)	67.38 (20.63)	63.63 (14.74)	76.00 (1.73)	40.00 (43.84)	0.428
Facebook/Twitter	64.52 (22.36)	56.48 (22.36)	60.29 (21.98)	73.00 (2.83)	NA	0.292
Other social websites	63.45 (22.16)	65.45 (21.46)	56.40 (26.32)	71.00 (NA)	52.33 (40.05)	0.924
Opt for private tuition outside university	64.09 (22.24)	57.91 (22.50)	61.29 (20.01)	71.00 (NA)	37.00 (39.60)	0.549
Help sessions at university	63.84 (22.35)	64.52 (19.08)	55.17 (24.65)	64.33 (30.55)	9.00 (NA)	0.430

(Note: N = 184. 1 = never, 2 = sometimes, 3 = often, 4 = very often, and 5 = always. SD = standard deviation. NA = not available. P = p-value of the Kruskal–Wallis tests. * indicates significance at the 0.05 level. ** indicates significant at the 0.05 level for the Kruskal–Wallis tests, but not significant at the 0.05 level for the pairwise comparisons.)

Table5-34 (b): Mean (SD) examination scores by sources of help for Indian University

Q14. Source of help						
	1	2	3	4	5	p

Lecturers	63.82 (13.76)	55.00 (15.75)	57.49 (14.80)	63.98 (13.13)	55.83 (11.51)	0.321
Classmates	53.31 (11.67)	51.54 (13.30)	58.32 (17.72)	58.51 (13.08)	66.23 (10.83)	0.212
Senior students who have passed the topic	60.00 (16.65)	57.00 (12.51)	59.58 (18.39)	60.09 (15.09)	65.25 (9.79)	0.801
Textbooks	51.17 (16.78)	58.75 (15.91)	58.46 (12.58)	56.91 (17.01)	64.61 (13.51)	0.516
Lecture notes/slides	66.85 (14.33)	50.07 (15.11)	57.56 (13.58)	61.08 (16.17)	61.21 (13.95)	0.192
Facebook/Twitter	59.35 (14.47)	58.30 (17.31)	56.43 (13.53)	68.25 (17.32)	NA	0.703
Other social websites	60.51 (14.87)	55.10 (14.34)	55.15 (14.12)	80.25 (0.35)	60.88 (4.77)	0.105
Opt for private tuition outside university	60.34 (13.77)	45.31 (13.31)	53.00 (20.24)	63.75 (14.70)	50.00 (NA)	0.241
Opt to study the topics at training institutes teaching similar courses	59.70 (14.94)	52.50 (13.44)	52.28 (13.97)	64.50 (12.60)	72.08 (7.48)	0.151
Watch related content on YouTube	61.49 (12.83)	56.76 (15.02)	51.54 (14.72)	62.71 (15.23)	69.20 (8.96)	0.047**

(Note: For and Q14, 1 = never, 2 = sometimes, 3 = often, 4 = very often, and 5 = always SD = standard deviation. SD = standard deviation. NA = not available. P = p-value of the Kruskal–Wallis tests. *indicates significance at the 0.05 level. ** indicates significant at the 0.05 level for the Kruskal–Wallis tests, but not significant at the 0.05 level for the pairwise comparisons.)

6.5.12 Source of help helpful in terms of scores obtained: Hypothesis 12

Table 5-35 (a) summarizes the mean examination scores by sources of help for Australian University. The results of the Kruskal–Wallis tests suggested that there was a statistically significant difference in examination scores among students using textbooks as a helpful source (p = 0.013). In particular, the results of pairwise comparisons suggested that students who regarded textbooks as always useful had statistically significantly higher examination scores (*M* = 70.75, *SD* = 22.16) than students who regarded textbooks as not useful (*M* = 55.50, *SD* = 25.84) (*p* = 0.025). Note for “Opt for private tuition outside university” for Q15, the results of the Kruskal–Wallis tests were significant, but the results of the pairwise comparisons were not significant at the 0.05 level. Thus, these factors were concluded as not statistically significant to the examination scores.

Table 5-35 (b) summarizes the mean examination scores by sources of help for Indian University. The analysis results also suggested that there was a statistically significant difference in examination scores among students with different opinions of the usefulness of opting to study the topics at training institutes teaching similar courses ($p = 0.017$). In particular, the results of pairwise comparisons suggested that students who did not think it was useful had statistically significantly lower examination scores ($M = 50.09$, $SD = 13.65$) than students who would "not opt for this option" ($M = 63.97$, $SD = 13.82$); $p = 0.016$). Also, the students who thought that private tuition was useful most of the time and "always useful" had higher examination scores than the students who thought it was "not useful". It suggests that there may be some students who are top scorers and would not opt for this option but at the same time there may be students who may score better if they opt for "private tuition" as a source of help.

There was no statistically significant difference in examination scores across other categories of sources of help ($p > 0.05$) for both universities.

So, hypothesis H_1 was accepted for both universities.

Table 6-35(a): Mean (SD) examination scores by sources of help and activities in lectures for Australian University

Q15. Source of helpful help					
	1	2	3	4	P
Classmates	70.24 (26.09)	60.44 (20.75)	61.76 (21.79)	63.24 (20.34)	0.063
Senior students who have passed the topic	64.91 (24.58)	58.16 (25.69)	62.25 (16.91)	64.92 (18.06)	0.261
Lecturers	64.43 (22.36)	59.77 (19.07)	64.34 (20.23)	63.03 (27.09)	0.473
Textbooks	55.50 (25.84)	62.31 (20.33)	62.29 (21.43)	70.75 (22.16)	0.013*
Lecture notes/slides	68.60 (21.28)	63.02 (19.28)	60.87 (24.13)	66.23 (23.22)	0.337
Discussion forums	63.05 (22.20)	63.11 (24.03)	69.64 (16.57)	55.60 (28.29)	0.634

Facebook/Twitter	64.29 (22.78)	58.36 (20.13)	62.54 (21.01)	NA	0.274
Other social websites	63.43 (22.28)	67.00 (17.01)	59.70 (26.55)	52.33 (40.05)	0.950
Opt for private tuition outside university	63.57 (22.17)	56.46 (19.82)	74.92 (19.27)	36.00 (28.05)	0.018**
Help sessions at university	62.95 (22.59)	65.75 (18.30)	67.73 (24.16)	58.25 (33.65)	0.940

(Note: For Q15, 1 = not useful, 2 = useful sometimes, 3 = useful most of the time, and 4 = always useful. SD = standard deviation. NA = not available. P = p-value of the Kruskal–Wallis tests. *indicates significance at the 0.05 level. ** indicates significant at the 0.05 level for the Kruskal–Wallis tests, but not significant at the 0.05 level for the pairwise comparisons)

Table 5-35 (b): Mean (SD) examination scores by sources of help and activities in lectures for Indian University

Q15. Usefulness of help						
	1	2	3	4	5	p
Lecturers	66.29 (14.49)	58.52 (15.85)	55.73 (14.45)	59.33 (14.52)	70.50 (5.42)	0.287
Classmates	47.67 (3.62)	59.49 (15.95)	57.22 (14.95)	62.23 (13.78)	70.25 (NA)	0.411
Senior students who have passed the topic	52.67 (13.76)	63.42 (13.06)	55.19 (14.32)	55.97 (19.42)	60.31 (15.10)	0.245
Textbooks	57.25 (7.01)	55.93 (16.64)	60.55 (14.51)	59.48 (14.66)	51.15 (17.20)	0.725
Lecture notes/slides	65.56 (16.15)	59.61 (14.53)	57.24 (15.48)	57.71 (13.67)	72.67 (1.70)	0.409
Facebook/Twitter	54.88 (15.59)	60.16 (14.48)	57.54 (14.37)	69.58 (13.67)	59.70 (14.79)	0.546
Other social websites	55.01 (16.13)	58.55 (11.83)	55.55 (13.46)	62.61 (19.73)	62.47 (14.01)	0.375
Opt for private tuition outside university	54.44 (15.12)	59.19 (17.69)	51.96 (12.66)	66.75 (19.45)	61.99 (13.65)	0.167
Opt to study the topics at training institutes teaching similar courses	50.09 (13.65)	53.53 (11.15)	57.28 (14.88)	58.16 (16.50)	63.97 (13.82)	0.017*
Watch related content on YouTube	60.30 (15.62)	55.25 (14.19)	59.49 (12.68)	59.03 (17.58)	61.68 (14.06)	0.774

(Note. For Q15, 1 = not useful, 2 = useful sometimes, 3 = useful most of the time, 4 = always useful, and 5 = never opt for this option. SD = standard deviation. SD = standard deviation. NA

= not available. P = p-value of the Kruskal–Wallis tests. *indicates significance at the 0.05 level. ** indicates significant at the 0.05 level for the Kruskal–Wallis tests, but not significant at the 0.05 level for the pairwise comparisons)

6.5.13 Effect of attendance in lectures on performance in terms of scores obtained: Hypothesis 13

Table 5-36 (a) summarises the mean (SD) examination scores by the number of programming lectures attended for Australian University. The results of the Wilcoxon ranked–sum tests suggested that there was a statistically significant difference in examination scores among students attending different numbers of lectures (p = 0.002). In particular, according to the results of pairwise comparisons, students who attended 100% of the lectures had statistically significantly higher examination scores (M = 71.80, SD = 22.55) than students who attended up to 80% of the lectures (M = 57.77, SD = 22.53) (p = 0.002).

Table 5-36 (b) summarises Mean (SD) examination scores by the number of programming lectures attended for Indian University. The analysis results of the Wilcoxon ranked–sum tests suggested that there was no statistically significant difference in examination scores across categories of attendance of programming lectures/labs ($p > 0.05$).

The results obtained at Australian University were consistent with the results of the study conducted by (Massingham and Herrington, 2006) which concluded that "At the same time it is clear that attendance has an impact on performance. Students who attended lectures and tutorials had a better chance of success on all assessment tasks, in particular the final examination". Another study conducted by (Credé et al., 2010), also concluded that attendance correlates strongly with both performance in an individual class and college GPA. Also, the attendance–grade relationship was slightly stronger for science classes than for non-science classes. Another study conducted by Marburger concluded that students who missed class on a given date were significantly more likely to respond incorrectly to questions relating to material covered that day than students who were present(Marburger, 2006). Stanca also concluded that after controlling for unobservable student characteristics, attendance has a statistically significant and quantitatively relevant effect on student learning(Stanca, 2006).

The results obtained at Australian University could not be replicated for Indian University. For Indian University, the analysis results of the Wilcoxon ranked–sum tests suggested that there was no statistically significant difference in examination scores across categories of attendance at programming lectures/labs. At Indian University 70% attendance was compulsory for lectures and laboratories. The literature is inconclusive about this argument. A study conducted by Marburger concluded that an enforced mandatory attendance policy significantly reduces absenteeism and improves examination performance(Marburger, 2006). On the contrary, the study conducted by Credé et al., concluded that class attendance is a generally desirable behaviour, and there is encouraging evidence that mandatory policies are not necessary for dramatically improving class attendance or class performance(Credé et al., 2010). Since attendance was compulsory at Indian University, further study needs to be conducted to analyse the reasons why attendance proved to be ineffective in terms of student scores. Roddan also concluded that attendance did not elicit high correlations with examination scores(Roddan, 2002). He suggested that the low but significant correlation with overall attendance indicates that merely showing up at labs and tutorials is not the most important factor in getting a good grade and having a good attendance record does not mean that students are paying attention or understanding the material. He concluded that attendance, which would ordinarily be expected to correlate highly with examination performance, actually turned out not to do so.

The purpose of asking this question was two-fold: to determine if attending lectures helped students in learning programming by scoring better scores and if making attendance mandatory leads to better performance. The results suggest that attending more lectures may lead to better scores and making attendance mandatory may not necessarily improve student performance. Further study needs to be conducted to find out the reason for attendance having no correlation with student scores at Indian University and at the same time the reason for the high correlation of scores with attendance at Australian University needs to be further investigated, so that recommendations can be made to other universities to improve students' scores in learning programming.

So, hypothesis H_1 was accepted for Australian University and hypothesis H_0 was accepted for Indian University.

Table 6-36(a): Mean (SD) examination scores by number of programming lectures attended for Australian University

		Mean (SD)	p
Q16. Number of programming lectures attended	0%	52.50 (17.25)	0.002*
	Up to 20%	65.06 (19.65)	
	Up to 40%	59.09 (20.06)	
	Up to 60%	66.92 (21.39)	
	Up to 80%	57.77 (22.53)	
	100%	71.80 (22.55)	

(***Note: SD = standard deviation. P = p-value of the Wilcoxon ranked-sum tests (for gender) and the Kruskal–Wallis tests. * indicates significance at the 0.05 level)***

Table 5-36 (b): Mean (SD) examination scores by number of programming lectures attended for Indian University

		Mean (SD)	p
Q18. Number of programming lectures attended	Up to 20%	50.00 (NA)	0.207
	Up to 40%	51.15 (13.14)	
	Up to 60%	58.79 (17.12)	
	Up to 80%	58.39 (14.15)	
	Up to 100%	70.46 (7.83)	

(***Note: SD = standard deviation. P = p-value of the Wilcoxon ranked-sum tests (for gender) and the Kruskal–Wallis tests. * indicates significance at the 0.05 level)***

6.5.14 Effect of viewing lectures online on scores obtained by the students: Hypothesis 14

Table 5-37 summarises the mean (SD) examination scores by frequency and reason for viewing the programming lectures online for Australian University. Out of the various options available regarding viewing the lectures online, the results of the Kruskal–Wallis tests were significant for "The ones suggested by classmates" option, but the results of the pairwise comparisons were not significant at the 0.05 level. Thus, these factors were concluded as not statistically significant to the examination scores. There was no statistically significant difference in examination scores across other categories of viewing the programming lectures online ($p > 0.05$).

This question was only asked of Australian University students as there is no provision of watching lectures online for Indian University students.

Though the results of the Kruskal-Wallis test were significant, further research needs to be conducted in this area as the results from the study conducted by Traphagan et al. suggested that “for students with webcast access, more webcast viewing was associated with higher performance” (Traphagan et al., 2010). They also concluded that “webcast viewing appears to nullify the negative effect student absenteeism can have on student performance.” Another study conducted by Toppin also concluded that using VLC (Video Lecture Content) can potentially play a vital role in increasing academic performance and thereby improving retention(Toppin, 2011). There was no statistically significant difference in examination scores across other categories of viewing the programming lectures online.

At Indian University the students do not have the option to view the lectures online and 70% attendance is compulsory in lectures and laboratory.

Thus, hypothesis H_0 was accepted for Australian University.

Table 6-37: Mean (SD) examination scores by frequency and reason of viewing the programming lectures online

		1	2	3	4	5	p

Q17. How often do you view the programming lectures online?	All	62.10 (23.47)	64.19 (23.91)	64.65 (20.79)	68.43 (16.95)	61.04 (20.09)	0.776
	Important topics	61.74 (26.55)	62.36 (22.41)	62.76 (21.07)	67.46 (19.59)	64.83 (17.11)	0.846
	The ones difficult to understand	61.93 (27.76)	61.50 (20.83)	64.07 (21.71)	66.62 (19.70)	63.42 (19.05)	0.857
	The ones suggested by classmates	65.53 (24.91)	63.30 (20.41)	53.83 (20.52)	71.80 (10.28)	58.05 (18.19)	0.017**
	The ones suggested by lecturers	64.01 (25.85)	65.98 (22.06)	60.55 (20.15)	61.62 (19.61)	62.00 (17.17)	0.531
	If I need to understand a concept again	63.78 (27.63)	68.07 (20.58)	57.51 (20.51)	64.68 (21.11)	62.41 (20.02)	0.128
	If I need to take a note on some key points I missed	64.83 (25.34)	61.38 (22.85)	61.24 (20.07)	66.76 (22.30)	63.70 (19.19)	0.530

(Note: N = 184. 1 = never, 2 = sometimes, 3 = often, 4 = very often, and 5 = always. For Q20, 1 = not at all, 2 = slightly useful, 3 = useful, 4 = very useful, and 5 = extremely useful. SD = standard deviation. p = p-value of the Kruskal–Wallis tests. * indicates significance at the 0.05 level. ** indicates significant at the 0.05 level for the Kruskal–Wallis tests, but not significant at the 0.05 level for the pairwise comparisons)

6.5.15 Effect of activity in the lecture theatre on scores: Hypothesis 15

Table 5-38 (a) summarises the mean (SD) examination scores by sources of help and activities in lectures for Australian University. The results of the Kruskal–Wallis tests were significant for the students who opted for “Listen to the lecture”, but the results of the pairwise comparisons were not significant at the 0.05 level. Thus, these factors were concluded as not statistically significant to the examination scores.

Table 5-38 (b) summarises the mean (SD) examination scores by sources of help and activities in lectures for Indian University. For “Look up terms discussed in the lecture”, the results of the Kruskal–Wallis tests were significant, but the results of the pairwise comparisons were not significant at the 0.05 level. Thus, this factor was concluded as not statistically significant to the examination scores.

The analysis results of the Kruskal–Wallis tests suggested that there was no statistically significant difference in examination scores across other categories of sources of help and activities in lecture ($p > 0.05$).

The results obtained from this study were in contrast with the results obtained by Di Vesta and Gray, who suggested that "taking notes clearly led to an increase in the number of ideas recalled"(Di Vesta and Gray, 1972). The reason for this contrast may be that "the kind of note taking which serves a role in encoding should be much more efficient than one used only for external storage purposes." In this study, where one of the activities in the lecture theatre listed was taking notes or annotating the notes provided by the lecturer and their effect on student performance, the kind of notes taken by the students could not be evaluated and need further investigation.

Grabe concluded that taught lectures with overhead presentations provide the opportunity to make detailed notes. In addition, "students who used notes performed better in their examinations than non-note users" (Grabe, 2005). Grabe here refers to the notes provided by the lecturer beforehand(Grabe, 2005). Some activities for both Australian and Indian Universities showed interesting results and thus they can be further explored and the variables that proved to be ineffective may be omitted from any further study to be conducted.

So, hypothesis H_0 was accepted for both universities.

Table 6-38(a): Mean (SD) examination scores by sources of help and activities in lectures for Australian University

Q18. What do you do in the programming lecture theatre?					
	1	2	3	4	p
Listen to the lecture	42.33 (8.02)	61.38 (23.74)	62.20 (21.43)	68.06 (22.52)	0.044**
Listen and make notes	62.39 (21.66)	60.63 (22.37)	64.30 (22.11)	68.09 (23.23)	0.314
Annotate if you have printed notes	63.25 (22.45)	65.51 (20.40)	62.82 (23.50)	57.75 (27.05)	0.798
Play games on mobile phone/laptop	65.76 (22.43)	59.68 (21.57)	59.67 (19.05)	58.75 (33.39)	0.214
Look up terms discussed in the lecture	64.27 (22.42)	63.56 (22.85)	60.63 (18.87)	53.40 (25.44)	0.582
Use social media to socialize	66.41 (21.88)	60.88 (21.73)	50.33 (23.16)	56.25 (32.63)	0.094
Browse the internet in general	65.90 (22.02)	61.09 (22.18)	64.56 (21.64)	55.00 (39.85)	0.471

(*Note: For Q18, 1 = never, 2 = sometimes, 3 = large part of lecture, 4 = whole lecture. SD = standard deviation. NA = not available. P = p-value of the Kruskal–Wallis tests. *indicates significance at the 0.05 level. ** indicates significant at the 0.05 level for the Kruskal–Wallis tests, but not significant at the 0.05 level for the pairwise comparisons*)

Table 5-38 (b): Mean (SD) examination scores by sources of help and activities in lectures for Indian University

Q19. What do you do in the programming lecture theatre?					
Listen to the lectures	60.75 (15.88)	64.33 (13.02)	57.35 (15.17)	59.00 (15.17)	0.581

Listen and make notes	53.75 (14.50)	61.78 (14.41)	55.91 (15.04)	65.57 (12.08)	0.173
Annotate if you have printed notes	66.88 (12.94)	56.72 (14.09)	55.04 (17.10)	65.88 (10.34)	0.061
Play games on mobile phone/laptop	58.72 (14.47)	58.29 (15.40)	59.21 (15.52)	75.50 (NA)	0.725
Look up terms discussed in the lecture	68.18 (12.79)	56.33 (14.42)	56.22 (14.53)	64.88 (14.91)	0.043**
Use social media to socialize	56.62 (15.00)	61.53 (14.78)	63.30 (8.12)	75.50 (NA)	0.320
Browse the internet in general	57.47 (15.79)	61.38 (13.48)	49.46 (12.27)	63.25 (14.37)	0.227
Listen to the lectures	60.75 (15.88)	64.33 (13.02)	57.35 (15.17)	59.00 (15.17)	0.581

(Note: For Q19, 1 = never, 2 = sometimes, 3 = large part of lecture, 4 = whole lecture. SD = standard deviation. NA = not available. P = p-value of the Kruskal–Wallis tests. *indicates significance at the 0.05 level. ** indicates significant at the 0.05 level for the Kruskal–Wallis tests, but not significant at the 0.05 level for the pairwise comparisons)

6.5.16 Effect of practising laboratory work in the laboratory, library or home on performance of students in terms of scores: Hypothesis 16

Table 5-39(a) summarises the mean examination scores by laboratory work preference for Australian University. The results of the Wilcoxon ranked-sum tests suggested that students who preferred to do laboratory work at home (p = 0.022), would have statistically significantly higher examination scores than students who choose to do the laboratory work in the laboratory or library.

Table 5-39(b) summarises the mean examination scores by laboratory work preference for Indian University. Students who preferred to do laboratory work in the library had statistically significantly lower examination scores than students who did not (M = 52.11, SD = 14.70 vs. M = 60.54, SD = 14.33; p = 0.038). There was no statistically significant difference in examination scores across other categories of laboratory work preference ($p > 0.05$) for both universities.

So, hypothesis H_1 is accepted for both universities.

Table 6-39(a): Mean (SD) examination scores by laboratory work preference for Australian University

Q19. Preferences for doing laboratory work			
	Yes	No	p
In the laboratory	62.99 (22.52)	64.74 (21.71)	0.649
At home	66.08 (21.43)	57.45 (23.20)	0.022*
In the library	62.22 (22.25)	63.74 (22.39)	0.610

(Note: SD = standard deviation. P = p-value of the Wilcoxon ranked-sum tests. NA = not available. * indicates significance at the 0.05 level)

Table 5-39 (b): Mean (SD) examination scores by laboratory work preference for Indian University

Q19. Preferences for doing laboratory work			
	Yes	No	p
In the laboratory	58.10 (14.43)	61.14 (15.74)	0.353
At home	60.20 (14.27)	54.54 (15.64)	0.158
In the library	52.11 (14.70)	60.54 (14.33)	0.038*

(Note: SD = standard deviation. P = p-value of the Wilcoxon ranked-sum tests. NA = not available. * indicates significance at the 0.05 level)

6.5.17 Effect of students' perception of usefulness of attending labs on scores: Hypothesis 17

Table 5-40(a) summarizes the mean examination scores by attitude regarding the laboratory for Australian University.

There was a statistically significant difference in examination scores among students with different opinions regarding the usefulness of attending labs (p = 0.019). In particular, students who thought attending labs was not at all useful had statistically significantly lower examination scores (M = 39.17, SD = 16.25) than students who thought attending labs was slightly useful (M = 71.21, SD = 15.10; p = 0.021) or very useful (M = 66.09, SD = 23.35; p = 0.020). The results are similar to the results obtained by a study conducted by Robins which suggested that "laboratory attendance records as early as Week 1 are highly predictive of final grade in the course" (Robins, 2010). The results were also similar to the results obtained in a similar study conducted by Butler and Morgan, which suggested that laboratory classes were considered the most important study activity placed first by 31% of participants(Butler and Morgan, 2007).

Table 5-40(b) shows the mean examination scores by attitude regarding the laboratory for Indian University.

For Indian University students there was no statistically significant difference in examination scores across any category of attitude regarding the laboratory ($p > 0.05$).

Thus, hypothesis H_1 was accepted for Australian University and hypothesis H_0 was accepted for Indian University.

Hypotheses 18, 19, 20 and 21 include the background of the students studying programming. Some questions related to the background of the programming students were done by Roddan, but the parameters studied were not the ones evaluated in this study(Roddan, 2002). He concluded that "When looking at the background characteristics of students, the research is not in agreement, and as yet, no one core set of significant variables has been identified that predicts attrition. Which variables are important, and how they are significant is widely debated".

Some promising variables were identified through this study but further work needs to be done as the impact of the parameters was not same for Australian University as Indian University.

The four parameters studied are mentioned in the four upcoming hypotheses that were tested.

Table 6-40(a): Mean (SD) examination scores by attitude regarding the laboratory for Australian University

Q20. Do you find attending labs useful?						
	1	2	3	4	5	p
Australian University	39.17 (16.25)	71.21 (15.10)	62.47 (20.22)	66.09 (23.35)	59.80 (25.31)	0.019*

(Note: N = 184. For Q20, 1 = not at all, 2 = slightly useful, 3 = useful, 4 = very useful, and 5 = extremely useful. SD = standard deviation. NA = not available. P = p-value of the Kruskal–Wallis tests. * indicates significance at the 0.05 level. ** indicates significant at the 0.05 level for the Kruskal–Wallis tests, but not significant at the 0.05 level for the pairwise comparisons)

Table 5-40 (b): Mean (SD) examination scores by attitude regarding the laboratory for Indian University

Q20. Do you find attending labs useful?						
	1	2	3	4	5	p
Indian University	67.50 (14.64)	61.32 (13.24)	60.46 (15.90)	53.83 (14.25)	53.91 (12.68)	0.194

(Note:N= For For Q20, 1 = not at all, 2 = slightly useful, 3 = useful, 4 = very useful, and 5 = extremely useful. SD = standard deviation. NA = not available. P = p-value of the Kruskal–Wallis tests. * indicates significance at the 0.05 level. ** indicates significant at the 0.05 level for the Kruskal–Wallis tests, but not significant at the 0.05 level for the pairwise comparisons)

6.5.18 Effect of the family background of students whose parents/carers or siblings attended university: Hypothesis 18

Table 5-41 (a) shows the mean examination scores by family background of attending university for Australian University. The results of the Wilcoxon ranked-sum tests suggested that students who were the first one in the family to attend university would have statistically significantly lower examination scores than students who were not (p = 0.018). There was no statistically significant difference in examination scores in theory across other categories, such as family background of programming (p > 0.05).

Table 5-41 (b) shows the mean examination scores by family background of attending University for Indian University. There was no statistically significant difference in examination scores in theory across any category of family background of programming (p > 0.05).

So, hypothesis H_1 is accepted for Australian University and hypothesis H_0 is accepted for Indian University.

Table 6-41(a): Mean (SD) examination scores by family background of attending university for Australian University

Q22. First one attending university or have other family members attended university?			
	Yes	No	p
First one	59.81 (21.39)	65.70 (22.70)	0.018*
Siblings	61.28 (22.69)	65.21 (21.99)	0.209
Parents/Carers	62.92 (25.07)	63.98 (19.30)	0.653

Note*: SD = standard deviation. P = p-value of the Wilcoxon ranked-sum tests. NA = not available. * indicates significance at the 0.05 level)*

Table 5-41 (b): Mean (SD) examination scores by family background of attending University for Indian University

Q24. First one attending university or have other family members attended university?			
	Yes	No	p
First one	58.87 (15.35)	58.72 (14.52)	0.881
Siblings	56.87 (15.06)	60.85 (14.25)	0.218
Parents/Carers	59.65 (15.97)	58.00 (13.46)	0.586

6.5.19 Effect of whether or not the home environment is conducive to study on scores: Hypothesis 19

Table 5-42 shows the mean examination scores by home environment for Australian University and Indian University. The results of the Wilcoxon ranked-sum tests suggested that students who thought that their home environment was conducive to study (*p* = 0.023) have statistically significantly higher examination scores than students who did not think their home environment conducive to study. For Indian University, there was no statistically significant difference in examination scores across any category of home environment ($p > 0.05$).

The reason for these apparent discrepancies may be that the students at Indian University live in a university provided hostel where they co-habitat with other students studying the same course or similar courses, along with their classmates and other seniors. Thus, they spend most of their time at the university and hostel, therefore the home environment did not seem to have any effect on their scores, whereas at Australian University the students either live on their own or with their family and thus a home environment conducive to study showed a positive effect on scores.

So, hypothesis H_1 is accepted for Australian University and hypothesis H_0 is accepted for Indian University.

Table 6-42: Mean (SD) examination scores by Home environment

Q23. Home environment conducive to study?			
	Yes	No	p
Australian University	64.94 (22.69)	57.36 (19.95)	0.023*
Indian University	58.23 (14.55)	64.25 (16.04)	0.249

(Note: SD = standard deviation. P = p-value of the Wilcoxon ranked-sum tests. NA = not available. * indicates significance at the 0.05 level)

6.5.20 Effect of availability of programming-related help at home from their parents/carers or siblings on performance of students: Hypothesis 20

Table 5-43 shows the mean examination scores by family background of programming in terms of availability of programming-related help at home for Australian and Indian University. The results of the Wilcoxon ranked-sum tests suggested that there was no statistically significant difference in examination scores whether programming-related help was available at home or not for both universities.

So, hypothesis H_0 was accepted for both universities.

Table 6-43: Mean (SD) examination scores by family background for programming.

Q24. Can you get programming-related help at home from your parents/carers or siblings?			
	Yes	No	p
Australian University	64.21 (22.74)	63.33 (22.34)	0.612
Indian University	60.04 (14.68)	58.48 (14.82)	0.674

(Note: SD = standard deviation. P = p-value of the Wilcoxon ranked-sum tests. NA = not available. * indicates significance at the 0.05 level)

6.5.21 Effect of parental attitude towards educational goals on student performance: Hypothesis 21

Table 5-44 shows the mean examination scores by parental attitude towards students' educational goals for Australian and Indian University. The results of the Wilcoxon ranked-sum tests suggested that there was no statistically significant difference in examination scores whether parents/carers were supportive of the educational goals or not for both universities.

So, hypothesis H_0 was accepted for both universities.

Table 6-44: Mean (SD) examination scores by parental support

Q25. Are your parents/carers supportive of your educational goals?			
	Yes	No	p
Australian University	62.91 (22.98)	68.39 (14.77)	0.355
Indian University	59.17 (14.74)	46.00 (5.66)	0.222

(Note: SD = standard deviation. P = p-value of the Wilcoxon ranked-sum tests. NA = not available. * indicates significance at the 0.05 level)

6.5.22 Effect of the frequency of studying the topic on scores for Australian University: Hypothesis 22

Table 5-45 shows the mean examination scores by experience of studying the programming topic for Australian University. The results of the Kruskal–Wallis tests suggested that there was a statistically significant difference in examination scores for students with the number of times of studying the programming topic ($p = 0.606$). This suggests that studying programming more than once had a positive impact on student scores.

This question was not asked to Indian University students, as the students study the topic only once and if they are not able to pass the topic, they reappear for the examination in the next semester without having to undertake the topic again.

So, hypothesis H_0 was accepted for Australian University.

Table 6-45: Mean (SD) examination scores by experience of studying the programming topic for Australian University

Q26. Are you studying this topic for the first time?		
	Mean (SD)	P
1st time	63.90 (22.52)	0.606
2nd time	59.80 (22.18)	
3rd time	63.50 (0.71)	

(Note: SD = standard deviation. P = p-value of the Wilcoxon ranked-sum tests (for gender) and the Kruskal–Wallis tests. * indicates significance at the 0.05 level)

6.5.23 Effect of choice for students to study a topic related to programming again on scores: Hypothesis 23

Table 5-46 shows the mean examination scores by study choice for Australian University and Indian University. For Australian University, the results of the Wilcoxon ranked-sum tests suggested that students who would choose to study a topic/subject related to programming again ($p = 0.000$) would have statistically significantly higher examination scores than students who would not choose to study a topic/subject related to programming again. There was no statistically significant difference in examination scores for Indian University, based on the study choice ($p > 0.05$).

So, hypothesis H_0 was accepted for Australian University and hypothesis H_1 was accepted for Indian University.

This question was asked differently at Indian University as per recommendation from the coordinator at university: would you like to take a career or a job related to programming, testing, being a technical writer, or graphic designer?

6.5.24 Hypothesis 24 for Indian University

H_0: The students who would like to take up a career in programming do not perform better than those who would like to take up another career.

H_1: The students who would like to take up a career in programming perform better than those who would like to take up another career.

There was no statistically significant difference in examination scores across any category of attitude regarding future career. This question was asked only to Indian University students.

So, hypothesis H_0 was accepted for Indian University.

Table 6-46: Mean (SD) examination scores by study choice

Q27. If given an option, would you choose to study a topic/subject related to programming again?			
	Yes	No	p
Australian University	67.65 (23.75)	58.27 (19.36)	0.000*
Indian University	59.45 (14.63)	51.33 (14.84)	0.202

(Note: SD = standard deviation. P = p-value of the Wilcoxon ranked-sum tests. NA = not available. * indicates significance at the 0.05 level)

6.5.25 Effect of choice on students who would like to take up a career in programming on scores for Australian University: Hypothesis 24

Table 5-47 shows the mean examination scores by study/career choice for Australian University. The results of the Wilcoxon ranked-sum tests suggested that students who would like to take a career or a job related to programming (p = 0.004) would have statistically significantly higher examination scores than students who would not like to take a career or a job related to programming.

Table 6-47: Mean (SD) examination scores by career choice

	Yes	No	p
Q28. Would you like to take a career or a job related to programming?	66.93 (24.90)	60.62 (19.69)	0.004*

(Note: SD = standard deviation. P = p-value of the Wilcoxon ranked-sum tests. NA = not available. * indicates significance at the 0.05 level)

6.5.26 Effect of choice to take a career or a job related to programming, testing, technical writer, graphic designer: Hypothesis 25

Table 5-48 shows the mean examination scores by study/career choice for Indian University. There was no statistically significant difference in examination scores across any category of attitude regarding a future career (p > 0.05).

Table 6-48: Mean (SD) examination scores by career choice

Q29. Would you like to take a career or a job related to programming, testing, being a technical writer, or graphic designer?							
		1	2	3	4	5	p
	Programming	60.38 (13.50)	56.91 (16.46)	62.38 (16.24)	53.33 (12.87)	75.50 (NA)	0.424
	Testing	55.84 (16.56)	59.18 (14.04)	56.84 (17.30)	55.60 (12.19)	66.44 (12.84)	0.291
	Technical writer	69.06 (21.16)	60.61 (13.84)	57.24 (11.53)	55.80 (16.17)	53.97 (21.34)	0.503
	Graphic designer	54.63 (15.18)	60.54 (16.47)	58.08 (15.75)	60.10 (12.67)	63.11 (11.24)	0.628
	Teaching	52.92 (11.21)	58.61 (21.56)	57.31 (14.65)	58.38 (15.80)	62.06 (13.89)	0.587
	Research	60.33 (9.52)	65.82 (14.52)	56.75 (17.21)	58.66 (14.56)	53.09 (15.02)	0.075

(Note: For Q29, 1 = 1st choice, 2 = 2nd choice, 3 = 3rd choice, 4 = 4th choice, and 5 = 5th choice. SD = standard deviation. SD = standard deviation. NA = not available. P = p-value of the Kruskal–Wallis tests. *indicates significance at the 0.05 level. ** indicates significant at the 0.05 level for the Kruskal–Wallis tests, but not significant at the 0.05 level for the pairwise comparisons)

6.6 Summary

This chapter gave the description of the sample, the statistical analysis techniques used, and the statistical tests performed on the data. The data for both Australian University, Australia and Indian University, India were compared using statistical tests and the factors that affect learning of programming in terms of scores were identified. It was found that the factors affecting the performance of students in terms of scores differ on most parameters. The following chapter further explores the variables uncovered in this study, based on the interrelationship between the variables of interest.

CHAPTER 7 : ANALYSIS OF INTER-RELATIONSHIP BETWEEN FACTORS

7.1 Factors to be analysed

1. Do male students revise more than female students?
2. Do male students do preliminary preparation before lecture or laboratory more than female students?
3. Do students with prior programming experience tend to revise less or more than the students with no prior programming experience?
4. What kind of material is revised mostly by students with prior programming experience?
5. Would students with prior programming experience like to continue study programming if given an option and would they like to have a career in programming?
6. Do students who have studied algorithms or flowcharts have more interest in the topic?

   Parameters that can be evaluated are; (These questions have been asked in the questionnaire)

- Do these students do preliminary preparation?
- Do these students do revision?

7. Do the students who do preliminary preparation before lecture also tend to do the preliminary preparation before laboratory?
8. Do the students who do preliminary preparation also tend to revise the topic?
9. Do the students who do preliminary preparation before laboratory also do the revision?
10. Do the students who do preliminary preparation before lecture also tend to attend more lectures?
11. Do the students who do preliminary preparation before laboratory also tend to attend more lectures?
12. Do the students who do revision also tend to attend more lectures?
13. Do the students who studied programming before attending the course tend to attend more lectures?
14. Do the students who study programming out of interest tend to attend more lectures?

7.1.1 Analysis methods

To answer the research questions, the following variables were used in this study.

- Frequency of programming topics revised: Frequency of programming topics revised was computed by averaging the responses of the 11 sub-questions of Q13 (1 = never, 2 = sometimes, 3 = often, 4 = very often, and 5 = always). The scores of frequencies of programming topics revised ranged from 1 to 5. Higher scores indicated more frequent programming topics revised.

- Frequency of studying before going to the programming lecture: Frequency of studying before going to the programming lecture was computed by averaging the 6 sub-questions of Q10 (1 = never, 2 = sometimes, 3 = often, 4 = very often, and 5 = always). The scores of frequencies of studying before going to the programming lecture ranged from 1 to 5. Higher scores indicated more frequent studying before going to the programming lecture.

- Preliminary preparation before the laboratory: Preliminary preparation before the laboratory was determined using the 10 sub-questions of Q11 (yes vs. no). If any of the answers of the 10 sub-questions were "yes", then preliminary preparation before the laboratory = "Yes"; otherwise, preliminary preparation before laboratory = "No". Preliminary preparation before the laboratory is a binary variable.

- Prior programming experience: Prior programming experience was determined by the 5 sub-questions of Q5 (yes vs. no) (For Indian University, it was Q3). If any of the answers of the 5 sub-questions were "yes", then prior programming experience = "Yes"; otherwise, prior programming experience = "No". Prior programming experience is a binary variable.

- Experience of designing Flowcharts/Algorithms: Experience of designing Flowcharts/Algorithms was determined by the 2 sub-questions of Q7. There are three categories of experience of designing Flowcharts/Algorithms: none, Flowcharts or Algorithms, and both.

- Number of programming lectures attended: Number of programming lectures attended was determined by Q16 (for Indian University, it was Q18). There are two categories of number of programming lectures attended: < 60% vs. 60%+.

- Study programming out of interest: Study programming out of interest was determined using the first sub-question of Q8 (Interested to know about programming). If the answer was "somewhat important" or "most important", then study programming out of interest = "Yes"; otherwise, study programming out of interest = "No".

- Gender: a binary variable (male vs. female)

- Q27 (for Indian University, it was Q28). If given an option would you choose to study a topic/subject related to programming again? (Yes vs. No)

- Q28. Would you like to take a career or a job related to programming? (Yes vs. No)

To answer the research questions for Australian University and Indian University, the following analyses were conducted.

8. **Multiple linear regression** (Leutenegger and Edgington, 2007) was used to determine if there was a relationship between the dependent variable, frequency of programming topics revised, and the following independent variables, gender, prior programming experience, experience of designing Flowcharts/Algorithms, frequency of studying before going to the programming lecture, and preliminary preparation before laboratory.

The three assumptions of linear regression were checked:

- Independence of observations – residuals are independent.

- Normality: the distribution of the residuals is normal.

- Homoscedasticity: the residuals have constant variance (equal variance).

Normality was examined though the quantile-quantile (QQ) plots. The residual plots (residuals versus the fitted values) were used to investigate if the variance was constant/equal.

8. Multiple linear regression (Leutenegger and Edgington, 2007) was used to determine if there was a relationship between the dependent variable, frequency of studying before going to the programming lecture, and the following independent variables, gender, and experience of designing Flowcharts/Algorithms.

The three assumptions of linear regression were checked:

- Independence of observations – residuals are independent.
- Normality: the distribution of the residuals is normal.
- Homoscedasticity: the residuals have constant variance (equal variance).

Normality was examined though the quantile-quantile (QQ) plots. The residual plots (residuals versus the fitted values) were used to investigate if the variance was constant/equal.

3. **Chi-square tests of independence** were used to determine if there was an association between prior programming experience(Agresti, 2002), and Q27 (If given an option would you choose to study a topic/subject related to programming again?) and Q28 (Would you like to take a career or a job related to programming?).

4. **Multiple logistic regression for binary responses** was used to determine if there was a relationship between preliminary preparation before laboratory, and gender and frequency of studying before going to the programming lecture(Agresti, 2002). The Wald chi-square test was used to determine if a factor was significant. Odds ratios and 95% confidence intervals were computed to determine the strength of the association. Hosmer-Lemeshow goodness-of-fit test (Agresti, 2002) was used to determine the model adequacy (p-value > 0.05 indicates good model fit).

5. **Multiple logistic regression for binary responses** was used to determine if there was a relationship between number of programming lectures attended and frequency of studying before going to the programming lecture, preliminary preparation before laboratory, frequency

of programming topics revised, prior programming experience, and study programming out of interest(Agresti, 2002). The Wald chi-square test was used to determine if a factor were significant. Odds ratios and 95% confidence intervals were computed to determine the strength of the association. Hosmer-Lemeshow goodness-of-fit test was used to determine the model adequacy (p-value > 0.05 indicates good model fit) (Agresti, 2002).A p-value less than 0.05 indicate significance.

7.2 Results of Analysis for Australian University and Indian University

7.2.1 Relationship between the dependent variable, programming topics revised, and the following independent variables, gender, prior programming experience, experience of designing Flowcharts/Algorithms, frequency of studying before going to the programming lecture, and preliminary preparation before laboratories

Multiple linear regression was used to determine if there were a relationship between the dependent variable, programming topics revised, and the following independent variables, gender, prior programming experience, experience of designing Flowcharts/Algorithms, frequency of studying before going to the programming lecture, and preliminary preparation before laboratories.

7.2.1.1 Australian University

The Table 6-1 shows the regression results. The analysis results for Australian University indicate that there was a statistically significant relationship between frequency of programming topics revised and frequency of studying before going to the programming lecture (p = 0.000, Table 6-1). The parameter estimates of frequency of studying before going to the programming lecture was 0.539, indicating that for a one-unit increase of frequency of studying before going to the programming lecture, the frequency of programming topics revised would increase by 0.539 of a unit. In other words, there was a statistically significantly positive relationship between frequency of programming topics revised and frequency of studying before going to the programming lecture.

There was no statistically significant relationship between frequency of programming topics revised and, gender (p = 0.589), preliminary preparation before laboratory (p = 0.064), prior programming experience (p = 0.364), and experience of designing Flowcharts/Algorithms (p = 0.291) (Table 6-1). Table 6-2 shows the estimated means of frequency of programming topics revised by each level of independent variable.

The QQ plot (Figure 6-1) and the residual plot (Figure 6-2) suggested that the normality assumption and the homoscedasticity assumption of the model were satisfied.

Table 7-1: Regression results

Source	Type III Sum of Squares	DF	Mean Square	F	P	Partial Eta Squared
Intercept	14.944	1	14.944	44.719	0.000	0.197
Gender	0.098	1	0.098	0.292	0.589	0.002
Preliminary preparation before laboratory	1.165	1	1.165	3.485	0.064	0.019
Prior programming experience	0.277	1	0.277	0.829	0.364	0.005
Experience of designing Flowcharts/Algorithms	0.831	2	0.416	1.244	0.291	0.013
Frequency of studying before going to the programming lecture	36.641	1	36.641	109.647	0.000*	0.376
Error	60.819	182	0.334			
Total	1069.176	189				

(Note: DF = degrees of freedom, F = F-statistic, p = p-value. Partial eta squared represents the effect size. * indicates significant at the 0.05 level)

To explain what the intercept is in the model, we need to know how the variables were coded (see below) and what the regression model is (see the 2nd table (parameter estimates).

q13avg = frequency of programming topics revised

Gender: 1 = Male, 2 = Female

q11bin (Preliminary preparation before laboratory): 0 = No, 1 = Yes

q5bin (Prior programming experience): 0 = No, 1 = Yes

q7total (Experience of designing Flowcharts/Algorithms): 0 = None, 1 = Flowchart or algorithms, 2 = Both

q10avg (Frequency of studying before going to the programming lecture)

Note that for the predictors, gender, q11bin, q5bin, q7total are categorical variables, and q10avg is a continuous variable.

So according to the coding of the variables and the output of parameter estimate, the regression model can be written as

q13avg = intercept + b0*I(gender) + b1*I(q11bin) + b2*I(q5bin) + b3*I(q7total_1) + b4*I(q7total_2) + b5*q10avg

where b0-b5 are the regression coefficients, and I(gender), I(q11bin), I(q5bin), I(q7total_1), and I(q7total_2) are indicator functions

I(gender) = 1 if gender = 1; I(gender) = 0 if gender = 2

I(q11bin) = 1 if q11bin = 0; I(q11bin) = 0 if q11bin = 1

I(q5bin) = 1 if q5bin = 0; I(q5bin) = 0 if q5bin = 1

I(q7total_1) = 1 if q7total = 0; I(q7total_1) = 0, otherwise

I(q7total_2) = 1 if q7total = 1; I(q7total_2) = 0, otherwise

So, according to the results of parameter estimates, the regression equation can be written as

q13avg = 1.038 + 0.064*I(gender) -0.288*I(q11bin) + 0.082*I(q5bin) - 0.139*I(q7total_1) + 0.008*I(q7total_2) + 0.539*q10avg

The intercept is the estimated mean value of q13avg when all predictors = 0. In other words, the intercept (1.038) is the estimated mean value of programming topics revised, when I(gender) = 0 (i.e., gender = 2), I(q11bin) = 0 (i.e., q11bin = 1), I(q5bin) = 0 (i.e., q5bin = 1), I(q7total_1) = 0 (i.e., q7total = 1 or 2), I(q7total_2) = 0 (i.e., q7total = 0 or 2), and q10avg = 0. That is, the intercept is the estimated mean value of programming topics revised, when Gender = Female, Preliminary preparation before laboratory = Yes, Prior programming experience = Yes, Experience of designing Flowcharts/Algorithms = Both, and Frequency of studying before going to the programming lecture = 0.

The testing results for the intercept (in the first table, "tests of between subjects effects") tells that the intercept of the model is statistically significantly different from 0 as the p-value < 0.05.

Tests of Between-Subjects Effects

Dependent Variable: q13avg

Source	Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared	Noncent. Parameter	Observed Powerb
Corrected Model	48.412a	6	8.069	24.145	.000	.443	144.872	1.000

Intercept	14.944	1	14.944	44.719	.000	.197	44.719	1.000
Gender	.098	1	.098	.292	.589	.002	.292	.084
q11bin	1.165	1	1.165	3.485	.064	.019	3.485	.459
q5bin	.277	1	.277	.829	.364	.005	.829	.148
q7total	.831	2	.416	1.244	.291	.013	2.488	.268
q10avg	36.641	1	36.641	109.647	.000	.376	109.647	1.000
Error	60.819	182	.334					
Total	1069.176	189						
Corrected Total	109.231	188						

a. R Squared = .443 (Adjusted R Squared = .425)

b. Computed using alpha = .05

Parameter Estimates

Dependent Variable: q13avg

Parameter	B	Std. Error	t	Sig.	95% Confidence Interval Lower Bound	95% Confidence Interval Upper Bound	Partial Eta Squared	Noncent. Parameter	Observed Powerb
Intercept	1.038	.192	5.404	.000	.659	1.417	.138	5.404	1.000
[Gender=1.00]	.064	.118	.541	.589	-.169	.297	.002	.541	.084
[Gender=2.00]	0a	.	.	.	.	.	.	.	.
[q11bin=.00]	-.288	.154	-1.867	.064	-.593	.016	.019	1.867	.459
[q11bin=1.00]	0a	.	.	.	.	.	.	.	.
[q5bin=.00]	.082	.090	.910	.364	-.095	.258	.005	.910	.148
[q5bin=1.00]	0a	.	.	.	.	.	.	.	.
[q7total=.00]	-.139	.113	-1.226	.222	-.362	.085	.008	1.226	.230
[q7total=1.00]	.008	.134	.060	.952	-.256	.272	.000	.060	.050
[q7total=2.00]	0a	.	.	.	.	.	.	.	.
q10avg	.539	.051	10.471	.000	.437	.640	.376	10.471	1.000

a. This parameter is set to zero because it is redundant.

b. Computed using alpha = .05

Table 7-2: Estimated means of frequency of programming topics revised by each level of independent variable

		Estimated mean (SE)
Gender	Male	2.187 (0.079)
	Female	2.123 (0.129)
Preliminary preparation before laboratory	No	2.011 (0.154)
	Yes	2.299 (0.065)
Prior programming experience	N	2.196 (0.103)
	Yes	2.114 (0.098)
Experience of designing Flowcharts/Algorithms	None	2.060 (0.097)
	Flowcharts or Algorithms	2.206 (0.124)
	Both	2.198 (0.117)

(Note: SE = standard deviation)

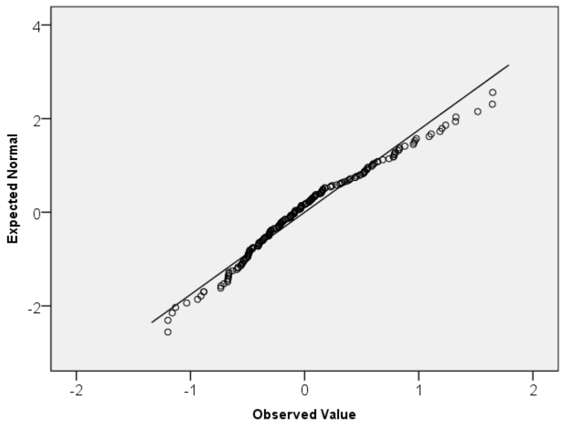


Figure 7-1: QQ plot

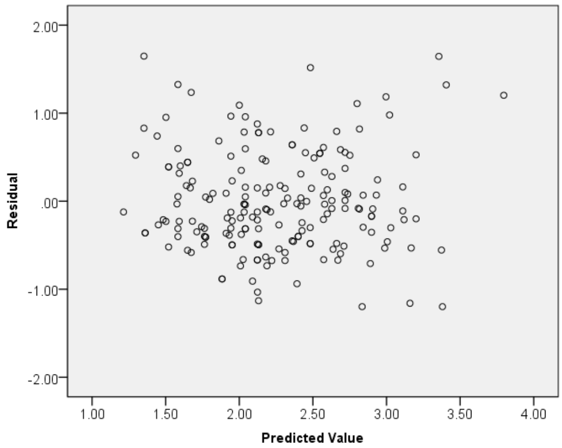


Figure 7-2: Residual plot

7.2.1.2 Indian University

Table 6-3 shows the regression results for Indian University. The analysis results for Indian University indicate that there was a statistically significant relationship between frequency of programming topics revised and frequency of studying before going to the programming lecture (p = 0.002, Table 6-3). The parameter estimates of frequency of studying before going to the programming lecture was 0.355, indicating that for one-unit increase of frequency of studying before going to the programming lecture, the frequency of programming topics revised would increase by 0.355 unit. In other words, there was a statistically significantly positive relationship between frequency of programming topics revised and frequency of studying before going to the programming lecture.

There was no statistically significant relationship between frequency of programming topics revised and, gender (p = 0.278), preliminary preparation before laboratories (p = 0.362), prior programming experience (p = 0.662), and experience of designing Flowcharts/Algorithms (p = 0.823) (Table 6-3). Table 6-4 shows the estimated means of frequency of programming topics revised by each level of independent variable.

The QQ plot (Figure 6-3) and the residual plot (Figure 6-4) suggested that the normality assumption and the homoscedasticity assumption of the model were satisfied.

Table 7-3: Regression results

Source	Type III Sum of Squares	DF	Mean Square	F	P	Partial Eta Squared
Intercept	21.695	1	21.695	45.976	0.000*	0.371
Gender	0.564	1	0.564	1.195	0.278	0.015
Preliminary preparation before laboratory	0.397	1	0.397	0.841	0.362	0.011
Prior programming experience	0.091	1	0.091	0.193	0.662	0.002
Experience of designing Flowcharts/Algorithms	0.184	2	0.092	0.195	0.823	0.005
Frequency of studying before going to the programming lecture	4.687	1	4.687	9.933	0.002*	0.113
Error	36.80	78	0.472			
Total	596.559	85				

(Note: DF = degrees of freedom, F = F-statistic, p = p-value. Partial eta squared represents the effect size. * indicates significant at the 0.05 level)

The coding of the variables is the same as the regression model of Table 6.1. So based on the results of parameter estimates, the regression equation can be written as

q13avg = 1.880 – 0.192*I(gender) -0.245*I(q11bin) - 0.106*I(q5bin) -0.019*I(q7total_1) + 0.094*I(q7total_2) + 0.355*q10avg

The intercept is the estimated mean value of q13avg when all predictors = 0. In other words, the intercept (1.880) is the estimated mean value of programming topics revised, when I(gender) = 0 (i.e., gender = 2), I(q11bin) = 0 (i.e., q11bin = 1), I(q5bin) = 0 (i.e., q5bin = 1), I(q7total_1) = 0 (i.e., q7total = 1 or 2), I(q7total_2) = 0 (i.e., q7total = 0 or 2), and q10avg = 0. That is, the intercept is the estimated mean value of programming topics revised, when Gender = Female, Preliminary preparation before laboratory = Yes, Prior programming experience = Yes, Experience of designing Flowcharts/Algorithms = Both, and Frequency of studying before going to the programming lecture = 0.

The testing results for the intercept (in the first table, “tests of between subjects effects”) tells that the intercept of the model is statistically significantly different from 0 as the p-value < 0.05.

Tests of Between-Subjects Effects

Dependent Variable: q13avg

Source	Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared	Noncent. Parameter	Observed Power[b]
Corrected Model	9.009[a]	6	1.502	3.182	.008	.197	19.093	.904
Intercept	21.695	1	21.695	45.976	.000	.371	45.976	1.000
Gender	.564	1	.564	1.195	.278	.015	1.195	.190
q11bin	.397	1	.397	.841	.362	.011	.841	.148
q3bin	.091	1	.091	.193	.662	.002	.193	.072
q7total	.184	2	.092	.195	.823	.005	.391	.079
q10avg	4.687	1	4.687	9.933	.002	.113	9.933	.875
Error	36.806	78	.472					
Total	596.559	85						
Corrected Total	45.815	84						

a. R Squared = .197 (Adjusted R Squared = .135)

b. Computed using alpha = .05

Parameter Estimates

Dependent Variable: q13avg

Parameter	B	Std. Error	t	Sig.	95% Confidence Interval Lower Bound	95% Confidence Interval Upper Bound	Partial Eta Squared	Noncent. Parameter	Observed Power[b]
Intercept	1.880	.349	5.384	.000	1.185	2.575	.271	5.384	1.000
[Gender=1.00]	-.192	.176	-1.093	.278	-.541	.158	.015	1.093	.190

[Gender=2.00]	0[a]	.	.	.	.	.	.	.	.
[q11bin=.00]	-.245	.267	-.917	.362	-.776	.286	.011	.917	.148
[q11bin=1.00]	0[a]	.	.	.	.	.	.	.	.
[q3bin=.00]	-.106	.242	-.439	.662	-.588	.376	.002	.439	.072
[q3bin=1.00]	0[a]	.	.	.	.	.	.	.	.
[q7total=.00]	-.019	.272	-.068	.946	-.560	.523	.000	.068	.051
[q7total=1.00]	.094	.177	.529	.598	-.259	.447	.004	.529	.082
[q7total=2.00]	0[a]	.	.	.	.	.	.	.	.
q10avg	.355	.113	3.152	.002	.131	.579	.113	3.152	.875

a. This parameter is set to zero because it is redundant.

b. Computed using alpha = .05

Table 7-4: Estimated means of frequency of programming topics revised by each level of independent variable

		Estimated mean (SE)
Gender	Male	2.350 (0.162)
	Female	2.542 (0.177)
Preliminary preparation before laboratory	No	2.324 (0.261)
	Yes	2.568 (0.100)
Prior programming experience	No	2.393 (0.225)
	Yes	2.499 (0.144)
Experience of designing Flowcharts/Algorithms	None	2.402 (0.204)
	Flowcharts or Algorithms	2.515 (0.195)
	Both	2.421 (0.201)

(Note: SE = standard deviation)

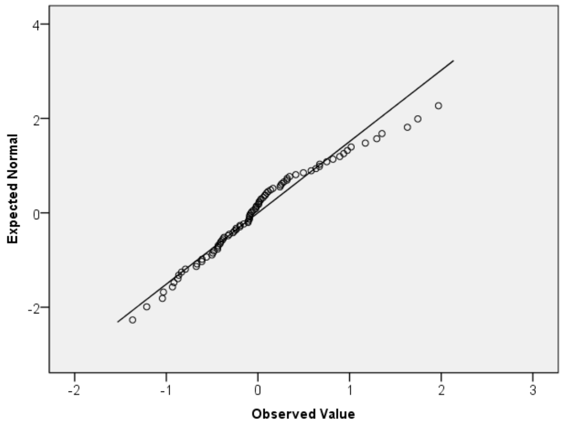


Figure 7-3: QQ plot

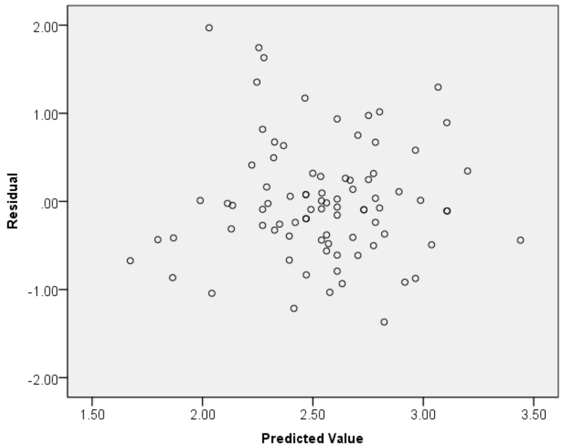


Figure 7-4: Residual plot

7.2.2 Relationship between the dependent variable, frequency of studying before going to the programming lecture, and the following independent variables, gender and experience of designing Flowcharts/Algorithms.

Multiple linear regression was used to determine if there was a relationship between the dependent variable, frequency of studying before going to the programming lecture, and the following independent variables, gender and experience of designing Flowcharts/Algorithms.

7.2.2.1 Australian University

Table 6-5 shows the regression results for Australian University. There was no statistically significant relationship between frequency of studying before going to the programming lecture and gender (p = 0.117) and experience of designing Flowcharts/Algorithms (p = 0.568) (Table 6-5). Table 6-6 shows the estimated means of frequency of studying before going to the programming lecture by each level of independent variable.

The QQ plot (Figure 6-5) and the residual plot (Figure 6-6) suggested that the normality assumption and the homoscedasticity assumption of the model were satisfied.

Table 7-5: Regression results

Source	Type III Sum of Squares	DF	Mean Square	F	P	Partial Eta Squared
Intercept	498.602	1	498.602	657.721	0.000*	0.777
Gender	1.880	1	1.880	2.480	0.117	0.013
Experience of designing Flowcharts/Algorithms	0.860	2	0.430	0.567	0.568	0.006
Error	143.276	182	0.758			
Total	1160.639	189				

(Note: DF = degrees of freedom, F = F-statistic, p = p-value. Partial eta squared represents the effect size. * indicates significant at the 0.05 level)

The coding of the variables is the same as the regression model of Table 6.1. So based on the results of parameter estimates, the regression equation can be written as

q10avg = 2.656 – 0.273*I(gender) -0.169*I(q7total_1) -0.155*I(q7total_2)

The intercept is the estimated mean value of q10avg when all predictors = 0. In other words, the intercept (2.656) is the estimated mean value of frequency of studying before going to the programming lecture, when I(gender) = 0 (i.e., gender = 2), I(q7total_1) = 0 (i.e., q7total = 1 or 2), and I(q7total_2) = 0 (i.e., q7total = 0 or 2). That is, the intercept is the estimated mean value of programming topics revised, when Gender = Female, and Experience of designing Flowcharts/Algorithms = Both.

The testing results for the intercept (in the first table, “tests of between subjects effects”) tells that the intercept of the model is statistically significantly different from 0 as the p-value < 0.05.

Tests of Between-Subjects Effects

Dependent Variable: q10avg

Source	Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared	Noncent. Parameter	Observed Power[b]
Corrected Model	2.822[a]	3	.941	1.241	.296	.019	3.723	.329
Intercept	498.602	1	498.602	657.721	.000	.777	657.721	1.000
Gender	1.880	1	1.880	2.480	.117	.013	2.480	.347
q7total	.860	2	.430	.567	.568	.006	1.135	.143
Error	143.276	189	.758					
Total	1160.639	193						
Corrected Total	146.099	192						

a. R Squared = .019 (Adjusted R Squared = .004)

b. Computed using alpha = .05

Parameter Estimates

Dependent Variable: q10avg

Parameter	B	Std. Error	t	Sig.	95% Confidence Interval Lower Bound	95% Confidence Interval Upper Bound	Partial Eta Squared	Noncent. Parameter	Observed Power[b]
Intercept	2.656	.199	13.333	.000	2.263	3.048	.485	13.333	1.000
[Gender=1.00]	-.273	.173	-1.575	.117	-.615	.069	.013	1.575	.347
[Gender=2.00]	0[a]	.	.	.	.	.	.	.	.
[q7total=.00]	-.169	.161	-1.050	.295	-.487	.148	.006	1.050	.181
[q7total=1.00]	-.155	.200	-.776	.439	-.550	.240	.003	.776	.121
[q7total=2.00]	0[a]	.	.	.	.	.	.	.	.

a. This parameter is set to zero because it is redundant.

b. Computed using alpha = .05

Table 7-6: Estimated means of frequency of studying before going to the programming lecture by each level of independent variable

		Estimated mean (SE)
Gender	Male	2.274 (0.076)
	Female	2.547 (0.164)
Experience of designing Flowcharts/Algorithms	None	2.350 (0.100)
	Flowcharts or Algorithms	2.364 (0.158)
	Both	2.519 (0.150)

(Note: SE = standard deviation)

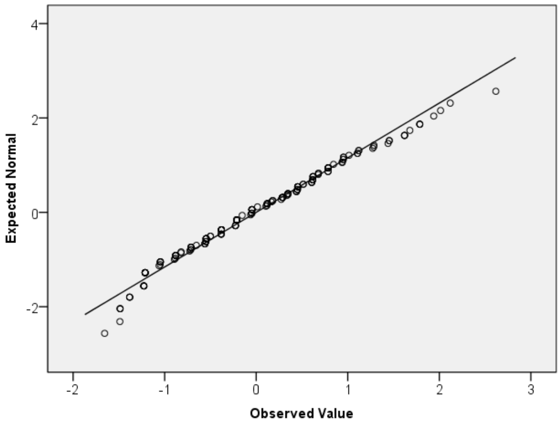


Figure 7-5: QQ plot

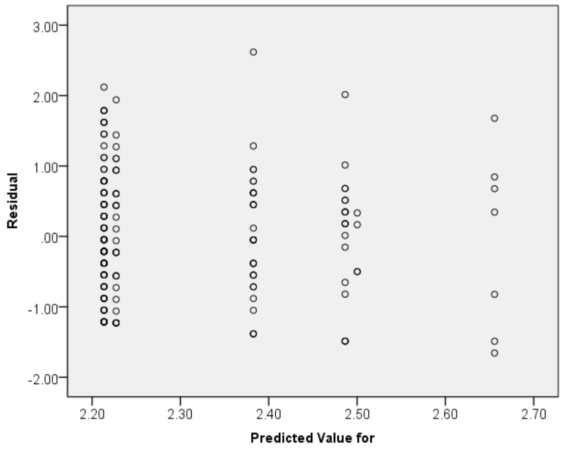


Figure 7-6: Residual plot

7.2.2.2 Indian University

Table 6-7 shows the regression results for Indian University. There was no statistically significant relationship between frequency of studying before going to the programming lecture and, gender (p = 0.981) and experience of designing Flowcharts/Algorithms (p = 0.300) (Table 6-7). Table 6-8 shows the estimated means of frequency of studying before going to the programming lecture by each level of independent variable.

The QQ plot (Figure 6-7) and the residual plot (Figure 6-8) suggested that the normality assumption and the homoscedasticity assumption of the model were satisfied.

Table 7-7 : Regression results

Source	Type III Sum of Squares	DF	Mean Square	F	P	Partial Eta Squared
Intercept	388.955	1	388.955	677.421	0.000	0.892
Gender	0.000	1	0.000	0.001	0.981	0.000
Experience of designing Flowcharts/Algorithms	1.403	2	0.701	1.221	0.300	0.029

Error	47.082	82	0.574			
Total	502.720	86				

(Note: DF = degrees of freedom, F = F-statistic, p = p-value. Partial eta squared represents the effect size. * indicates significant at the 0.05 level)

The coding of the variables is the same as the regression model of Table 6.1. So based on the results of parameter estimates, the regression equation can be written as

q10avg = 2.452 – 0.004*I(gender) -0.331*I(q7total_1) -0.222*I(q7total_2)

The intercept is the estimated mean value of q10avg when all predictors = 0. In other words, the intercept (2.452) is the estimated mean value of frequency of studying before going to the programming lecture, when I(gender) = 0 (i.e., gender = 2), I(q7total_1) = 0 (i.e., q7total = 1 or 2), and I(q7total_2) = 0 (i.e., q7total = 0 or 2). That is, the intercept is the estimated mean value of programming topics revised, when Gender = Female, and Experience of designing Flowcharts/Algorithms = Both.

The testing results for the intercept (in the first table, "tests of between subjects effects") tells that the intercept of the model is statistically significantly different from 0 as the p-value < 0.05.

Tests of Between-Subjects Effects

Dependent Variable: q10avg

Source	Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared	Noncent. Parameter	Observed Power[b]
Corrected Model	1.617[a]	3	.539	.939	.426	.033	2.817	.248
Intercept	388.955	1	388.955	677.421	.000	.892	677.421	1.000
Gender	.000	1	.000	.001	.981	.000	.001	.050
q7total	1.403	2	.701	1.221	.300	.029	2.443	.260
Error	47.082	82	.574					
Total	502.720	86						
Corrected Total	48.700	85						

a. R Squared = .033 (Adjusted R Squared = -.002)

b. Computed using alpha = .05

Parameter Estimates

Dependent Variable: q10avg

Parameter	B	Std. Error	t	Sig.	95% Confidence Interval		Partial Eta Squared	Noncent. Parameter	Observed Power[b]
					Lower Bound	Upper Bound			
Intercept	2.452	.201	12.185	.000	2.052	2.853	.644	12.185	1.000
[Gender=1.00]	-.004	.189	-.024	.981	-.380	.371	.000	.024	.050
[Gender=2.00]	0[a]	.	.	.	.	.	.	.	.
[q7total=.00]	-.331	.230	-1.441	.153	-.788	.126	.025	1.441	.296
[q7total=1.00]	-.222	.191	-1.161	.249	-.602	.158	.016	1.161	.209
[q7total=2.00]	0[a]	.	.	.	.	.	.	.	.

a. This parameter is set to zero because it is redundant.

b. Computed using alpha = .05

Table 7-8: Estimated means of frequency of studying before going to the programming lecture by each level of independent variable

		Estimated mean (SE)
Gender	Male	2.264 (0.111)
	Female	2.268 (0.143)
Experience of designing Flowcharts/Algorithms	None	2.119 (0.172)
	Flowcharts or Algorithms	2.228 (0.144)
	Both	2.450 (0.140)

(Note: SE = standard deviation)

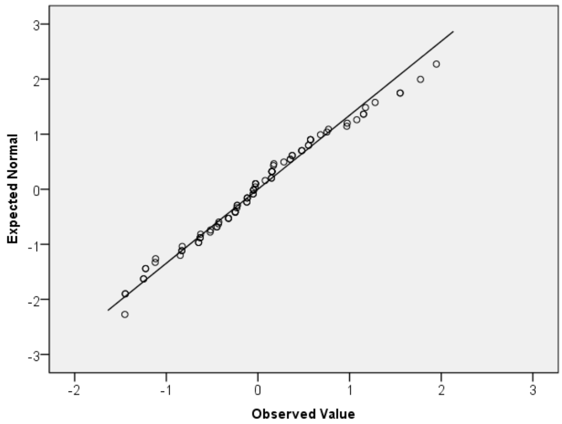


Figure 7-7: QQ plot

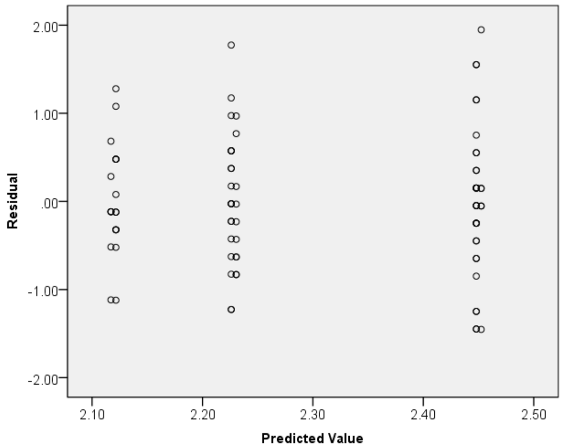


Figure 7-8: Residual plot

7.2.3 Association between prior programming experience, and Q27 (If given an option would you choose to study a topic/subject related to programming again?) and Q28 (Would you like to take a career or a job related to programming?)

Chi-square tests of independence were used to determine if there was an association between prior programming experience, and Q27 (If given an option would you choose to study a topic/subject related to programming again?) and Q28 (Would you like to take a career or a job related to programming?).

7.2.3.1 Australian University

Table 6-9 represents Two-way table of prior programming experience, and Q27 (If given an option would you choose to study a topic/subject related to programming again?) and Q28 (Would you like to take a career or a job related to programming?)

The analysis results for Australian University indicate that there was a statistically significant association between prior programming experience, and Q27 (If given an option would you choose to study a topic/subject related to programming again?) (χ^2 (N = 187, 1) = 4.500, p = 0.034). In particular, students with prior programming experience were more likely (62.5%) to choose to study a topic/subject related to programming again than those with no prior programming experience (47.0%).

The analysis results also indicate that there was a statistically significant association between prior programming experience, and Q28 (Would you like to take a career or a job related to programming?). (χ^2 (N = 187, 1) = 12.017, p = 0.001). In particular, students with prior programming experience were more likely (56.7%) to choose to take a career or a job related to programming than those with no prior programming experience (31.3%).

Table 7-9: Two-way table of prior programming experience, and Q27 (If given an option would you choose to study a topic/subject related to programming again?) and Q28 (Would you like to take a career or a job related to programming?)

		Prior programming experience	
		No	Yes
If given an option would you choose to study a topic/subject related to programming again?	Yes	39 (47.0)	65 (62.5)
	No	44 (53.0)	39 (37.5)
Would you like to take a career or a job related to programming?	Yes	26 (31.3)	59 (56.7)
	No	57 (68.7)	45 (43.3)

(Note: Numbers in parentheses are %)

7.2.3.2 Indian University

Table 6-10 represents Two-way table of prior programming experience, and Q27 (If given an option would you choose to study a topic/subject related to programming again the analysis results for Indian University indicate that there was no statistically significant association between prior programming experience, and Q28 (If given an option would you choose to study a topic/subject related to programming again?) (χ^2 (N = 79, 1) = 0.537, p = 0.604).

Table 7-10 : Two-way table of prior programming experience, and Q27 (If given an option would you choose to study a topic/subject related to programming again?)

		Prior programming experience	
		No	Yes
If given an option would you choose to study a topic/subject related to programming again?	Yes	15 (88.2)	58 (93.5)
	No	2 (11.8)	4 (6.5)

(Note: Numbers in parentheses are %.)

7.2.4 Relationship between preliminary preparation before laboratory, and gender and frequency of studying before going to the programming lecture.

Multiple logistic regression for binary responses was used to determine if there were a relationship between preliminary preparation before laboratories, and gender and frequency of studying before going to the programming lecture.

7.2.4.1 Australian University

The analysis results for Australian University are presented in Table 6-11. There was a statistically significantly relationship between preliminary preparation before laboratories and frequency of studying before going to the programming lecture (p = 0.000). Students with higher frequency of studying before going to the programming lecture odds ratio were more likely to do preliminary preparation before laboratories (OR = 5.251, 95% CI = (2.105, 13.098)). There was no statistically significant relationship between preliminary preparation before laboratories and gender (p = 0.338). The Hosmer-Lemeshow goodness-of-fit test indicated that the model fit was adequate (p = 0.192).

Table 7-11: Results of the logistic regression

	Parameter estimate	SE	Wald	DF	p	Odds ratios
Constant	0.155	1.255	0.015	1	0.902	
Gender	1.046	1.092	0.918	1	0.338	0.351 (0.041, 2.986)
Frequency of studying before going to the programming lecture	1.658	0.466	2.644	1	0.000*	5.251 (2.105, 13.098)

(Note: The logistic regression modelled the probability of preliminary preparation before laboratories = "Yes". For gender, female was the reference group. SE = standard deviation, Wald = Wald chi-square statistic, DF = degrees of freedom, p = p-value. * indicates significance at the 0.05 level. Numbers in parentheses are 95% confidence limits)

7.2.4.2 Indian University

The analysis results for Indian University are presented in Table 6-12. There was a statistically significantly relationship between preliminary preparation before laboratories and frequency of studying before going to the programming lecture (p = 0.001). Students with higher frequency of studying before going to the programming lecture odds ratio were more likely to do preliminary preparation before laboratories (OR = 16.030, 95% CI = (2.973, 86.444)). There was no statistically significant relationship between preliminary preparation before laboratory and gender (p = 0.479). The Hosmer-Lemeshow goodness-of-fit test indicated that the model fit was adequate (p = 0.811).

Table 7-12: Results of the logistic regression

	Parameter estimate	SE	Wald	DF	p	Odds ratios
Constant	3.324	1.542	4.647	1	0.031	
Gender	0.593	0.837	0.501	1	0.479	1.809 (0.351, 9.332)
Frequency of studying before going to the programming lecture	2.774	0.860	10.415	1	0.001*	16.030 (2.973, 86.444)

(Note: The logistic regression modelled the probability of preliminary preparation before laboratories = "Yes". For gender, female was the reference group. SE = standard deviation, Wald = Wald chi-square statistic, DF = degrees of freedom, p = p-value. * indicates significance at the 0.05 level. Numbers in parentheses are 95% confidence limits)

7.2.5 Relationship between number of programming lectures attended and frequency of studying before going to the programming lecture, preliminary preparation before laboratory, frequency of programming topics revised, prior programming experience, and study programming out of interest.

Multiple logistic regression for binary responses was used to determine if there were a relationship between number of programming lectures attended and frequency of studying before going to the programming lecture, preliminary preparation before laboratory, frequency of programming topics revised, prior programming experience, and study programming out of interest.

7.2.5.1 Australian University

The analysis results for Australian University are shown in Table 6-13. There was a statistically significant relationship between the number of programming lectures attended and frequency of studying before going to the programming lecture (p = 0.037). Students with higher frequency of studying before going to the programming lecture were more likely to have attended 60%+ programming lectures (OR = 1.683, 95% CI = (1.032, 2.745)).

There was a statistically significant relationship between the number of programming lectures attended and prior programming experience (p = 0.035). Students with no prior programming experience were more likely to have attended 60%+ programming lectures than students with prior programming experience (OR = 1.987, 95% CI = (1.051, 3.756)).

There was no statistically significant relationship between the number of programming lectures attended and, preliminary preparation before laboratory (p = 0.820), frequency of programming topics revised (p = 0.569), and study programming out of interest (p = 0.246).

The Hosmer-Lemeshow goodness-of-fit test indicated that the model fit was adequate (p = 0.871).

Table 7-13: Results of the logistic regression

	Parameter estimate	SE	Wald	DF	p	Odds ratios
Constant	0.688	0.573	1.441	1	0.230	
Study programming out of interest	0.451	0.388	1.349	1	0.246	1.570 (0.733, 3.359)
Prior programming experience	0.687	0.325	4.469	1	0.035*	1.987 (1.051, 3.756)
Frequency of studying before going to the programming lecture	0.521	0.250	4.354	1	0.037*	1.683 (1.032, 2.745)
Preliminary preparation before laboratory	0.127	0.560	0.052	1	0.820	0.881 (0.294, 2.638)
Frequency of programming topics revised	0.157	0.276	0.325	1	0.569	0.855 (0.498, 1.467)

(Note: The logistic regression modelled the probability of number of programming lectures attended = "60%+". For study programming out of interest, prior programming experience, and preliminary preparation before laboratories, "Yes" was the reference group. SE = standard deviation, Wald = Wald chi-square statistic, DF = degrees of freedom, p = p-value. * indicates significance at the 0.05 level. Numbers in parentheses are 95% confidence limits)

7.2.5.2 Indian University

The analysis results for Indian University are shown in Table 6-14. There was no statistically significant relationship between number of programming lectures attended and, frequency of studying before going to the programming lecture (p = 0.364), preliminary preparation before laboratory (p = 0.639), frequency of programming topics revised (p = 0.123), prior programming experience (p = 0.053), and study programming out of interest (p = 0.242).

The Hosmer-Lemeshow goodness-of-fit test indicated that the model fit was adequate (p = 0.871).

Table 7-14: Results of the logistic regression

	Parameter estimate	SE	Wald	DF	p	Odds ratios
Constant	1.450	1.239	1.370	1	0.242	
Study programming out of interest	-1.718	1.138	2.276	1	0.131	0.179 (0.019, 1.671)
Prior programming experience	1.619	0.838	3.730	1	0.053	5.048 (0.976, 26.106)
Frequency of studying before going to the programming lecture	0.359	0.395	0.825	1	0.364	1.431 (0.660, 3.102)
Preliminary preparation before laboratory	0.458	0.975	0.220	1	0.639	1.580 (0.234, 10.691)
Frequency of programming topics revised	-0.612	0.397	2.375	1	0.123	0.542 (0.249, 1.181)

(Note: The logistic regression modelled the probability of number of programming lectures attended = "60%+". For study programming out of interest, prior programming experience, and preliminary preparation before laboratories, "Yes" was the reference group. SE = standard deviation, Wald = Wald chi-square statistic, DF = degrees of freedom, p = p-value. * indicates significance at the 0.05 level. Numbers in parentheses are 95% confidence limits)

7.3 Summary

This chapter explored the interrelationship between the primary variables/factors analysed in the previous chapter. The variables to be further explored for analysis were identified and suitable analysis methods were used to determine the inter-relationship between factors. The analysis was done for both Universities and some significant relationships were derived from the analysis. Some similarities of associations were observed between the two chosen universities. The upcoming chapter explains reasons to study the Course Management System (FLO) and social media (Facebook) and the results obtained from this study.

CHAPTER 8 : PEER GROUP INTERACTION AND SOCIAL INTEGRATION THROUGH USE OF CMS AND A FACEBOOK GROUP

This chapter explains the various trials conducted for both the CMS and Facebook group and the results achieved. Also included is a comparison between Flinders Learning Online (FLO) and Facebook usage by the students across four semesters. This comparison was done to study if either or both of the social media, i.e. Facebook or the CMS system, help improve student engagement and serve as an additional source of peer-to-peer interaction and social integration in the process of learning programming at a tertiary level of education. The analysis was done by comparing student performance in terms of scores. The parameters used for analysis were the number of posts initiated, number of responses and their effect on both the Facebook group and FLO.

8.1 Why was Social Media chosen as a mode of peer group interaction and social integration?

Social media has been gaining popularity at an exponential rate. Membership of online social networks has recently exploded at an exponential rate. Be it a University campus, public transport or an airport waiting area, or even a bus stop, people, especially the younger generation, can be seen socializing. There is a fair amount of professional and popular interest in the effects of social media on college student development and success (Abramson, 2011). Indeed, the internet is playing an increasingly important role, not only in students' social lives, but also their academic lives. Educators are now turning to Web 2.0 tools, drawing upon their ability to assist in creating, collaborating and sharing content. As a result of this, the use of social sharing sites is increasing daily (Uzunboylu et al., 2011); (Lenhart and Madden, 2007); (Selwyn, 2007).

At Australian University, the Course Management System(FLO) is available to all students. It has a group discussion/chat feature that could be used as a mode of peer group interaction and social integration. Thus, two forms of peer to peer interaction and social integration were studied in parallel.

8.2 Why Facebook was chosen out of the three most popular types of social media available

The most popular social media website for college students is Facebook, and research shows that anywhere between 85 and 99% of college students use it (Hargittai, 2007); (Jones and Fox, 2009); (Matney and Borland, 2009). It is quickly turning into one of the most popular tools for social communication (Ross et al., 2009). It has been gaining exponential popularity and web space since its inception. Hargittai suggests that Facebook has become one of the most popular online destinations(Hargittai, 2007). Stutzman suggests that Facebook owns each campus, as the use of Facebook is exceptional(Stutzman, 2006). Maloney suggests that, "social networking sites such as Facebook and Myspace have shown, among other things, that students will invest time and energy in building relationships around shared interests and knowledge communities" (Maloney, 2007).

Facebook can be accessed both on the web and through mobile devices. Whilst Facebook may appear to be like many other SNSs, its combination of self-presentation, prurient viewing of others' personal information and situational relevance to campus life has certainly proved attractive to student users (Selwyn, 2007).

8.3 Why was Facebook used for this study?

As is evident from the literature, Facebook is the most popular form of social media, so it was an ideal choice for the study.

Mason discusses the features of Facebook, claiming that Facebook is imbued with many of the desired qualities of an effective education technology in its discussion elements support peer learning (Mason, 2006). Maloney suggests that, in particular, the conversational and communal qualities of Facebook are seen to "mirror much of what we know to be good models of learning, in that they are collaborative and encourage active participatory role for users"

(Maloney, 2007). Smith and Peterson argue that “knowledge is not constructed in an individual vacuum, but in the communication and exchanges embedded in social networks” (Smith and Peterson, 2007).

A study conducted by McCartney et al. explored the idea that students get stuck while learning programming(McCartney et al., 2007). While analysing the strategies they use for getting unstuck, it was found that social interaction with friends and instructors surfaced as a key strategy and the importance of social interaction was striking. However, it has been observed at the universities that students are socialising more online than face-to-face. It is not uncommon to find students sitting next to each other in a lecture theatre, laboratory or at campus, socialising online rather than face-to-face. This is where Facebook can play an important role. Facebook or the CMS system can help improve student engagement and serve as an additional source of peer-to-peer interaction and social integration. Many programmers experience the problem of getting stuck at a point where a little help from a peer may save hours of agonizing searching for answers. Since the lecturer cannot be present online all the time, social media fills the need to access immediate help.

At Australian University as well as Indian University, a large number of Facebook groups have been created by lecturers and students. These groups are used informally to communicate with the students studying similar topics or students with similar interests. Some of the groups at both universities are closed groups, while other groups are open. These groups are used by students to communicate with each other. Thus, it was reasonable to explore the effect these groups may have on the educational outcomes of the students in a formal setting.

8.4 The features of Facebook that have been used in this study

The Facebook features that have been used in this study are the Facebook walls and Facebook groups. In the Facebook group that was created for this study, we can use the most-used feature of Facebook, which is the Facebook ‘wall’ which is an asynchronous chat facility owned by each user (Lenhart et al., 2010). The wall is the most conventional – computer mediated – communication feature of Facebook, and certainly a central element of Facebook’s rapid growth into a social networking site par excellence (Selwyn, 2007).

8.4.1 Definition of a Facebook group:

Facebook Groups make it easy to connect with specific sets of people, like family, teammates or co-workers. **Groups** are dedicated spaces where updates, photos or documents can be shared and messages can be sent and replied to by other group members (Facebook, 2017).

Facebook groups may be open to anyone or joined by invitation only. A Facebook group allows members to create a community by promoting, sharing and discussing common topics (techopedia, 2017).

The main purpose of groups is to help Facebook users socialize around any topic or community. Groups also have the ability to message their members, as well as restrict who can and cannot join.

8.4.2 Group Notifications

All messages (or any other objects) posted to the wall in within groups will generate notifications. These notifications drive users back to the group, however users can also opt-out of these notifications by clicking on the "Edit Notifications" button.

8.4.2.1 Group Chat

Group chat is probably one of the most significant features of the new Facebook Groups product. All members of a group have the ability to engage in a single chat window (O'Neill, 2010).

In this study, a closed Facebook group has been used. Closed groups can be seen by the public. If you create a closed group, the name of it, its members, and its description can be seen by the public. To join the group, permission must first be gained from the administrator. It is private, in a way, in that the public (i.e., those who are not members of the group) cannot see what people in the group have posted.

8.5 Retention through social engagement

Studies clearly indicate that social engagement enhances retention (Greenhow et al., 2009), (Godwin-Jones, 2008), (Winke and Goertler, 2008), (Solomon and Schrum, 2007) referenced in (Blattner and Lomicka, 2012). Most of the students at Australian University enrolled in the programming topic come from different areas of study like civil engineering, mechanical engineering or computer engineering. Usually they do not know each other well enough to ask for help or collaboration with their studies. Studies by Teague and Roe show that collaboration helps in learning programming. Collaboration is only possible if students know each other well enough to discuss their topic-related problems(Teague and Roe, 2008). Moreover, the coupling of smart devices with social media and easy accessibility of the internet at universities has created a new form of isolation for students while being at university, despite being surrounded by other students. It is not uncommon to see students sitting next to each other and not communicating with each other at all. Most of them are so engrossed in their smartphones, communicating with their friends on social media, that they almost ignore the person sitting next to them. In this scenario this setting can be used to the advantage of students by incorporating Facebook/social media into their studies. In this manner, they can communicate with their peers and discuss the programming topic on social media. A study by Wankel and Blessinger has proved that social engagement can benefit retention and this can be achieved with the help of social media(Wankel and Blessinger, 2013), since it provides a digital form of social engagement (Walsh, 2012). “Social Networking Sites (SNS) used for academic purposes have shown positive results, as students interact outside of the classroom and therefore these SNSs assist in the learning process and building community” (Hung and Yuen, 2010). “Blending the real and virtual worlds inside and outside of the classroom has been shown to increase peer to peer and academic engagement, especially for first year students” (McCarthy, 2010).

Existing studies have discovered that a lack of engagement in learning programming is a key determinant of a student's poor performance. Therefore, it is beneficial to perceive a student's lack of engagement ahead of time, so that appropriate actions can be taken to re-engage her/him before she/he decides to give up. However, first year topics, especially programming topics, usually have very large enrolments, making it hard for a lecturer to keep track of each individual student's engagement level. Even though a lecturer endeavors to do that, it is often too late to take effective action after noticing a student has disengaged through their submitted work.

8.6 Use of a Course Management System

Course Management Systems (CMS) have been widely adopted by universities across the globe, including Australian University. A student's voluntary participation in a programming topic's discussion forum provided by the CMS was used as an indicator of their engagement in learning, as the lecturer can constantly monitor and re-engage those who present low or no engagement.

The benefit of this solution was that the CMS can easily be extended with the function that automatically generates continuous up-to-date reports on each individual.

8.7 Why use Facebook when universities have their own Course Management Systems like Moodle/Blackboard?

Most reputable Universities have their own student learning system. The system allows students to have access to the content related to the topic/subject of study. Discussion boards are available, where the lecturer can make announcements and students can initiate discussions and others can read their comments and reply. When a new discussion is initiated, the students get a notification either via their University email or when they log on to the student learning system. In either case, with a CMS, the student must log in to access the notification, whereas students get an automatic notification of a new post on Facebook. Moodle (Modular Object-Oriented Dynamic Learning Environment) is a free open-source Course Management System or **e-Learning** platform, which serves educators and learners across the globe. For Facebook, the communication is spontaneous, whereas for the Blackboard/Moodle learning systems, students have to check discussion boards(Stewart, 2012).

8.8 Analysis of the use of the CMS system and Facebook group as a tool to help improve student engagement and serve as an additional source of peer-to-peer interaction and social integration in the process of learning programming

The topic's CMS, named FLO (Flinders Learning Online), provided a discussion forum for students to communicate with their peers (mainly asking and answering questions). It has a feature named *general discussion* which can serve as an internal social media for the university, whereby all the students enrolled in a topic can communicate with each other. The students enrolled in each topic may initiate and respond to a discussion related to the topic, while the lecturer and the tutors also intervened whenever necessary by responding to students' posts. Students' participation in the discussion forum was completely voluntary.

At the start of the semester, the students were given written as well as verbal information about the *general discussion* option in FLO. No incentive was offered and no extra effort was made to motivate students to participate.

A closed Facebook Group for the programming topic for each of the semesters from 2013 to 2014 was created and the students were informed that: (a) the discussion forum on the Facebook Group was in parallel to that provided by the CMS, (b) the group would not be monitored or responded to by the lecturer/tutor, (c) they can use their existing Facebook accounts to sign in, and (d) no incentive was attached to their voluntary participation in the group.

In contrast with the Facebook group, which was primarily a peer-to-peer discussion channel, the students were informed that on FLO their chats will be monitored by the Lecturer and the Lecturer would also respond if required by the discussions.

This study was conducted across four semesters. For each semester the students were introduced to the Facebook group by the topic coordinator. Participation was voluntary. The performance of students who joined the Facebook group was measured in terms of scores. The students were also informed about the general discussion option in FLO. The performance of the students who participated in the general discussion on FLO was also measured in terms of scores.

A comparison of the use of the Facebook group and FLO was made. It was found that the students preferred to use FLO over Facebook.

8.8.1 Trials Conducted

8.8.1.1 First test with the Facebook group:

The first study with the Facebook group was performed by introducing a private closed Facebook group to the students studying programming in the first year of their tertiary education. It was introduced in the first semester of 2013. Since the study was approved by the ethics committee in April and the semester had already started on 4th March, it could only be introduced in the 7th week, just after the mid semester break from 15th April to 26th April. It was introduced by the topic coordinator during the lecture and a link was provided through which a request to join the group was sent to the administrator. After verifying that the request was sent by those students studying the programming topic, the students were allowed to join the group. In total 14 students joined the group.

8.8.1.2 Comparison with the Usage of FLO

In total 34 discussions were initiated on FLO by 18 students. There were 89 replies, out of which 25 replies were by the lecturer or tutors, leaving 64 replies by students. The total number of active student participants on FLO across the entire period of the study was 34.

8.8.1.3 Second test with the Facebook group

A Facebook group was created in the second semester of 2013. The group was introduced to the students in the second week of the semester. It was introduced to the students by the topic co-ordinator, who was also teaching the topic in semester 2, 2013.

A total of 204 students had enrolled in the topic and 144 students completed the topic. The rest of the students left the topic either at the beginning or in the middle of the semester. Out of the 144 students who continued to study the topic, 21 students joined the Facebook group. A few students who joined the Facebook group were active throughout the semester and others also posted content related to the topic but not regularly. At the end of the semester the scores of the students who joined the group were recorded for analysis. The total number of posts, number of comments, number of likes to the comments and the number of students who viewed the posts was recorded. The date of each post was also recorded.

8.8.2 Comparison with the use of FLO

A total of 58 discussions were posted by 38 students. There were 165 replies. Out of those 165 replies, 23 replies were posted by the Lecturer or tutors. The total number of active student participants was 46.

8.8.3 Third test with the Facebook Group

The third test with the Facebook Group was undertaken in the first semester of 2014. After seeking permission from the topic coordinator, the group was introduced to the students in the fourth week of the semester. In total 22 members joined the group. After the success of the use of the Facebook group in semester 2, 2013, it was expected that it would be successful again in Semester 1, 2014, but it turned out to be a failure as no student initiated any discussion in the group. The topic-coordinator reminded students during the lecture that there was a Facebook group available for additional support by peers but still no discussion was initiated.

8.8.4 Comparison with the use of FLO

A total of 40 discussions were posted by 29 students and a tutor. So, the number of discussions posted by students was 39. There were 97 replies. Out of those 97 replies, 23 replies were posted by the Lecturer or tutors. The number of active student participants was 39. In addition, 7 discussions were initiated by the Lecturer and tutors (7 by the lecturer and 1 by a tutor).

8.8.5 Fourth test with the Facebook group

Another group was introduced to the students in semester 2, 2014. This time the group was introduced to the students in the first lecture. A total number of 38 students joined the group, which was the greatest number out of the four rounds of the experiment being conducted. A discussion was initiated by a group member at the end of week 3. This discussion was regarding software and not a problem related to learning programming. None of the members of the group responded to the post initiated by the group member. After that, no discussion was initiated until week 6. So, the research was discussed with the topic-coordinator again and, after seeking his permission, the researcher decided to initiate a discussion by posting a few objective type questions in the group. Only one student replied to the first question: all other members of the group chose to remain silent. The question was then answered by the researcher with an explanation and the post was seen by 31 members of the group. A second question was not answered by any member of the group. It was answered by the researcher with an explanation and was seen by 28 members of the group. The third question was not answered by any member of the group. It was again answered by the researcher and the post was seen by 31 members of the group. The fourth question was answered by one member of the group. Its answer was posted by the researcher and the post was seen by 32 members of the group. The fifth question was not answered by any group member. The answer was again posted by the researcher and this post was seen by 33 students. The sixth question was answered by three members of the group. Again, the answer was given by the researcher and the post was seen by 33 members. This shows that even though most students refrained from answering the questions, they were actively looking at the posts. Overall the group was not successful, as no discussion related to the programming topic was initiated by the students in the group.

8.8.6 Comparison with the use of FLO

A total of 29 discussions were posted by 21 students. There were 82 replies. Out of those 82 replies, 17 replies were posted by the Lecturer or tutors. The number of active student participants was 29. Also 2 discussions were initiated by the Lecturer and tutors (1 by the lecturer and 1 by a tutor).

8.9 Results

8.9.1 Facebook Study Analysis

8.9.1.1 Results from the first test of the Facebook group

No activity took place within the group. So, nothing could be studied, as no group activity took place. To study if the introduction of the group at the beginning of the semester would prove to be helpful, another group was formed by the administrator (researcher) in the second semester of 2013.

The total number of students enrolled in the topic was 196. Out of 196 students, 52 failed the topic. The failure rate was 26.53%.

8.9.1.2 Results from the second test of the Facebook group

The discussions related to the topic were initiated throughout the semester. There was no restriction of the time at which a post could be initiated so the posts were initiated at various times. Some discussions at unusual times, including late at night, midnight or early in the morning were recorded. This suggests that Facebook can be a 24/7 classroom. Students who were stuck with a problem asked for help at any time of the day or night. Problems ranging from simple to complex were discussed. Most of the discussions in the posts were regarding the practical work. The most popular post had 21 comments and 2 likes. The active members tried to solve a practical problem faced by the student who initiated the post. A YouTube link was also posted as a solution to the problem. The time of initiation of the post could not be recorded but after the first reply, the discussion continued for almost 1 hour and 15 minutes. The most popular post, which involved 3 participants, discussed the problem of drawing charts/graphs in their practical work. Study of the posts suggests Facebook was mostly used to support the participants' practical work. On average, there were 2.4 likes per post.

Another interesting observation was that although only very few students initiated the posts, every time a new post was initiated it was viewed by all the members of the Facebook group. Thus, students continued to watch the group discussions, even if they were not actively involved: they still obtained some information related to the topic, which may have proved useful to them.

By collecting more data with a larger sample size, the results can be further analysed as to whether or not a Facebook group can contribute as an additional learning resource for the programming students.

More interestingly, out of 21 students, 4 were regular, frequent posters, who replied actively to other students' posts. All of them obtained very good scores in the examination. A key question is whether they got involved because they were high performing students or did the Facebook group contribute in some way to their learning? Some students were not regular posters but would occasionally post or reply to a problem/post in the group. This further confirms that though the students may not be actively participating by posting regularly, they still go through each and every post with interest and do not hesitate to provide a solution when required.

Table 7-1 shows the Frequency of Posts and Corresponding Scores of Students. It suggests that seven students out of 21 posted at least once, which suggests that they were active in the group. Most of the students who posted in the group passed the topic. Also, the frequency of posting had a direct impact on scores, as shown in Table 7-1.

Table 8-1: Table with Frequency of Posts and Corresponding Scores of Students

Student Number	Scores	Frequency of posts
stu1	90	13
stu2	75	8
stu3	72	7
stu4	26	1
stu5	58	1
stu6	68	2
stu7	69	1

It has been observed that the students who posted regularly on Facebook scored better than the students who seldom posted. It cannot be definitely concluded that the students who joined the group and actively participated performed better than the students who did not join the group, since the study was undertaken on a small scale.

The results are consistent with the results obtained by other researchers on the use of Facebook group in learning. In a study conducted by Wu and Hsu, participants described their Facebook group as "a pressure-free environment for English learning because it is a virtual community composed of a closed group, which opens for limited members and makes them feel less stressful" (Wu and Hsu, 2011).

Another study on integrating Facebook with peer assessment and blended learning indicated that Facebook had a constructive impact in an ESL writing course (Shih, 2011). In another study, Ross et al. reported that using the Facebook tools increases students' motivation(Ross et al., 2009).

There were 204 students enrolled in the topic. 94 students failed the topic. The failure rate was 46.07%.

8.9.1.3 Results from the third test of the Facebook group

The third test with the Facebook group was undertaken in the first semester of 2014. After seeking permission from the topic co-ordinator, the group was introduced to the students in the fourth week of the semester. In total, 22 members joined the group. After the successful use of the Facebook group in semester 2, 2013, it was expected that it would be successful again in Semester 1, 2014, but it turned out to be a complete failure, as no student initiated any discussion in the group. The topic-coordinator reminded students during lectures that there was a Facebook group available for additional support by peers, but still no discussion was initiated.

8.9.1.4 Results from the fourth test of the Facebook group

The students were introduced to another Facebook group in semester 2, 2014. This time the group was introduced to the students in the first lecture. A total number of 38 students joined the group, which was the maximum number who joined throughout the research period. A discussion was initiated by a group member at the end of week 3. This discussion was regarding software and not a problem related to learning programming. None of the members of the group responded to the post initiated by the group member. After that no discussion was initiated until week 6. So, the research was discussed with the topic-coordinator again and, after seeking his permission, the researcher decided to initiate a discussion by posting a few objective type questions in the group. Only one student replied to the first question and other members of the group chose to remain silent. The question was then answered by the researcher with explanation and the post was seen by 31 members of the group. A second question was not answered by any member of the group. It was answered by the researcher with explanation and was seen by 28 members of the group. The third question was not answered by any member of the group. It was again answered by the researcher and the post was seen by 31members of the group. The fourth question was answered by one member of the group. Its answer was posted by the researcher and the post was seen by 32 members of the group. The fifth question was not answered by any group member. The answer was again posted by the researcher and this post was seen by 33 students. The sixth question was answered by three members of the group. Again, the answer was given by the researcher and the post was seen by 33 members. This shows that even though most students refrained from answering the questions, they were looking at the posts with interest. Overall the group was not successful as no discussion related to the programming topic was initiated in the group.

8.9.2 FLO Study Analysis

8.9.2.1 Semester postings on FLO

Table 5 lists the semester postings on FLO. It lists the number of students in each semester who initiated posts on FLO, the number of posts initiated by those students, the number of replies by students active on FLO and the number of replies by the Lecturer/Tutor.

Table 8-2 : Semester postings on FLO

Semester	No. of Students	No. of posts initiated	No. of replies by students	No. of replies by the Lecturer/Tutor
2013 Semester 1	34	34	65	25
2013 Semester 2	46	55	155	23
2014 Semester 1	30	40	98	23
2014 Semester 2	29	27	36	17

8.9.2.2 Variables Used and analysis methods

The following data were recorded for students active on FLO:

- Total scores of the students obtained in the final examination.

- Number of times a new post was initiated on FLO by the student. (FLO is a Student learning system of the University where all students studying the topic can communicate with each other and with the lecturer, tutors and demonstrators.)

- Number of times the student responded to the posts on FLO.

- Total number of posts (Number of times a new post was initiated on FLO + the number of times the students responded to the posts on FLO) or the number of times the students were active on FLO.

- Final examination grades were also recorded for students not active on FLO.

Data were available for 4 semesters, including 2013 semester 1, 2013 semester 2, 2014 semester 1, and 2014 semester 2.

Fisher's exact tests (Agresti, 2002) were used to determine if there was a relationship between the students' final examination grades and use of FLO (active on FLO vs. not active on FLO). Spearman's rank correlation coefficients as described by(Hollander and Wolfe, 1999) were used to determine if there were a relationship between the total number of posts and final examination grades for students active on FLO. A p-value less than 0.05 indicated significance. Data were analyzed for each semester independently and for all semesters combined. All data analysis was conducted using SPSS version 23.

8.9.2.3 Analysis of the results and interpretations

The discussions initiated on FLO were studied in detail and the posts which were not directly related to the topic were removed from the data. These covered issues such as students enquiring about lost textbooks, lost chargers and when their grades will be up on FLO. After this data was removed, the data analysis was done again for all four semesters.

Table 7-3 shows the frequency table of final examination grades after use of FLO for the 4 semesters (all semesters combined and by each semester). The total number of students enrolled in the class for the 4 semesters was 798. 17% of students were active on FLO and 31% of students failed the class. The results of Fisher's exact test suggested that with all data of all 4 semesters combined, there was a statistically significant relationship between final examination grades and use of FLO ($p = 0.000$). Students not active on FLO were more likely to fail the final examination than students active on FLO (34% vs. 18% with failed grades).

For 2013 semester 1, the total number of students enrolled in the class was 196. 13% of students were active on FLO and 26% of students failed the class. The results of Fisher's exact test suggested that for 2013 semester 1, there was no statistically significant relationship between final examination grades and use of FLO ($p = 0.343$).

For 2013 semester 2, the total number of students enrolled in the class was 204. 22% of students were active on FLO and 46% of students failed the class. The results of Fisher's exact test suggested that for 2013 semester 2, there was a statistically significant relationship between final examination grades and use of FLO (p = 0.000). Students not active on FLO were more likely to fail the final examination than students active on FLO (53% vs. 20% with failed grades).

For 2014 semester 1, the total number of students enrolled in the class was 206. 18% of students were active on FLO and 24% of students failed the class. The results of Fisher's exact test suggested that for 2014 semester 1, there was a statistically significant relationship between final examination grades and use of FLO (p = 0.000). Students not active on FLO were more likely to fail the final examination than students active on FLO (28% vs. 3% with failed grades).

For 2014 semester 2, the total number of students enrolled in the class was 192. 15% of students were active on FLO and 29% of students failed the class. The results of Fisher's exact test suggested that for 2014 semester 2, there was no statistically significant relationship between final examination grades and use of FLO (p = 0.382).

Table 8-3: Two-way frequency table of final examination grades vs. use of FLO

		Grades		
		Failed	Not failed	Total
All 4 semesters	Active on FLO	25 (18%)	111 (82%)	136 (17%)
	Not active on FLO	225 (34%)	437 (66%)	662 (83%)
	Total	250 (31%)	548 (69%)	798
2013 semester 1	Active on FLO	9 (35%)	17 (65%)	26 (13%)
	Not active on FLO	43 (25%)	127 (75%)	170 (87%)
	Total	52 (26%)	144 (74%)	196
2013 semester 2	Active on FLO	9 (20%)	36 (80%)	45 (22%)
	Not active on FLO	85 (53%)	74 (47%)	159 (78%)
	Total	94 (46%)	110 (54%)	204
2014 semester 1	Active on FLO	1 (3%)	36 (97%)	37 (18%)
	Not active on FLO	48 (28%)	121 (72%)	169 (82%)
	Total	49 (24%)	157 (76%)	206
2014 semester 2	Active on FLO	6 (21%)	22 (79%)	28 (15%)
	Not active on FLO	49 (30%)	115 (70%)	164 (85%)

	Total	55 (29%)	137 (71%)	192

Table 7-4 shows the descriptive statistics of final examination scores and numbers of posts on FLO for students active on FLO, for all 4 semesters combined and by semester. Table 7-8 shows the results of the Spearman's correlation coefficients of final examination scores and number of posts on FLO for students active on FLO for all 4 semesters combined and by semester. Figure 7-2 shows the scatter plot of final examination scores and number of posts on FLO, for all 4 semesters. When data of all 4 semesters were combined, there was a statistically significant relationship between final examination scores and numbers of posts on FLO for students active on FLO (Spearman's rho = 0.263, p = 0.002). The positive Spearman's correlation coefficient (0.263) suggested that there was a positive relationship between final examination scores and number of posts on FLO for students active on FLO. In other words, final examination scores would increase when numbers of posts on FLO increased, and vice versa.

There was also a positive relationship between final examination scores and numbers of posts on FLO for students active on FLO for 2013 semester 1 (Spearman's rho = 0.390, p = 0.049) and for 2013 semester 2 (Spearman's rho = 0.304, p = 0.042). There was no statistically significant relationship between final examination scores and numbers of posts on FLO for students active on FLO for 2014 semesters 1 and 2.

Table 8-4: Descriptive statistics of final examination scores and numbers of posts on FLO for students active on FLO

		Mean (SD)	Min	Max
All 4 semesters	Final examination scores	66.77 (25.28)	2	100
	Number of posts on FLO	3.16 (4.01)	1	25
2013 semester 1	Final examination scores	58.23 (28.83)	2	97
	Number of posts on FLO	2.92 (4.08)	1	19
2013 semester 2	Final examination scores	63.78 (25.01)	4	99
	Number of posts on FLO	4.09 (5.17)	1	25
2014 semester 1	Final examination scores	81.08 (21.15)	6	100

	Number of posts on FLO	3.32 (3.48)	1	17
2014 semester 2	Final examination scores	60.61 (20.14)	9	95
	Number of posts on FLO	1.68 (1.16)	1	5

Table 8-5: The results of the Spearman's correlation coefficients of final examination scores and numbers of posts on FLO for students active on FLO for all 4 semesters combined and by semester

	Spearman's rho
All 4 semesters	0.263 (0.002)*
2013 semester 1	0.390 (0.049)*
2013 semester 2	0.304 (0.042)*
2014 semester 1	0.052 (0.760)
2014 semester 2	0.267 (0.169)

(Note: numbers in parentheses are p-values. * indicates significance at the 0.05 level)

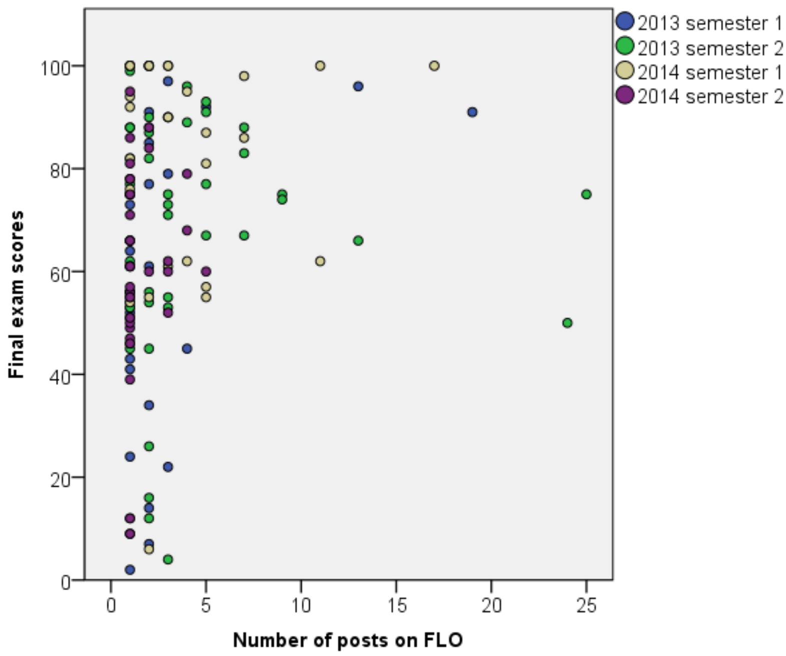


Figure 8-1: Scatter plot of final examination scores and numbers of posts on FLO

8.10 Summary

This chapter analysed the use of Social media, i.e. Facebook and the CMS system, to help improve student engagement and serve as an additional source of peer-to-peer interaction and social integration in the process of learning programming.

Its use was also compared with the use of the Facebook group. The trials that were conducted across four semesters and the students' results across four semesters were studied. It was observed that both CMS and the Facebook group help improve student engagement and serve as an additional source of peer-to-peer interaction and social integration in the process of learning programming but the students preferred to use CMS over the Facebook group. The various reasons for this choice were studied and it was found that, amongst other reasons, the most obvious reason was that Lecturer and tutor involvement in the FLO discussion group was high, whereas the Facebook group was a peer-to-peer only learning group. So, the students may have found the discussions and solutions on FLO more trustworthy due to lecturer and tutor involvement, when compared with discussions and solutions on the Facebook group.

CHAPTER 9 : DISCUSSION OF RESULTS

This chapter examines in detail those factors which affect student performance at both universities.

9.1 Hypothesis to be tested

9.1.1 Main Null Hypothesis: factors and attributes that affect the learning of programming in Australia and India are not same.

9.1.1.1 Similarities between the two universities

The students at both universities were asked if they had previously studied programming at school, home or university. It was observed that similar numbers of students had studied programming prior to the formal course. For Australian University, 13% had studied programming in 9th or 10th grade and at Indian University, 15.2% had similarly studied programming. For Australian University, 20.1% of students had studied programming in 11th or 12th grade. Likewise, 22.8% had done so at Indian University. Interestingly, for Australian University, 26.1% of students had studied programming at home or completed individual study at their own instigation, as had some 30.4% at Indian University. There was thus a considerable difference in the level of programming studied before starting tertiary level work between the two universities.

The students were asked about the type of study they had completed before attending laboratory sessions and some similarities were observed for the two universities. At Australian University, 46.7% chose to study textbook slides related to the laboratory, whilst some 50.6% of students chose to do the same at Indian University. The percentages of students who preferred to read the previous laboratory work material before going to the laboratory were almost identical at 47.3% and 48.1% for Australian University and Indian University respectively. At Australian University 33.2% of students were involved in practicing new, similar programs related to the laboratory and 35.4% did the same at Indian University.

The students were asked about their preferences about completing laboratory work. It was found that 78.8% students at Australian University and 75.9% students at Indian University preferred to do laboratory work in a laboratory.

The students were asked about their family history of attending university. At both universities, the students had similar family backgrounds. At Australian University 38.3% were the first one in the family to attend a university and at Indian University the figure was 34.2%. For both universities, the percentage of students whose parents/carers had attended university was approximately the same i.e. 50.3% and 50.6%. The percentage of students whose siblings had attended University was also found to be comparable for both Australian University (50.6%) and Indian University (44.8%). The students were asked about their reasons for choosing to study programming. Interestingly the value of mean(SD) from the statistical analysis suggested that the students had two reasons in common to choose to study programming; one being that it was a mandatory course for both Australian University (0.77) and Indian University (0.78); the other being the potential to achieve high paying work in the industry after graduation. The figures were: Australian University (0.67) and Indian University (0.67).

9.2 A Hypotheses Summary

Table 8-1 represents the Hypothesis Summary. The following hypotheses were accepted after the statistical analysis of the data, based on the research questions shared with 198 participants from Australian University, Australia, and 94 participants from Indian University, India.

Hypothesis Number	Null Hypothesis	Accept/Reject Australian University	Accept/Reject Indian University

1	Gender of students does not affect the performance of students.	Accept	Reject
2	Studying programming anywhere or anyhow does not affect the scores of the students	Reject	Accept
3	Prior study of any of the above-mentioned languages does not affect the performance/scores of the students	Accept	Accept
4	Studying flowcharts or algorithms does not affect the performance/scores of the students	Accept	Accept
5	Posting questions/asking for help on Google, Twitter, Facebook, and Email was not helpful to the students	Accept	Accept
6	Reasons to study programming do not affect the scores of students	Reject	Accept
7	Preliminary preparation was not helpful in improving performance of the students	Accept	Accept
8	The students who do preliminary preparation	Accept	Accept

	before laboratory did not perform better than those who did not.		
9	Revision does not result in better performance and thus obtaining good scores	Reject	Accept
10	One kind of revision was not helpful over the other kind of revision in terms of scores obtained by the students	Reject	Reject
11	Seeking help does not prove to be helpful in terms of scores obtained	Reject	Accept
12	Seeking help from any source does not prove to be helpful in terms of scores obtained	Reject	Reject
13	Attending a higher percentage of lectures does not lead to better performance in terms of scores obtained.	Reject	Accept
14	The students who view lectures online did not obtain better scores than those who did not (Australian University only)	Accept	Not Applicable

15	The students who performed a particular activity in the lecture theatre did not get better scores than those who did not	Accept	Accept
16	Practising laboratory work at a particular venue does not help to improve performance of students in terms of scores	Reject	Reject
17	The students who found attending labs useful performed better than those who did not.	Reject	Accept
18	The students whose parents/carers or siblings had attended university did not perform better than those whose parents/carers or siblings did not attend university	Reject	Accept
19	The students whose home environment was conducive to study did not perform better than those who did not have home environment conducive to study	Reject	Accept
20	The students who can get programming-related help	Accept	Accept

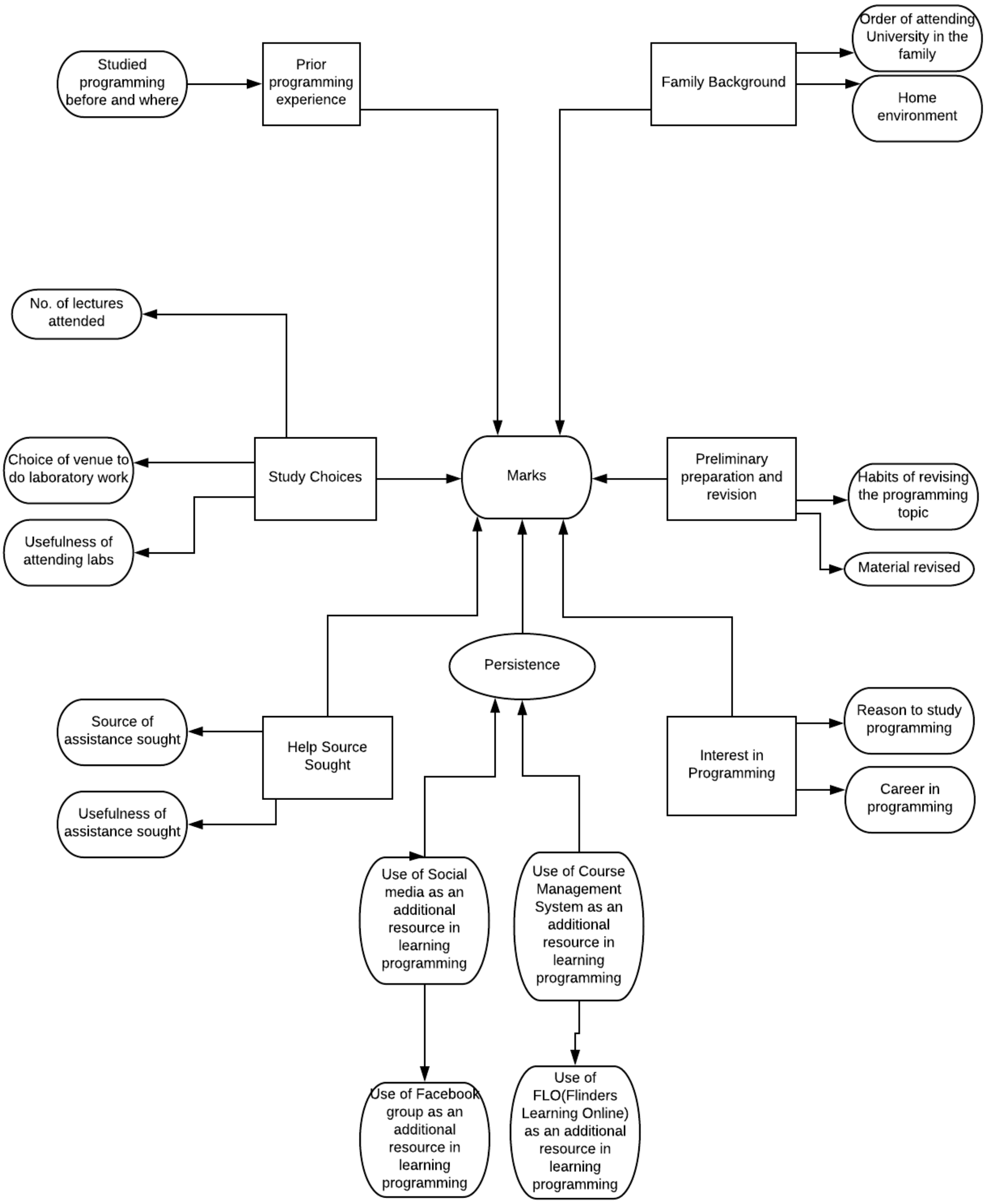
	at home from their parents/carers or siblings do not perform better than those who do not		
21	The students whose parents/carers were supportive of their educational goals did not perform better than those who were not supportive	Accept	Accept
22	The students who were studying this topic for second or more time did not perform better than those who were studying it for the first time	Accept	Not Applicable
23	The students who chose to study a topic related to programming again were not better performers	Accept	Reject
24(a)	The students who would like to take up a career in programming performed better than those who do not (Australian University only)	Reject	Not Applicable
24(b)	Those students who would like to take up a career in programming do not perform better than those who would like to take up	Not Applicable	Accept

	another career like testing, technical writer, graphic designer (Indian University only)		

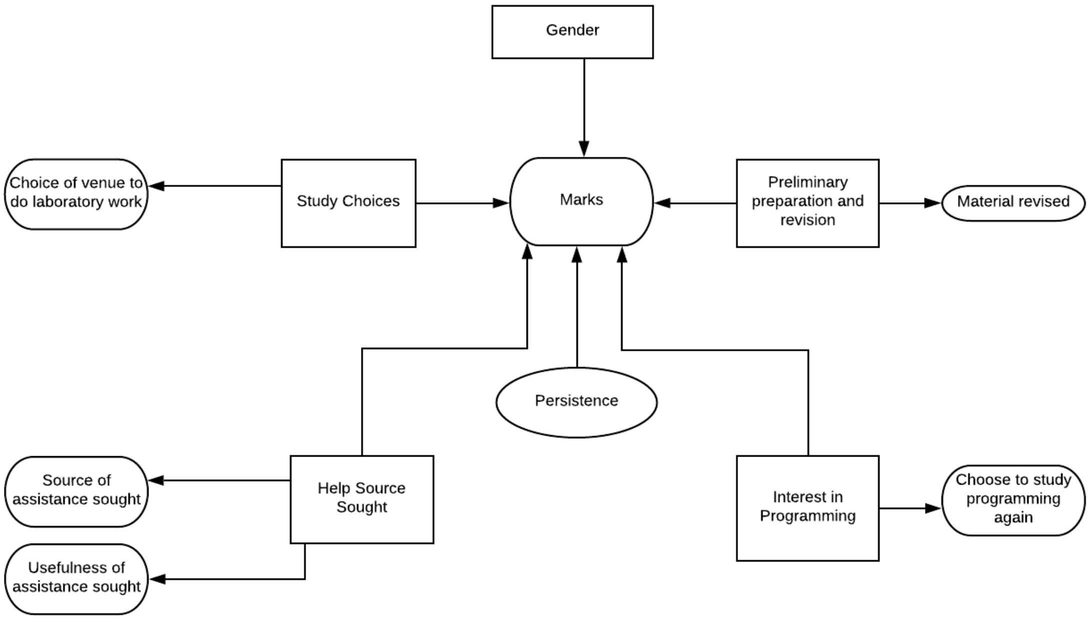
Table 8-1: Hypothesis Summary

9.2.1 Summary of Inferential Statistics

9.2.1.1 Research Model Australia



9.2.1.2 Research Model India



9.3 Answers to research questions

To answer RQ*1: "Does gender affect students' performance"*, the data analysis results of this study suggested that the gender of the students affected the performance of students at Indian University but not at Australian University.

The following discussion is to answer RQ2: "Do students' habits surrounding preliminary preparation and revision affect student performance in terms of scores", and its sub-questions:

1. Please suggest the frequency if you study before going to the programming Lecture?
2. What do you study before going to the Laboratory?
3. Please tell us about your habits for revising the programming topic.
4. If you happen to revise the programming topic, please let us know if you revise the following?

The data analysis results suggested that preliminary preparation had no statistically significant effect on the performance of students from both universities. Revision had a statistically significant effect on student performance of students at Australian University. The students who consistently revised while the semester was in progress had statistically significantly higher examination scores than students who never or less frequently revised while the semester was in progress. Revision did not have any effect on the performance of Indian University students. The “kind of revision” done had a statistically significant effect on the performance of students. For Australian University, there was a statistically significant difference in examination scores among students with the habit of revising on a website specifically designed to revise the topic. In particular, the results of pairwise comparisons suggested that students who revised on such a website “very often” had statistically significantly higher examination scores than students “sometimes revising” on such a website.

For Indian University, the analysis of the results suggested that there was a statistically significant difference in examination scores among students with different habits of revising programming topics, such as using theory from lecture slides given by the lecturer. In particular, the results of pairwise comparisons suggested that students who “never revised” programming topics in this way had statistically significantly higher examination scores than students who “sometimes revised” like this.

The following discussion is to answer RQ3: “*Do students’ study choices affect student performance*”, and its sub-questions:

1. What do you do in the programming Lecture theatre? (You can tick more than one box)
2. Please write the approximate number of programming lectures you have attended
3. How often do you view the programming lectures online?
4. Where do you prefer to do the laboratory work?
5. Do you find attending laboratories useful?

The data analysis of the results suggested that attendance had a positive effect on the performance of students at Australian University. There was a statistically significant difference (higher) in examination scores among students attending different numbers of lectures ($p = 0.002$). In particular, according to the results of pairwise comparisons, students who attended 100% of the lectures had statistically significantly higher examination scores than those students who attended up to 80% of the lectures. The attendance numbers had no effect on the performance of students at Indian University. Viewing lectures online had no effect on the performance of Australian University students: this provision was not available at Indian University. No particular activity performed in the lecture theatre made any statistically significant effect on the final performance of students. The choice of venue for laboratory work had a statistically significant effect on the performance of students for both universities. For Australian University, the results suggested that students who preferred to do laboratory work at home, would have statistically significantly higher examination scores than students who chose to remain in the laboratory or library. For Indian University, students who preferred to do laboratory work in the library had statistically significantly lower examination scores than students who did not.

For Australian University, there was a statistically significant difference in examination scores among students with different opinions regarding the usefulness of attending laboratories. In particular, students who thought attending labs was not at all useful had statistically significantly lower examination scores than students who thought attending laboratories was either slightly or very useful. For Indian University students, there was no statistically significant difference in examination scores across any category of attitude regarding attending the laboratory.

To answer RQ4: **"**What kind and source of help is sought by the students when required and what source of help proves to be useful", and its sub-questions:

1. If you posted questions related to the topic/course/subject online on social networking or other websites to obtain help, was the help useful or not?
2. From what sources do you try to seek help?

3. Was seeking help from these sources useful?

   Posting questions online had no effect on student performance at either university.

   Seeking help did not affect the scores of students for either university. For Australian University, there was a statistically significant difference in examination scores among students with different opinions regarding the importance of connecting with classmates. In particular, students who thought classmates were never helpful had statistically significantly higher examination scores than the students who thought classmates were sometimes useful. This suggests that the students at Australian University like to work in isolation and do not regard classmates as a useful source of help.

   The help sources sought by students had a significant effect on the performance of students of both the Universities. At Australian University the results of pairwise comparisons suggested that students who regarded "textbooks always useful" had statistically significantly higher examination scores than students who "regarded textbooks not useful". Whereas, at Indian University, there was a statistically significant difference in examination scores among students with different opinions of the usefulness of "I opt to study the topics at training institutes teaching similar courses". In particular, the results of pairwise comparisons suggested that students who did not think it was useful had statistically significantly lower examination scores.

   The following discussion is to answer RQ5: "*Does intrinsic interest in programming leads to better performance*", and its sub-questions:

1. Why did you choose to study programming? (You can tick more than one box)

2. If given the option, would you choose to study a topic/subject related to programming again? (i.e. a topic other than what you have already studied)

3. Would you like to take a job in or make a career out of programming?

   The data analysis results suggested that the reason for studying programming had a significant effect on the performance of students of Australian University, as the statistical analysis results suggested that the students who regarded “interested to know about programming” as the “most important” reason to study programming had better examination grades than those who regarded “interested to know about programming” as “not important”. The reason for studying programming had no effect on the performance of Indian University students.

Those who would choose to study a programming topic again tended to do better than those who would not at Australian University. For Australian University, the results suggested that students who would choose to study a topic/subject related to programming again would have statistically significantly higher examination scores than students who would not make that choice. There was no statistically significant difference in examination scores for Indian University based on this study choice. The students’ choice to take up programming as a career had some positive effect on the performance of Australian University students. For Australian University, the results suggested that students who would like to take a job in or make a career out of programming would have statistically significantly higher examination scores than those who would not. There was no statistically significant difference in examination scores across any category of attitude regarding a future career.

The following discussion is to answer RQ6: “Does prior programming experience proves to be helpful in learning programming leading to better performance”, and its sub-questions:

1. Did you study programming before attending this course?
2. Have you studied any of the below mentioned languages?
3. Have you studied designing Flowcharts/Algorithms?

Prior programming experience had some effect on student performance, as the students who studied programming at home before attending the course had statistically significantly higher examination scores than students who had studied programming at other levels. Conversely, prior programming experience had no effect on students’ performance at Indian University. Studying a programming language prior to studying the topic had no effect on the performance of students at either university. Also, prior knowledge of flowcharts and algorithms had no statistically significant effect on the performance of students at either university.

The following discussion is to answer RQ7: *“Does student family background affect student performance?”*, and its sub-questions:

1. Are you the first one from your family to attend University, or do you have other family members who have attended University?

2. Is your home environment conducive to study?

3. Can you get programming related help at home from your parents/carers or siblings?

4. Are your parents/carers supportive of your educational goals?

Statistical analysis of the results suggested that the family background of the students, in terms of educational background, had an effect on the performance of students. For Australian University, the results suggested that those students who were the first one in the family to attend university would have statistically significantly lower examination scores than students whose parents/carers or siblings had attended University. The parameter of the home environment had a positive effect on the performance of students of Australian University. Based on the results, it was concluded that the students who believed that their home environment is conducive to study would have statistically significantly higher examination scores than those who did not. For Indian University, there was no statistically significant difference in examination scores across any category of home environment. The availability of programming related help had no effect on the performance of the students at either university. For both universities, the results indicated that there was no statistically significant difference in examination scores whether programming related help was available at home or not. The attitude of parents/carers towards the educational goals of students did not have any effect on the performance of students. The results suggested that there was no statistically significant difference in examination scores whether the parents/carers were supportive of their educational goals or not. The frequency of studying the topic had no effect on the performance of students. It was therefore concluded that there was no statistically significant difference in examination scores for students who were studying the programming topic in terms of the number of times it was taken.

9.3.1 Discussion of results for factors affecting learning programming

The differences in the results of the two universities may be due to their differences in educational culture. The learning culture at a Australian university is different from that at an Indian university. Students follow different codes of conduct; at Australian universities, students are allowed to bring and use their portable/mobile devices in the lecture theatre: this is prohibited in Indian universities. The literature suggests that collaborative learning enhances the learning experience (Teague and Roe, 2008). At Australian universities, there is provision for residence on campus, but most students choose to live either with their carers or off campus. As students live off campus and meet each other only during the lectures, laboratory sessions or workshops, the opportunities to collaborate with each other for study purposes are formal and limited. In contrast, at Indian universities, most of the students stay in university-organised hostels. As a result, there are opportunities to collaborate informally and it is common for students to support each other when one asks for help.

Another cause of the difference in the results between the two universities may be the difference in the examination structure, examination frequency, assessment structure and numbers of lectures per week. Further research needs to be conducted to explore the detailed reasons for the differences found to date.

9.3.2 Discussion of the interrelationships between factors

After analysing the relationships across various factors, it was interesting to discover some similarities between the two universities, despite the differences between teaching approaches, assessment and examination structure. This suggests that students' responses to learning programming was similar overall. These results can be used to modify and adapt approaches to teaching and learning in both institutions to obviate issues and support the students' further.

There is a significant relationship between the frequency of revision of programming topics and the frequency of studying prior to the programming lectures at both universities.

For Australian University, the parameter difference was 0.539, indicating that for a one-unit increase of frequency in studying before going to the programming lecture, the frequency of programming topics revised would also increase by a unit of 0.539.

Similarly for Indian University, the parameter difference was 0.355, indicating that for a one-unit increase of frequency in studying before going to the programming lecture, the frequency of programming topics revised would also increase by a unit of 0.355.

There was a significant relationship between undertaking preliminary preparation before laboratories and the frequency of studying before going to the programming lecture at both universities. This suggests that there exists an interrelationship across the learning factors. Thus, focusing on both direct and indirect factors may help students improve their learning of programming and hence improve their overall scores.

To answer RQ8(Research Question 8): “Can social media, i.e. Facebook or the CMS system, help to improve student engagement and serve as an additional source of peer-to peer-interaction and social integration in the process of learning programming? a number of questions were asked.

The results from this study have positively tested our hypothesis that students’ voluntary participation in the topic’s forum provided through the CMS can enhance student engagement and serve as an additional source of peer-to-peer interaction and social integration in the process of learning programming, as there is a positive correlation between their level of participation in terms of the number of posts and their awarded grades/scores. However, the same results could not be proved for the use of a Facebook group, due to the small sample size of the group who chose to take part in this aspect of the research.

The results suggest that, when compared with Facebook, CMS proved to be the tool that the students preferred.

9.3.3 Discussion of the Facebook Group and CMS

The results of the first, third and fourth tests were in contrast with the second test of the Facebook group study as so few students joined the group. Of those who joined, very few actively participated in the group discussions. In the first (Semester 1, 2013), third (Semester 1, 2014) and fourth (Semester 2, 2014) semesters of the study, almost no student initiated a post/discussion. A few students actively participated in the second study (Semester 2, 2013).

A questionnaire was conducted where the students were asked the reason for not joining or participating in the Facebook group. Very few students responded but the responses were similar. Most students suggested that since their identity was disclosed and the students knew the name of the person asking the question, they had a fear of being judged by their peers. One of the students responded "The people that joined seemed far more advanced in the topic than I was (based on their results they shared). It made it too embarrassing to ask questions or discuss things as I viewed the situation as them not caring enough to want to talk or help when they are likely busy doing other things." Another student replied similarly saying that it "sometimes felt that if I did not understand a basic concept I may be judged". So, the students were asked if they knew the names of the students sitting next to them in the lecture theatre or laboratory. The question was asked to remove the fear of being judged by someone who may not even identify the name of the person asking a question or responding on a Facebook group. Most of them did not know the names of the students with whom they were studying. The fear of being judged by other students may be one of the factors to contribute to the low or no student responses.

Prior studies demonstrating the successful use of Facebook involved the researcher directly communicating with the students. In most of the cases, the researcher was the lecturer teaching the topic. In this case, however, the researcher was not directly involved in the experiment. Also, it was conducted in a different manner, as it was proposed as a peer-to-peer learning group. The Lecturer was not involved in this study except for introducing the group to the students. Similar earlier experiments involved lecturer intervention and monitoring. This may be another factor that may have contributed to the low or no student involvement in the experiment.

It was observed that if discussions were initiated in the Facebook group by some students, then the other students also started participating by posting questions. This activity was observed in the second test of the Facebook group, where a student initiated a discussion at the start of the semester and the other students not only viewed the post but also started posting their questions on Facebook and participated in the discussions as well. After the success of the second trial of the Facebook group, the third trial did not work as expected, as no student initiated a discussion. The researcher tried to initiate a discussion by posting questions on the Facebook group but to no avail. This suggests that the students only wished to participate in those discussions initiated by their peers.

Another interesting observation was that the students freely initiated posts on Facebook at any time of day or night, outside lecture and laboratory hours. This was noted from the “seen by” feature of the Facebook group. This feature shows the ‘number of’ group members who viewed the post and ‘the time’’ at which they viewed the post. Some posts were initiated late at night or early in the morning which shows that students may seek help from their peers or Lecturers when required. The additional benefit of communicating on FLO is that the students may get response at odd times as well. In this study the students responded to the posts at odd hours as well.

9.3.4 Discussion of CMS(FLO)

It was observed, based on the results of the students’ responses to FLO, that students preferred to communicate with each other through FLO rather than Facebook. Across all four study semesters, the responses on FLO were noteworthy. A moderate but consistent number of students-initiated posts on FLO and a moderate number of students responded to the posts. The students discussed varied numbers of issues related to the programming topic. Some students posted questions related to the problems they were facing while trying to solve a particular assignment. The response was remarkable, and students brainstormed together to get answers to their questions. An important factor was that the students did not provide complete solutions for each as that would constitute plagiarism. They discussed their problems and worked towards solutions, supported by suggestions from fellow students as well as the lecturer and tutors. The use of FLO may have also contributed to the low response to the Facebook group, as the students were already posting their queries on FLO as they did in their other topics.

It was obvious from the responses that the use of FLO was more successful than that of Facebook. Another factor that may have contributed to the result is that there was no lecturer involvement in the Facebook group. It was presented as a peer-to-peer discussion group, whereas the FLO discussion group also included lecturer involvement. At the start of the semester, the students were informed that their discussions would be monitored and responded to by the lecturer, as well as the tutors and demonstrators teaching the course. The likelihood of the answers being accurate was therefore high. This may have motivated the students to participate in the discussions on FLO more actively.

Based on the results, it was observed that throughout each semester, the lecturer and tutors responded to student queries. The answers were provided out of the class and laboratory hours by the lecturers/tutors making FLO an efficient and convenient learning space. This kind of support may have helped students in their learning. The students may either view the discussions passively or actively participate. The discussions took place during at any time of day, including early in the morning or late at night, indicating that FLO may serve as an important, convenient and dependable peer to peer interaction and social integration resource for learning programming.

Thus, FLO proved to be a virtual classroom outside the classroom. Interestingly, the number of questions asked on FLO was greater than the number of questions being asked during a lecture. Usually the curriculum of a programming topic is so large that each week the lecturer must complete a certain amount of coursework during the lecture. This generally does not leave much time for student questions. Thus, this problem may be solved by tapping the potential of the CMS system by using its discussion feature, making the CMS system a form of virtual classroom

After modification of the data, the results from the statistical analysis also suggest that, with all the data for the four semesters combined, there was a statistically significant relationship between *final examination grades* and *use of FLO*. Students not active on FLO were more likely to fail in the final examination than those who were active.

After the modification of data when data of all four semesters was combined, there was a statistically significant relationship between *final examination scores* and *number of posts on FLO* for students active on FLO. In other words, final examination scores would increase when number of posts on FLO increased, and vice versa.

Also, it was observed that the students discussed various problems on FLO and worked towards the solution. Unlike the Facebook group where the students tried to solve problems on their own or with the help of other students but not the lecturer and tutors, on FLO, the lecturer and tutors stepped in to facilitate solutions. This clearly shows that FLO acts as a classroom outside classroom with support from the lecturer as well as tutors.

It may be concluded that those students who were not actively participating on FLO were more likely to fail in the final examination than those students who were actively participating. Also, the final examination scores were found to be improved when the number of posts on FLO increased, and vice versa.

9.3.5 Theoretical and practical impact of the study

The study conducted can help structure the course material for programming course for both countries based on the results achieved through the hypothesis. The hypothesis that show an impact on student performance can be built into the course structure so that positive outcome can be achieved. For example, in revision done impacts the performance of Australian students so the course structure can be formed with revision as a component. The lectures can include a questionnaire or problems based on previous week's work so that students revise the topic. It can also be given a component in the assessment of students so that students need to revise the topic. In both the Universities one kind of Revision was helpful over the other kind of revision in terms of scores obtained by the students, so based on the kind of revision that impacts results it can be built up into the course structure. The students can be encouraged to use the help source that impact results positively. The attendance can be enforced at Australian University as attending higher number of lectures has positive impact on their marks. Also, at both Australian University and Indian University, practising laboratory work at a particular venue lead to better performance of the students, so the students may be encouraged to use the venue to practice laboratory work. At Australian University the students may be encouraged to attend laboratory sessions as the students who found attending laboratory sessions useful performed better. Interventions can be made to know why certain studens wouldn't like to take up a career in programming as the students who chose to have a career in programming performed better at Australian University.

For the other part of research, it was found that the use of Course management system as well as Social media, Facebook had positive impact on student learning. Thus, the students may be encouraged to get involved in the course/topic to ask questions as well as to clear their doubts with their classmates as well as Lecturer. It can help students progress during the week when they don't have opportunity to meet the Lecturer, Tutor or classmates to ask for help. By the use of this medium the students can obtain help irrespective of their location.

It can also help in student retention as the use of this medium indicated that it helps in student engagement which can lead to better retention in the topic if interventions are made ahead of time to determine the lack of engagement of students on these mediums.

The study can be implemented based on hypothesis outcomes to help students in learning programming. Positive factors can be chosen from both Universities and where applicable, they can be implemented into the course structure.

9.4 Summary

This chapter discussed the results achieved through statistical analysis. All the hypotheses were analysed and a summary of the hypotheses was presented. It also provided answers to the research questions. The next chapter presents a conclusion to the study.

CHAPTER 10 : CONCLUSIONS AND FUTURE WORK

This chapter provides the conclusion to the study, its limitations and outlines future work.

10.1 Conclusion

This research was significant as it identified new factors affecting learning of computer programming.

Out of 24 hypotheses tested at Australian University, Australia, 12 had statistical significance. Alongside this, out of 22 hypotheses for Indian University, India, 5 hypotheses had statistical significance. A total of 3 common hypotheses were statistically significant for both universities.

The results suggest that some factors may have a direct effect on learning programming, whereas other factors may have an indirect. It was also concluded that there are interrelationships emerging from the data across various factors investigated in this study. Surprisingly, some of these were common to both universities.

The results also suggest that the factors affecting learning programming are learning culture dependent. Some factors positively affected the results in Australia and a surprisingly different set of factors had a positive effect on results in India. Following this study, an improved, positive, research-driven pedagogy can now be designed to improve learning of programming for all students.

The peer group interaction and social integration study, involving the Course Management System (FLO) and social media (Facebook), suggests that students' voluntary participation in discussion forums should be monitored as an engagement indictor, so that tailored remedial action can be taken to prevent imminent failure, as the hypothesis that this was significant was confirmed. Data collected from the topic's discussion forum provided by the CMS (FLO) revealed positive correlations between students' levels of participation and their grades as: (a) inactive students are more likely to fail the topic, and (b) more active students are likely to get higher scores in the topic. We are conscious that our data is small and we need to collect more data to re-test, refine and confirm the current data.

The results are supported by the data collected from a Facebook Group, despite its small sample size. A preliminary survey reveals the reasons include a) fear of being judged, b) perceived limited benefits, and c) that the work involved creates an extra burden. However, higher engagement did correlate with higher outcomes. More systematic study is required to establish the reasons for rejection of this mode of learning in greater detail and then to harness the positive power of social media for teaching programming.

It is concluded that, compared with Facebook, FLO consistently proved to be the students' tool of preference.

10.2 Limitations

This study was a small-scale, preliminary study so up-scaling is necessary to validate the results further. Researcher involvement in this study was almost nil. To conduct the study in greater depth, the researcher would need to have greater, active involvement to encourage greater student participation. Scores were used as a key assessment structure, which may not be a true representation of the degree of learning achieved.

This study was conducted at two Universities in Australia and India. The study should now be rolled out to multiple Universities in Australia and India, with the results aggregated for each country separately and then cross-culturally.

10.3 Future work

This study needs to be conducted on a large scale to validate these results. Larger numbers of universities in both Australia and India should be evaluated, including a wider range of contexts, such as rural, regional and remote universities, rather than focusing on metropolitan centers. Such study can then be extended globally.

These results should then feed into a global study. Furthermore, other modes of data gathering, such as semi structured student interviews, should be added to understand participant reasoning of the events noted through surveys. The design of the current study only included quantitative data. In the future, qualitative data should also be gathered for analysis.

The parameters which showed effects in either of the two Universities, should be retained to see if they have similar or different effects in other universities if the sample size is increased.

The parameters from which nothing concrete could be concluded, could be retained to determine their effect on a larger sample population. All these issues could be investigated in future work.

The students were assessed for their performance based on the scores obtained in the course. Other ways of assessing students' performance could be used, which may give a better insight into student learning and understanding of the programming progress. One method may to ask students to write a piece of code with various concepts that have been taught during the semester, which may give a broader depiction of the students' understanding of the topic.

The number of participants from Indian University, India, was lower than that from Australian University, Australia. Since the researcher was coordinating the research with the help of the topic coordinator at Indian University, the students were asked to complete questionnaires online. It is likely that direct interaction with the researcher would increase the number of responses.

Additional assessment structure should be included in the next study, such as writing a piece of code as this would make more learning elements visible.

A number of interrelationships between factors were investigated in this study. Some additional factors should be investigated in the future to study their relationship with those factors already investigated here.

Since, the researcher was not directly involved with the students, this study could not be conducted on a large scale. Although the peer group interaction and social integration studies have generated some interesting results, to measure the effectiveness of the Course Management System (FLO) and Facebook group in learning programming, the study needs to be conducted with a larger sample size. Ways to get more students involved in the study need to be investigated. There was no lecturer involvement in the Facebook experiment. In future, lecturer involvement could be increased through interventions such as posting information related to the topic, or answering student questions. Problems could also taken into classroom discussions. Furthermore, the lecturer could post theory-based discussion questions. Some additional assignments or questions could be posted on FLO or Facebook for students who are high achievers and good performers may respond positively to the additional challenge of such complex programming tasks.

The lecturer could add lecture slides or lecture notes to FLO or Facebook group. Often, lecturers wish to discuss additional examples, especially for advanced programming but, due to a lack of time, these discussions have to be abandoned. In these ways, discussions could continue even after the formal classroom lectures have been completed.

As FLO proved to be the preferred tool for peer group interaction and social integration, it may serve as an additional resource in learning programming and an engagement indicator for the lecturer. Ways to involve more students on FLO should be explored, as the research suggests that students may be motivated to use FLO as a platform for asking programming related questions and clear their doubts, while solving programming assignments or doing laboratory work. Lecturers and tutors should be asked to provide answers to student questions, which may not be answered during the lectures due to a lack of time. The chat facility of FLO could also be used during lectures or to ask questions if they are not able to understand a concept or point. This may prevent them from distractions during the lecture, when disengaged.

The results of the FLO and Facebook users in the programming topic should be compared with their scores in other topics to analyse overall levels of achievement and compare learning outcomes. This study mainly focused on the use of FLO and Facebook and its impact on student performance in terms of scores. The impact of use of FLO and Facebook on student retention should also be studied in the future, as retention of students in programming topics is another challenge faced by lecturers worldwide as is shown in the literature review.

In terms of social media, only Facebook was explored in this study. New, diverse forms of social media should now also be explored

10.4 Self-reflection on the research journey

After completing this research, I am excited to share my learning from the study, the results achieved and consideration of ways to improve the research in the future.

The process of conducting this research was a personal as well as a professional challenge. There were times of despair and frustration as well as positive achievements and the happiness and satisfaction that come with such achievements. These achievements kept me motivated.

Since I started learning programming, I have understood that the way I was taught programming was as helpful as my practice habits. I had an innate interest in learning programming and would wake up in the middle of the night if I found a solution to one of the set problems. I worked as a software developer before shifting into academia. It was only when I started teaching programming that I realised that students find it hard to learn programming. I had not reviewed any of the research literature about pedagogy at that time but I started looking for reasons as to why some students find learning this topic so much harder than others. As a direct result of my own exploration of pedagogy, I decided to conduct this research.

Achievements of this research: After conducting this research, it was found that educational context was culturally determined. Certain factors affected Australian students' performance, but those factors did not necessarily affect Indian students' performance. This suggests that educational culture may have a significant effect on learning programming. The differences in results may be caused by different teaching methodologies used, assessment structure or examination structure. Further research should be conducted to find out the reason for difference in results.

Variation in expected results: It was expected that even if the learning cultures were different, the factors should have same effect on students' performance as programming is a technical topic and demographics or learning culture should not affect the learning. Moreover, the chosen factors were general factors based on Tinto's model of learning.

What I would do differently if I were to conduct this research again? After conducting this research, with hindsight, I would augment the method of collecting data with some qualitative data to have a better insight into the students’ reasons for their responses. I would also try to collect data through other modes, so that the data collected is even more reliable and detailed.

APPENDICES

University entry requirements Australia

Selection into university courses is based on both eligibility and rank. Eligibility allows a perspective candidate to be considered for selection; rank determines whether a perspective candidate is competitive enough to be selected. (July 2015)

Eligibility

To be eligible for selection into a university course/program a perspective candidate must:

qualify for the SACE/NTCET

obtain an Australian Tertiary Admission Rank (ATAR)

meet any prerequisite subject requirements for the course/program.

Competitiveness

A perspective candidate's competitiveness in relation to other applicants is based on your ATAR which is a rank given to students on a range from 0 to 99.95. The ATAR is calculated from your university aggregate.

To obtain a university aggregate and an ATAR one must:

Comply with the rules regarding precluded combinations

Comply with the rules regarding counting restrictions

Complete at least 90 credits of study in Tertiary Admissions Subjects (TAS) and Recognised Studies at Stage 2 in a maximum of three attempts which need not be in consecutive years

Of the 90 credits of study a minimum of 60 credits of study must be from 20 credit TAS*.

* Normally 10 credit subjects do not count towards this requirement but some 10 credit subjects in the same area, when studied in pairs, can substitute for a 20 credit subject. These are called valid pairs.

Calculating the university aggregate

The university aggregate is calculated from a perspective candidate's best scaled scores from three 20 credit TAS plus the best outcome from the flexible option, which is the best 30 credits of scaled scores or scaled score equivalents from:

The scaled score of a 20 credit TAS

Half the scaled score of one or more 20 credit TAS

The scaled score of one or more 10 credit TAS

Scaled score equivalents for Recognised Studies to the value of 10 or the maximum of 20 credits

Subject to precluded combination and counting restriction rules. Subjects with scaled scores of 0.0 can be used in the calculation of the university aggregate. The subjects used in the calculation can only come from a maximum of three attempts which need not be in consecutive years.

Converting the university aggregate to an ATAR

The university aggregate is converted to an ATAR. The ATAR is an indicator of how well a particular student has performed relative to other students. It is calculated as follows:

The group of students who may qualify for a university aggregate in 2015 is called the 2015 cohort.

For each university aggregate (in the range 0-90.0) obtained by the students in this cohort, the percentage of students who obtained that aggregate or better is calculated. This is known as calculating the percentile distribution.

Each university aggregate in the range 0-90.0 now has a corresponding percentile rank in the range 0-100. For example, if an aggregate of 78.0 or better out of 90.0 has been obtained by the top 10% of the cohort, the aggregate of 78.0 will correspond to a percentile rank of 90.0 (100 – 10).

To derive an ATAR from the university aggregate we need to look at where the students in the cohort sit compared to the entire population who are in the same age group.

The 2015 cohort may differ from that of other years in that it may represent a smaller or larger percentage of the population who are in the same age group.

The percentage from the given year is known as the participation rate. It is calculated using population statistics obtained from the Australian Bureau of Statistics and measuring these against the size of the cohort.

The percentile rank is adjusted to take account of the participation rate and where the student sits relative to the entire population, and the result is the ATAR. For example, if a student has an ATAR of 95.00 it indicates that they have achieved as well as, or better than, 95% of the population. This process ensures the ATAR is comparable from year to year.

When the calculations are completed, a student's relative position on the ATAR range is unchanged from the student's relative position on the university aggregate range.

It is important to remember that the ATAR is a rank, not a score, and that it cannot be calculated arithmetically from a university aggregate.

Reporting the university aggregate and ATAR

The university aggregate is reported to students on a score range of 0-90.0 with intervals of 0.1.

The ATAR is reported to students on a percentile scale, i.e. on a range 0-99.95 with intervals of 0.05.

The university aggregate and ATAR are reported only to students who qualify for the SACE or NTCET.

Prerequisites

Some university courses/programs require students to have studied one or more specific Stage 2 subjects to a minimum standard in order to be eligible for selection into the course/program. These subjects are known as prerequisites.

In order to fulfil a prerequisite subject requirement, you must obtain a minimum grade of C- or better. The grade is used (rather than the scaled score) because the course/program administrators are interested in how well you performed in the subject itself as measured against the learning requirements of the Subject Outline.

Since prerequisites are used to determine eligibility, not rank, they do not have to contribute to the university aggregate.

Assumed knowledge

Many university courses/programs recommend that commencing students have background knowledge in one or more specified Stage 1 or Stage 2 subjects or have an identified skill which will enhance the student's understanding of the course/program content. This is known as assumed knowledge.

Assumed knowledge is not compulsory and is not used in the selection process for entry to university courses/programs. Statements of assumed knowledge are intended purely to assist students in understanding course/program content and to allow them to make subject choices which may be of benefit to them in their future tertiary studies.

For admission to an engineering degree an ATAR ranging from 75 to 95 is required, depending upon the Engineering stream like Bachelor of Robotics Engineering requires an ATAR of 95 and Bachelor of Engineering Computer systems requires an ATAR of 75.(SATAC, 2017),(University, 2017b)

Eligibility Criteria for Admission to BTech/BE in India

The admission Criteria for Bachelor of Technology/ Bachelor of Engineering at Indian University can be viewed at (University, 2017d)

The candidate

(i) has passed 10+2 or equivalent examination with at least 60% scores (55% for SC/ST candidates) in aggregate of three subjects, namely, Physics, Mathematics and any one subject out of Chemistry, Biology, Biotechnology and Computer Science OR Minimum 60% (55% for SC/ST) scores in a Diploma recognized by AICTE or a state board of technical education of at least 3 year duration

(ii) Has appeared in JEE (Main)-2015 with at least 20% aggregate scores (15% for SC/ST candidates).

(iii) Possesses a good moral character.

(iv) Is a citizen of India.

(v) Is born on or after October 1, 1990 (5 years relaxation in age for SC/ST/PH

Candidates).

Page 1.

* 1. Student identification number

(You have been asked to give your student identification number so that we can correlate your marks/grades with the answers provided by you. It does not affect your exam marks/grades in any manner. The Professor/Lecturer who is teaching this topic to you will not get to know who has participated and who has not participated in the survey and thus your questionnaire answers will not have any implications on your marks or grades.)

* 2. Degree you are enrolled in, at Flinders University

Bachelor of Engineering(Computer Science)

Bachelor of Engineering(Information Technology)

Bachelor of Engineering(Robotics)

Bachelor of Engineering(Mechanical)

Bachelor of Engineering(Civil)

Bachelor of Engineering(Biomedical)

Bachelor of Creative Arts

Bachelor of Computer Science

Bachelor of Information Technology

Other

* 3. Gender

Male

Female

* 4. Did you study programming before attending this course at

	Yes	No
School (9th or 10th Grade)		
11th and 12th grade		
At home (out of interest) /self-study		
At University as part of a different degree		
At University as part of the same degree you are currently studying		

Page 2.

* 5. Have you studied any of the below mentioned languages?

	Yes	No
C		
C++		
Visual Basic		
Java		
PHP		
Python		
Basic		
COBOL		
VC++		
PASCAL		

* 6. Have you studied designing Flowcharts/Algorithms?

	Yes	No
Flowcharts		
Algorithms		

* 7. Why did you choose to study programming? (You can tick more than one box)

Not important

Somewhat important

Most important

Interested to know about programming

It is up-coming in the work market

High paying work in the industry

Mandatory in the degree

* 8. If you posted questions related to the topic/course/subject online on social networking or other websites to obtain help was the help useful or not?

	Not Helpful	Somewhat Helpful	Very helpful	Did not post
Google				
Twitter				
Facebook				
Email				

Page 3.

* 9. Please suggest the frequency if you study before going to the programming Lecture?

	Never	Sometimes	Often	Very Often	Always
Study lecture slides related to the current lecture available on FLO					
Study textbook slides related to the current lecture available on FLO					
Study lecture slides from the previous lecture available on FLO					
Study textbook slides from the previous lecture available on FLO					
Read paper based textbook					
Do online tutorials /read about the topic to be covered online before the lecture					

* 10. What do you study before going to the Laboratory?

	Yes	No
Study lecture slides related to the laboratory available on FLO		
Study textbook slides realted to laboratory available on FLO		
Read paper basedtextbook		
Do online tutorials /read about the topic to be covered online before the laboratory		
Read Previous laboratory work		

Practice previous laboratory work

Read new programs related to previous laboratory work

Practice new programs related to previous laboratory work

Read New similar programs related to the topic to be covered in the laboratory

Practice new similar programs related to the topic to be covered in the laboratory

Page 4.

* 11. Please tell us about your habits of revising the programming topic.

	Never	Sometimes	Often	Very Often	Always
During mid-semester break					
During mid-semester exams					
Both during mid-semester break and mid-semester exams					
Revised while the semester was in progress					

12. If you happen to revise the programming topic, please let us know if you revise the following?

	Never	Sometimes	Often	Very Often	Always
Theory from lecture slides available on FLO					
Textbook Slides available on FLO					

Laboratory Work

View lectures online

Revised on a website designed to revise the topic

Revised previous week's laboratory work

Revised New similar programs

Read previously done laboratory work

Redo previously done laboratory work

Read new similar programs

Redo new similar programs

Page 5.

* 13. From what sources do you try to seek help ?

	Never	Sometimes	Often	Very Often	Always
Consult Classmates					
Consult Senior Students who have already passed the topic					
Consult Lecturer					
Read Textbook					
Read Lecture					

Notes/Slides

Discuss the problem on discussion forums on FLO

Discuss the problem on Facebook /twitter

Discuss the problem on other socializing website

Opt for private tuition outside University

Attend help sessions at University

* 14. Was seeking help from these sources useful?

Not Useful   Useful Sometimes   Useful most of the time   Always Useful

Consult Classmates

Consult Senior Students who have already passed the topic

Consult Lecturer

Read Textbook

Read Lecture Notes/Slides

Discuss the problem on discussion forums on FLO

Discuss the problem on Facebook /twitter

Discuss the problem on other socializing website

Opt for private tuition outside University

Attend help sessions at University

Page 6.

* 15. Please write approximate number of programming lectures you attended

0%

Upto 20%

Upto 40%

Upto 60%

Upto 80%

100%

* 16. How often do you view the programming lectures online ?

	Never	Sometimes	Often	Very often	Always
All					
Important topics					
The ones you find difficult to understand					
The ones suggested by your classmates					
The ones suggested by your Lecturer/Professor (important topics					
If I need to understand a concept again					
If I need to take a note of some key points that I missed during the Lecture					

* 17. What do you do in the programming Lecture theatre? (You can tick more than one box)

	Never	Sometimes	Large part of Lecture	Whole Lecture
Listen to the Lectures				
Listen and make notes				
Annotate if you have printed notes				
Play games on mobile phone/laptop				
Look up for terms being discussed in the lecture				
Use social media to socialize				
Browse the internet in general				

* 18. You prefer to do the laboratory work .

	Yes	No
In the laboratory		
At home		
Library		

* 19. Do you find attending labs useful?

	Not at all	Slightly Useful	Useful	Very Useful	Extremely Useful
Usefulness of Attending Labs					

* 20. Please give reason for your answer..

Page 7.

* 21. Are you the first one from your family to attend University or you have other members in the family who attended University?

	Yes	No
First One		
Siblings		
Parents		

* 22. Is your home environment conducive to study?

Yes

No

* 23. Can you get programming related help at home from your parents or siblings?

Yes

No

*	24. Are your parents supportive of your educational goals?

Yes

No

*	25. Are you studying this topic for the first time?

First Time

Second Time

Other (please specify)

*	26. If given an option would you choose to study a topic/subject related to programming again?(i.e a topic other than what you have already studied)

Yes

No

*	27. Would you like to take a career or a job related to programming?

Yes

No

*	28. What are your educational goals?

BIBLIOGRAPHY

ABRAMSON, L. 2011. Can social networking keep students in school. *NPR: Morning edition*.

AGRESTI, A. 2002. Logit models for multinomial responses. *Categorical Data Analysis, Second Edition*, 267-313.

AL-RAHMI, W. M., ALIAS, N., OTHMAN, M. S., MARIN, V. I. & TUR, G. 2018. A model of factors affecting learning performance through the use of social media in Malaysian higher education. *Computers & Education,* 121, 59-72.

ARULAMPALAM, W., NAYLOR, R. & SMITH, J. 2004. Factors affecting the probability of first year medical student dropout in the UK: a logistic analysis for the intake cohorts of 1980–92. *Medical Education,* 38, 492-503.

BAIRD, L. L. 2000. College climate and the Tinto model. *Reworking the student departure puzzle*, 62-80.

BARKER, R. J. & UNGER, E. A predictor for success in an introductory programming class based upon abstract reasoning development. ACM SIGCSE Bulletin, 1983. ACM, 154-158.

BARTLETT, F. C. & BURT, C. 1933. Remembering: A study in experimental and social psychology. *British Journal of Educational Psychology,* 3, 187-192.

BEAN, J. P. 1980. Dropouts and turnover: The synthesis and test of a causal model of student attrition. *Research in higher education,* 12, 155-187.

BEAN, J. P. 1982. Student attrition, intentions, and confidence: Interaction effects in a path model. *Research in Higher Education,* 17, 291-320.

BENNEDSEN, J. & CASPERSEN, M. E. 2007. Failure rates in introductory programming. *ACM SIGCSE Bulletin,* 39, 32-36.

BENNEDSEN, J., CASPERSEN, M. E. & KÖLLING, M. 2008. *Reflections on the teaching of programming: Methods and implementations*, Springer.

BERGER, J. B. & BRAXTON, J. M. 1998. Revising Tinto's interactionalist theory of student departure through theory elaboration: Examining the role of organizational attributes in the persistence process. *Research in Higher education,* 39, 103-119.

BERGIN, S. & REILLY, R. Programming: factors that influence success. ACM SIGCSE Bulletin, 2005. ACM, 411-415.

BHARUCHA, J. 2018. Exploring education-related use of social media: business students perspectives in a changing India. *Education+ Training,* 60, 198-212.

BICEN, H. & CAVUS, N. 2011. Social network sites usage habits of undergraduate students: Case study of Facebook. *Procedia-Social and Behavioral Sciences,* 28, 943-947.

BIFFIGNANDI, S. & TONINELL, D. 2005. Sampling and inference in web surveys. Part I–Frames and Sampling. *WebSM,* 4, 1-17.

BIGGS, J. 1996. Western misconceptions of the Connician-heritage learning culture. *The Chinese Learners: Cultural, Psychological and Contextual Influences. Hong Kong, China: Comparative Education Research Centre and the Australian Council for Educational Research Ltd*.

BIGGS, J. B. 1987. *Student Approaches to Learning and Studying. Research Monograph*, ERIC.

BLATTNER, G. & LOMICKA, L. 2012. Facebook-ing and the social generation: A new era of language learning. *Alsic. Apprentissage des Langues et Systèmes d'Information et de Communication,* 15.
BORNAT, R. & DEHNADI, S. Mental models, consistency and programming aptitude. Proceedings of the tenth conference on Australasian computing education-Volume 78, 2008. Australian Computer Society, Inc., 53-61.
BOYD, D. M. & ELLISON, N. B. 2007. Social network sites: Definition, history, and scholarship. *Journal of Computer‐Mediated Communication,* 13**,** 210-230.
BOYLE, R., CARTER, J. & CLARK, M. 2002. What Makes Them Succeed? Entry, progression and graduation in Computer Science. *Journal of Further and Higher Education,* 26**,** 3-18.
BRUNSDEN, V., DAVIES, M., SHEVLIN, M. & BRACKEN, M. 2000. Why do HE students drop out? A test of Tinto's model. *Journal of further and Higher Education,* 24**,** 301-310.
BUGEJA, M. J. 2006. Facing the facebook. *The Chronicle of Higher Education,* 52**,** C1.
BUTLER, M. & MORGAN, M. 2007. Learning challenges faced by novice programming students studying high level and low feedback concepts. *Proceedings ascilite Singapore***,** 99-107.
BYRNE, P. & LYONS, G. The effect of student attributes on success in programming. ACM SIGCSE Bulletin, 2001. ACM, 49-52.
CABRERA, A. F., NORA, A. & CASTANEDA, M. B. 1993. College persistence: Structural equations modeling test of an integrated model of student retention. *The journal of higher education,* 64**,** 123-139.
CARLSON, N. 2010. At last-The full story of how Facebook was founded. Business Insider. Retrieved from https://en.wikipedia.org/wiki/Facebook#cite_note-8 17.09. 2015.
CARTER, L. 2006. Why students with an apparent aptitude for computer science don't choose to major in computer science. *ACM SIGCSE Bulletin,* 38**,** 27-31.
CASPERSEN, M. E. & KOLLING, M. 2009. STREAM: A first programming process. *ACM Transactions on Computing Education (TOCE),* 9**,** 4.
CHEN, J. & LIN, T.-F. 2008. Class attendance and exam performance: A randomized experiment. *The Journal of Economic Education,* 39**,** 213-227.
CHESTON, C. C., FLICKINGER, T. E. & CHISOLM, M. S. 2013. Social media use in medical education: a systematic review. *Academic Medicine,* 88**,** 893-901.
CLARKE, T. & CLARKE, E. 2009. Born digital? Pedagogy and computer-assisted learning. *Education+ Training,* 51**,** 395-407.
CLEAR, T., EDWARDS, J., LISTER, R., SIMON, B., THOMPSON, E. & WHALLEY, J. The teaching of novice computer programmers: bringing the scholarly-research approach to Australia. Proceedings of the tenth conference on Australasian computing education-Volume 78, 2008. Australian Computer Society, Inc., 63-68.
COLLINS, A., BROWN, J. S. & HOLUM, A. 1991. Cognitive apprenticeship: Making thinking visible. *American educator,* 15**,** 6-11.
CREDÉ, M., ROCH, S. G. & KIESZCZYNKA, U. M. 2010. Class attendance in college a meta-analytic review of the relationship of class attendance with grades and student characteristics. *Review of Educational Research,* 80**,** 272-295.
CUTTS, Q., FINCHER, S., HADEN, P., ROBINS, A., SUTTON, K., BAKER, B., BOX, I., DE RAADT, M., HAMER, J. & HAMILTON, M. The ability to articulate strategy as a

predictor of programming skill. Proceedings of the 8th Australasian Conference on Computing Education-Volume 52, 2006. Australian Computer Society, Inc., 181-188.
D'MONTE, L. 2009. *What is Twitter?* [Online]. http://www.socialmediaoracle.com/. Available: http://www.socialmediaoracle.com/twitter/what-is-twitter/ Retrieved June 2014, "Swine flu's tweet tweet causes online flutter. http://www.business-standard.com/india/news/swine-flu%5Cs-tweet-tweet-causes-online-flutter/356604/ [Accessed].
DE RAADT, M., HAMILTON, M., LISTER, R., TUTTY, J., BAKER, B., BOX, I., CUTTS, Q., FINCHER, S., HAMER, J. & HADEN, P. Approaches to learning in computer programming students and their effect on success. Proceedings of the 28th HERDSA Annual Conference: Higher Eduation in a Changing World (HERDSA 2005), 2005. Higher Education Research and Development Society of Australasia (HERDSA), 407-414.
DENNING, P. J. 2004. The field of programmers myth. *Communications of the ACM,* 47**,** 15-20.
DI VESTA, F. J. & GRAY, G. S. 1972. Listening and note taking. *Journal of educational psychology,* 63**,** 8.
DODERO, G. & DI CERBO, F. Extreme apprenticeship goes blended: An experience. Advanced Learning Technologies (ICALT), 2012 IEEE 12th International Conference on, 2012. IEEE, 324-326.
DUNN, O. J. 1964. Multiple comparisons using rank sums. *Technometrics,* 6**,** 241-252.
DURKHEIM, E. 1951. Suicide: A study in sociology (JA Spaulding & G. Simpson, trans.). *Glencoe, IL: Free Press.(Original work published 1897)*.
EBBINGHAUS, H. 1985. Remembering Ebbinghaus. *Contemporary Psychology,* 30**,** 519-523.
EBIZMBA. 2016. Top 15 Most Popular Social Networking Sites | April 2016. *The eBusiness Guide* [Online]. Available: http://www.ebizmba.com/articles/social-networking-websites.
ELDON, E. 2008. 2008 growth puts Facebook in better position to make money. *VentureBeat (San Francisco). http://venturebeat. com/2008/12/18/2008-growth-puts-facebook-in-better-position-to-make-money/(accessed on 29th Jan 2012)*.
ELKINS, S. A., BRAXTON, J. M. & JAMES, G. W. 2000. Tinto's separation stage and its influence on first-semester college student persistence. *Research in higher education,* 41**,** 251-268.
FACEBOOK. 2017. *Group Basics* [Online]. Available: https://www.facebook.com/help/162866443847527 [Accessed].
FINCHER, S., BAKER, B., BOX, I., CUTTS, Q., DE RAADT, M., HADEN, P., HAMER, J., HAMILTON, M., LISTER, R. & PETRE, M. 2005. Programmed to succeed?: A multi-national, multi-institutional study of introductory programming courses.
FINCHER, S., ROBINS, A., BAKER, B., BOX, I., CUTTS, Q., DE RAADT, M., HADEN, P., HAMER, J., HAMILTON, M. & LISTER, R. Predictors of success in a first programming course. Proceedings of the 8th Australasian Conference on Computing Education-Volume 52, 2006. Australian Computer Society, Inc., 189-196.
GEMMILL, E. L. & PETERSON, M. 2006. Technology use among college students: Implications for student affairs professionals. *NASPA journal,* 43**,** 280-300.

GETZLAF, S. B., SEDLACEK, G. M., KEARNEY, K. A. & BLACKWELL, J. M. 1984. Two types of voluntary undergraduate attrition: Application of Tinto's model. *Research in Higher Education,* 20**,** 257-268.

GHASEMI, A. & ZAHEDIASL, S. 2012. Normality tests for statistical analysis: a guide for non-statisticians. *International journal of endocrinology and metabolism,* 10**,** 486-489.

GODWIN-JONES, R. 2008. Emerging technologies: Web-writing 2.0: Enabling, documenting, and assessing writing online. *Language Learning & Technology,* 12**,** 7-13.

GOMES, A. J., SANTOS, A. N. & MENDES, A. J. A study on students' behaviours and attitudes towards learning to program. Proceedings of the 17th ACM annual conference on Innovation and technology in computer science education, 2012. ACM, 132-137.

GRABE, M. 2005. Voluntary use of online lecture notes: Correlates of note use and note use as an alternative to class attendance. *Computers & Education,* 44**,** 409-421.

GREENHOW, C., ROBELIA, B. & HUGHES, J. E. 2009. Learning, teaching, and scholarship in a digital age Web 2.0 and classroom research: What path should we take now? *Educational researcher,* 38**,** 246-259.

GROVER, A. & STEWART, D. W. 2010. Defining interactive social media in an educational context. *Cutting edge social media approaches to business education: Teaching with LinkedIN, Facebook, Twitter, Second Life, and Blogs***,** 7-38.

HAGAN, D. & MARKHAM, S. Does it help to have some programming experience before beginning a computing degree program? ACM SIGCSE Bulletin, 2000. ACM, 25-28.

HAGAN, D., SHEARD, J. & MACDONALD, I. Monitoring and evaluating a redesigned first year programming course. ACM SIGCSE Bulletin, 1997. ACM, 37-39.

HALPIN, R. L. 1990. An application of the Tinto model to the analysis of freshman persistence in a community college. *Community College Review,* 17**,** 22-32.

HAMILTON, J. & TEE, S. Extending the Biggs 3P teaching and learning model: a structural equation modeling approach. Proceedings of the Ninth International Conference on Electronic Business, 2009.

HAMILTON, J. & TEE, S. 2010. Teaching and learning: A SEM blended learning systems approach. *Higher Education Research and Development (HERD),* 29**,** 1-16.

HANCOCK, T. S. 1999. 360-degree feedback: key to translating air force core values into behavioral change. DTIC Document.

HARGITTAI, E. 2007. Whose space? Differences among users and non-users of social network sites. *Journal of Computer‐Mediated Communication,* 13**,** 276-297.

HOCKINGS, C., COOKE, S., YAMASHITA, H., MCGINTY, S. & BOWL, M. 2008. Switched off? A study of disengagement among computing students at two universities. *Research Papers in Education,* 23**,** 191-201.

HOLLANDER, M. & WOLFE, D. 1999. A.(1999), Nonparametric statistical methods. *J Wiley New York*.

HOLLANDER, M., WOLFE, D. A. & CHICKEN, E. 2013. *Nonparametric statistical methods*, John Wiley & Sons.

HOPKINS, J. 2006. *Surprise! There's a third YouTube co-founder* [Online]. USA Today. Available: http://usatoday30.usatoday.com/tech/news/2006-10-11-youtube-karim_x.htm Retrieved May, 2015 https://en.wikipedia.org/wiki/YouTube#cite_note-4

https://en.wikipedia.org/wiki/YouTube [Accessed].

HOPKINS, W. G. *Quantitative Research Design* [Online]. SPORTSCIENCE · sportsci.org. Available: http://www.sportsci.org/jour/0001/wghdesign.html [Accessed].
HUGGARD, M. 2004. Programming Trauma: Can it be avoided? Meriel Huggard, School of Computer Science & Statistics, Trinity College, Dublin 2, Ireland. meriel. huggard@ tcd. ie. *Education*.
HUNG, H.-T. & YUEN, S. C.-Y. 2010. Educational use of social networking technology in higher education. *Teaching in higher education,* 15**,** 703-714.
IBM 2015. IBM Corp. Released 2015. IBM SPSS Statistics for Windows, Version 23.0. Armonk, NY: IBM Corp.
JENKINS, T. On the difficulty of learning to program. Proceedings of the 3rd Annual Conference of the LTSN Centre for Information and Computer Sciences, 2002. 53-58.
JONES, S. & FOX, S. 2009. Generations online in 2009. Data memo. Washington, DC: Pew Internet and American Life Project. Retrieved March 7, 2010.
JUNCO, R., HEIBERGER, G. & LOKEN, E. 2011. The effect of Twitter on college student engagement and grades. *Journal of computer assisted learning,* 27**,** 119-132.
KEMBER, D., JAMIESON, Q. W., POMFRET, M. & WONG, E. T. 1995. Learning approaches, study time and academic performance. *Higher Education,* 29**,** 329-343.
KIETZMANN, J. H., HERMKENS, K., MCCARTHY, I. P. & SILVESTRE, B. S. 2011. Social media? Get serious! Understanding the functional building blocks of social media. *Business horizons,* 54**,** 241-251.
LAU, W. W. & YUEN, A. H. 2011. Modelling programming performance: Beyond the influence of learner characteristics. *Computers & Education,* 57**,** 1202-1213.
LENHART, A. & MADDEN, M. 2007. Social networking websites and teens.
LENHART, A., PURCELL, K., SMITH, A. & ZICKUHR, K. 2010. Social media and young adults. *Pew internet & American life project,* 3.
LEUTENEGGER, S. & EDGINGTON, J. 2007. A games first approach to teaching introductory programming. *ACM SIGCSE Bulletin,* 39**,** 115-118.
LICKLIDER, J. C. 1965. Man-computer partnership. *Ekistics***,** 165-169.
LIGHT, R. J. 1990. *The Report of Harvard Assessment Seminars: Explorations with Students and Faculty about Teaching, Learning, and Student Life: Seminar Lecture at NIU*, Harvard University Graduate School of Education.
LISTER, R. One small step toward a culture of peer review and multi-institutional sharing of educational resources: a multiple choice exam for first semester programming students. Proceedings of the 7th Australasian conference on Computing education-Volume 42, 2005. Australian Computer Society, Inc., 155-164.
LISTER, R., ADAMS, E. S., FITZGERALD, S., FONE, W., HAMER, J., LINDHOLM, M., MCCARTNEY, R., MOSTRÖM, J. E., SANDERS, K. & SEPPÄLÄ, O. A multi-national study of reading and tracing skills in novice programmers. ACM SIGCSE Bulletin, 2004. ACM, 119-150.
LISTER, R. & LEANEY, J. First year programming: let all the flowers bloom. Proceedings of the fifth Australasian conference on Computing education-Volume 20, 2003. Australian Computer Society, Inc., 221-230.
MA, L., FERGUSON, J., ROPER, M. & WOOD, M. Investigating the viability of mental models held by novice programmers. ACM SIGCSE Bulletin, 2007. ACM, 499-503.
MALONEY, E. 2007. What Web 2.0 can teach us about learning. *Chronicle of higher education,* 53**,** B26.

MARBURGER, D. R. 2006. Does mandatory attendance improve student performance? *The Journal of Economic Education,* 37**,** 148-155.
MASON, R. 2006. Learning technologies for adult continuing education. *Studies in Continuing Education,* 28**,** 121-133.
MASSINGHAM, P. & HERRINGTON, T. 2006. Does attendance matter? An examination of student attitudes, participation, performance and attendance.
MATNEY, M. & BORLAND, K. Facebook, blogs, tweets: How staff and units can use social networking to enhance student learning. Presentation at the annual meeting of the National Association for Student Personnel Administrators, Seattle, WA, 2009.
MAYER, R. E., DYCK , J. L. & VILBERG , W. 1986. Learning to program and learning to think: what's the connection? *Communications of the ACM,* 29**,** 605-610.
MCCARTHY, J. 2010. Blended learning environments: Using social networking sites to enhance the first year experience. *Australasian Journal of Educational Technology,* 26.
MCCARTNEY, R., ECKERDAL, A., MOSTROM, J. E., SANDERS, K. & ZANDER, C. Successful students' strategies for getting unstuck. ACM SIGCSE Bulletin, 2007. ACM, 156-160.
MCCRACKEN, M., ALMSTRUM, V., DIAZ, D., GUZDIAL, M., HAGAN, D., KOLIKANT, Y. B.-D., LAXER, C., THOMAS, L., UTTING, I. & WILUSZ, T. 2001. A multi-national, multi-institutional study of assessment of programming skills of first-year CS students. *ACM SIGCSE Bulletin,* 33**,** 125-180.
MCGETTRICK, A., BOYLE, R., IBBETT, R., LLOYD, J., LOVEGROVE, G. & MANDER, K. 2005. Grand Challenges in Computing: Education—A Summary. *The Computer Journal,* 48**,** 42-48.
MCGETTRICK, A. D. 2004. *Grand challenges in computing: Education*, British Computer Society.
MILANOVIC, R. 2015. *The World's 21 Most Important Social Media Sites and Apps in 2015* [Online]. http://www.socialmediatoday.com/. Available: http://www.socialmediatoday.com/social-networks/2015-04-13/worlds-21-most-important-social-media-sites-and-apps-2015 [Accessed].
MONTMARQUETTE, C., MAHSEREDJIAN, S. & HOULE, R. 2001. The determinants of university dropouts: a bivariate probability model with sample selection. *Economics of education review,* 20**,** 475-484.
MORAN, M., SEAMAN, J. & TINTI-KANE, H. 2011. Teaching, Learning, and Sharing: How Today's Higher Education Faculty Use Social Media. *Babson Survey Research Group*.
MORAVEC, M., WILLIAMS, A., AGUILAR-ROCA, N. & O'DOWD, D. K. 2010. Learn before lecture: a strategy that improves learning outcomes in a large introductory biology class. *CBE-Life Sciences Education,* 9**,** 473-481.
MOSTRÖM, J. E. 2011. A study of student problems in learning to program.
MURGUIA, E. 1991. Ethnicity and the Concept of Social Integration in Tinto's Model of Institutional Departure. *Journal of College Student Development,* 32**,** 433-39.
MURRAY, K. E. & WALLER, R. 2007. Social networking goes abroad. *International Educator,* 16**,** 56.
NEWELL, A. & SIMON, H. A. 1972. *Human problem solving*, Prentice-Hall Englewood Cliffs, NJ.

NORA, A., ATTINASI JR, L. & MATONAK, A. 1990. Testing qualitative indicators of precollege factors in Tinto's attrition model: A community college student population. *The Review of Higher Education,* 13**,** 337-355.
O'NEILL, N. 2010. The New Facebook Groups: All You Need To Know. Available: http://www.adweek.com/digital/new-facebook-groups/.
PASCARELLA, E. T. & CHAPMAN, D. W. 1983. A multiinstitutional, path analytic validation of Tinto's model of college withdrawal. *American educational research journal,* 20**,** 87-102.
PASCARELLA, E. T. & TERENZINI, P. T. 1980. Predicting freshman persistence and voluntary dropout decisions from a theoretical model. *The journal of higher education,* 51**,** 60-75.
PSYCHESTUDY. *Ebbinghaus Forgetting Curve* [Online]. Available: https://www.psychestudy.com/cognitive/memory/ebbinghaus-forgetting-curve [Accessed].
RICHARD, J., SINGER, J. D., WILLETT, J. B. & LIGHT, R. J. 2009. *By design: Planning research on higher education*, Harvard University Press.
ROBINS, A. 2010. Learning edge momentum: A new account of outcomes in CS1. *Computer Science Education,* 20**,** 37-71.
ROBINS, A., ROUNTREE, J. & ROUNTREE, N. 2003. Learning and teaching programming: A review and discussion. *Computer Science Education,* 13**,** 137-172.
RODDAN, M. 2002. The determinants of student failure and attrition in first year computing science. *Computing Science, Glasgow University, project Summer*.
RODRIGUEZ, J. E. 2011. Social media use in higher education: Key areas to consider for educators.
ROSS, C., ORR, E. S., SISIC, M., ARSENEAULT, J. M., SIMMERING, M. G. & ORR, R. R. 2009. Personality and motivations associated with Facebook use. *Computers in human behavior,* 25**,** 578-586.
SAGEPUB.COM. SAGE Publishing. Available: http://www.sagepub.com/upm-data/6418_Chapter_3A_Meyers_I_Proof_2 [Accessed].
SAGEPUB.COM. 2005. *Data Screening* [Online]. Available: https://www.sagepub.com/sites/default/files/upm-binaries/6418_Chapter_3A_Meyers_I_Proof_2.pdf [Accessed].
SATAC. 2017. *Undergraduate entry requirements* [Online]. SATAC. Available: http://www.satac.edu.au/undergraduate-entry-requirements [Accessed].
SELWYN, N. Web 2.0 applications as alternative environments for informal learning-a critical review. Paper for CERI-KERIS International Expert Meeting on ICT and Educational Performance, 2007. 16-17.
SEYAL, A. H. & MEY, Y. S. 2015. Evaluating Students' Perception of Teaching Learning Computer Programming: A Study in a Bruneian Technological University. *Journal of Education and Vocational Research,* 6**,** 25-33.
SHAPIRO, S. S. & WILK, M. B. 1965. An analysis of variance test for normality (complete samples). *Biometrika,* 52**,** 591-611.
SHARMA, R., SHEN, H. & GOODWIN, R. Voluntary participation in discussion forums as an engagement indicator: an empirical study of teaching first-year programming. Proceedings of the 28th Australian Conference on Computer-Human Interaction, 2016. ACM, 489-493.

SHARMA, S. 2016. Delhi girl tops CBSE Class XII exams, pass percentage in private schools dips. *The Indian Express*.
SHEARD, J., SIMON, S., HAMILTON, M. & LÖNNBERG, J. Analysis of research into the teaching and learning of programming. Proceedings of the fifth international workshop on Computing education research workshop, 2009. ACM, 93-104.
SHIH, R.-C. 2011. Can Web 2.0 technology assist college students in learning English writing? Integrating Facebook and peer assessment with blended learning. *Australasian Journal of Educational Technology,* 27.
SHIRISH, T. S. 2013. *Research Methodology in Education*, Lulu. com.
SIMON, S., FINCHER, S., ROBINS, A., BAKER, B., BOX, I., CUTTS, Q., DE RAADT, M., HADEN, P., HAMER, J. & HAMILTON, M. Predictors of success in a first programming course. Proceedings of the Eighth Australasian Computing Education Conference (ACE 2006), 2006. Australian Computing Society Inc., 189-197.
SMITH, R. A. & PETERSON, B. L. 2007. "psst… what do you think?" the relationship between advice prestige, type of advice, and academic performance. *Communication Education,* 56**,** 278-291.
SOLOMON, G. & SCHRUM, L. 2007. *Web 2.0: New tools, new schools*, ISTE (Interntl Soc Tech Educ.
SOROKINA, O. 2015. 8 Types of Social Media and How Each Can Benefit Your Business. Available from: https://blog.hootsuite.com/types-of-social-media/.
SPADY, W. G. 1970. Dropouts from higher education: An interdisciplinary review and synthesis. *Interchange,* 1**,** 64-85.
SPADY, W. G. 1971. Dropouts from higher education: Toward an empirical model. *Interchange,* 2**,** 38-62.
STANCA, L. 2006. The effects of attendance on academic performance: Panel data evidence for introductory microeconomics. *The Journal of Economic Education,* 37**,** 251-266.
STEWART, S. 2012. *Using Facebook for teaching, learning and assessment* [Online]. Available: http://sarah-stewart.blogspot.com.au/2012/02/using-facebook-for-teaching-learning.html [Accessed].
STUTZMAN, F. Our lives, our facebooks. 26th INSNA conference, in Vancouver, Canada, 2006.
SUSAN, A. 2010. Sociable Professors-First-of-its-Kind Survey Finds College Faculty Increasingly Using Social Media.
SWEET, R. 1986. Student dropout in distance education: An application of Tinto's model. *Distance education,* 7**,** 201-213.
TABACHNICK, B. G. & FIDELL, L. S. 2007. *Experimental designs using ANOVA*, Thomson/Brooks/Cole.
TEAGUE, D. & ROE, P. Collaborative learning: towards a solution for novice programmers. Proceedings of the tenth conference on Australasian computing education-Volume 78, 2008. Australian Computer Society, Inc., 147-153.
TECHOPEDIA. 2017. *Facebook Group* [Online]. techopedia. Available: https://www.techopedia.com/definition/4942/facebook-group [Accessed].
TINTO, V. 1975. Dropout from higher education: A theoretical synthesis of recent research. *Review of Educational Research,* 45**,** 89-125.
TOLHURST, D., BAKER, B., HAMER, J., BOX, I., LISTER, R., CUTTS, Q., PETRE, M., DE RAADT, M., ROBINS, A. & FINCHER, S. Do map drawing styles of novice

programmers predict success in programming?: a multi-national, multi-institutional study. Proceedings of the 8th Australasian Conference on Computing Education-Volume 52, 2006. Australian Computer Society, Inc., 213-222.
TOPPIN, I. N. 2011. Video lecture capture (VLC) system: A comparison of student versus faculty perceptions. *Education and Information technologies,* 16**,** 383-393.
TOPUNIVERSITIES.COM. 2017. *Top Universities in Australia 2018* [Online]. topuniversities.com: topuniversities.com. Available: https://www.topuniversities.com/university-rankings-articles/world-university-rankings/top-universities-australia-2018 [Accessed].
TRAPHAGAN, T., KUCSERA, J. V. & KISHI, K. 2010. Impact of class lecture webcasting on attendance and learning. *Educational technology research and development,* 58**,** 19-37.
UNIVERSITY, F. 2017a. *Computer/Engineering Programming Week 1* [Online]. Flinders Website. Available: file:///C:/Users/Tanis/AppData/Local/Packages/Microsoft.MicrosoftEdge_8wekyb3d8bbwe/TempState/Downloads/L01a_S2_2017_2up.pdf [Accessed].
UNIVERSITY, F. 2017b. *Required ATAR* [Online]. Flinders University. Available: https://www.flinders.edu.au/study/pathways/year-12-entry/required-atar [Accessed].
UNIVERSITY, T. 2017c. *Accreditations* [Online]. Thapar University. Available: http://www.thapar.edu/misces/pages/accreditations [Accessed].
UNIVERSITY, T. 2017d. *ADMISSION TO THE FIRST YEAR OF BE/BTECH PROGRAM* [Online]. Available: http://www.thapar.edu/images/admission17/ADMISSION%20TO%20FIRST%20YEAR%20UG%20PROGRAM.pdf [Accessed].
UNIVERSITY, T. 2018. *Disciplinary* [Online]. Thapar University. Available: http://www.thapar.edu/students/pages/disciplinary [Accessed].
UNSW SYDNEY, N. 2017. *Making lectures easier to follow* [Online]. UNSW Sydney, NS Website. [Accessed].
UZUNBOYLU, H., BICEN, H. & CAVUS, N. 2011. The efficient virtual learning environment: A case study of web 2.0 tools and Windows live spaces. *Computers & Education,* 56**,** 720-726.
WALSH, K. 2012. Can social media play a role in improving retention in higher education? research says it can. EmergingEdTech.
WANKEL, L. A. & BLESSINGER, P. 2013. Inventive approaches in higher education: An introduction to using multimedia technologies. *Increasing student engagement and retention using multimedia technologies: Video annotation, multimedia applications, videoconferencing and transmedia storytelling.* Emerald Group Publishing Limited.
WIKISPACES. 1996. *Biggs's work 1* [Online]. Available: https://inclusive-teaching.wikispaces.com/Biggs%27s+work+1 [Accessed].
WILK, M. B. & GNANADESIKAN, R. 1968. Probability plotting methods for the analysis for the analysis of data. *Biometrika,* 55**,** 1-17.
WILSON, B. C. 2002. A study of factors promoting success in computer science including gender differences. *Computer Science Education,* 12**,** 141-164.
WILSON, B. C. & SHROCK, S. Contributing to success in an introductory computer science course: a study of twelve factors. ACM SIGCSE Bulletin, 2001. ACM, 184-188.
WINKE, P. & GOERTLER, S. 2008. Did we forget someone? Students' computer access and literacy for CALL. *Calico Journal,* 25**,** 482-509.

WINSLOW, L. E. 1996. Programming pedagogy—a psychological overview. *ACM SIGCSE Bulletin,* 28**,** 17-22.
WOSZCZYNSKI, A., HADDAD, H. & ZGAMBO, A. 2005a. An IS student's worst nightmare: Programming courses.
WOSZCZYNSKI, A. B., HADDAD, H. M. & ZGAMBO, A. F. Towards a model of student success in programming courses. Proceedings of the 43rd annual Southeast regional conference-Volume 1, 2005b. ACM, 301-302.
WU, P. & HSU, L. EFL learning on social networking site?: An action research on Facebook. Teaching & Learning with Vision Conference. Queensland, Australia, 2011.
XIA, C., FIELDER, J. & SIRAGUSA, L. 2013. Achieving better peer interaction in online discussion forums: A reflective practitioner case study. *Issues in Educational Research,* 23**,** 97-113.
ZU, N., CHEN, Z., DUAN, Z., LI, L. & MENG, Z. 2017. A review of social media in education: effects and attitudes. *International Journal of Social Media and Interactive Learning Environments,* 5**,** 191-200.