

**Journal of Education & Social Sciences**

ISSN: 2410-5767 (Online)

ISSN: 2414-8091 (Print)

**A Study of Difficulties of Students in Learning Programming**

**Affiliation:**

Noman Islam

IQRA University, Karachi. E-mail: noman.islam@iuk.edu.pk

Ghazala Shafi Sheikh

IQRA University, Karachi. E-mail: ghazala.shafi@iuk.edu.pk

Ridah Fatima

IQRA University, Karachi. E-mail: ridah.fatima@iuk.edu.pk

Farrukh Alvi

Sir Syed University of Engineering and Technology, Karachi. E-mail: farukhalvi1988@gmail.com

**Manuscript Information**

**Submission Date:** July 27, 2019

**Acceptance Date:** November 01, 2019

**Citation in APA Style:**

Islam, N., & Sheikh, G. S., & Fatima, R., & Alvi, F. J. (2019). A Study of Difficulties of Students in Learning Programming, *Journal of Education & Social Sciences*, 7(2), 38-46.

**DOI:** <https://doi.org/10.20547/jess0721907203>





## A Study of Difficulties of Students in Learning Programming

Noman Islam \*      Ghazala Shafi Sheikh †      Ridah Fatima ‡      Farrukh Alvi §

**Abstract:** Programming has been a challenging subject for computer science students. A programmer must have strong analytical and reasoning skills to program effectively. In order to inculcate such skills, instructor must prepare their lectures accordingly and employ several novel techniques to develop analytical and problem solving skills into the students. Despite the strong need, there have not been any studies to identify the major challenges faced by programming students. Such a study can help instructor in development of appropriate materials and planning of lectures accordingly. To fulfill this gap, this work performs a survey of 57 students who just completed their first programming course. Students were asked to fill a questionnaire. The questionnaire was analyzed further to determine what are the most challenging concepts for students to learn and when do the students feel most comfortable learning. The correlations among students response were determined and k-mode clustering was also performed. Based on analysis, several recommendations were drawn. These recommendations can help instructor to develop their learning material. The paper concludes that flipped/inverted class room model is the most appropriate model for teaching programming course.

**Keywords:** Programming issues, flipped class room, teaching, algorithm, pointers, debugging.

## Introduction

The past few years have witnessed tremendous demands for computing expertise (Pirani & Hussain, 2019). These demands are primarily due to the advancements in computing technology and the tremendous impact of computing on the economy and GDP of the country. Despite these demands, the universities in computer science departments have seen decline in enrollments and high dropout rate in their programs (S. I. Malik & Coldwell-Neilson, 2017). These high dropout rates are as high as 30-40% and are associated with high failure rate in programming and apathy of students towards programming (Koulouri, Lauria, & Macredie, 2015). This is also reflected in the feedback obtained from industry professionals reflecting that most of the graduates show substandard performance in elementary programming tasks.

Programming is the core of computer science education and it develops the ability in the students to solve problems and polish their analytical capabilities. An introductory programming course lays the foundation for subsequent courses in the degree program. Realizing the challenges, most of the educational institutes are revising the contents of their introductory programming course (Koulouri et al., 2015). It is also essential to think

\* IQRA University, Karachi. E-mail: noman.islam@iuk.edu.pk

† IQRA University, Karachi. E-mail: ghazala.shafi@iuk.edu.pk

‡ IQRA University, Karachi. E-mail: ridah.fatima@iuk.edu.pk

§ Sir Syed University of Engineering and Technology, Karachi. E-mail: farukhalvi1988@gmail.com

about the first programming language that should be taught to students (Sheikh & Islam, 2016). Different choices such as C/ C++, C#, Java and Python are available.

Despite the challenging nature of programming course, there is lack of studies to analyze what are the actual challenges that students faces while studying programming course. To be precise, what concepts are the most difficult for students to grasp, and what are the strategies that help students in easily learn programming. To fulfill this gap, this paper performs a study of major programming issues. A survey was conducted in which participants were the students who just completed their first programming course. The objective of the study is to analyze what are the major issues faced by students. Based on this study, instructor can equip themselves, better plan their lectures and perform focused group studies to rectify most of the deficiencies of the students.

Rest of the sections of this paper are organized as follows. The next section presents literature review. This is followed by the details of proposed study. The results are presented afterwards and recommendations were drawn. The paper concludes with summary of the work presented and future work.

## Literature Review

There has been only a small number of works reported on specifically analyzing the programming issues faced by students studying their first programming course. Milne et al. (Milne & Rowe, 2002) conducted a survey of major concepts that are difficult for students to learn in the course of object oriented programming. They concluded that pointers and related concepts are very difficult to understand. Lahtinen et al. (Lahtinen, Ala-Mutka, & Järvinen, 2005) gathered opinion from students and tutors for the challenges that students face in learning programming. In (S. I. Malik & Coldwell-Neilson, 2017), a survey was conducted to explore the learning difficulties that students face while studying programming. The perception of students and teachers both were analyzed and an ADRI based approach is proposed to address those difficulties. A survey on misconceptions and students' difficulties in learning programming has been explored in (Qian & Lehman, 2017). The authors also attempt to identify the factors contributing to these difficulties.

A study of teaching low level programming to masters students for computer simulation has been presented in (Zakharova & Zakharov, 2018). According to the authors, teaching low level programming requires solid foundations of data flow management, synchronization, load balancing and fault tolerance. To inculcate the skills in students, special project tasks were developed and computational experiments were performed.

Teaching programming using games have been discussed in (Arawjo, Wang, Myers, Andersen, & Guimbretière, 2017). In some of the studies, flipped classroom settings have been explored for teaching programming (Çakıroğlu & Öztürk, 2017). Another study on identifying difficulty patterns in learning programming has been carried out in (Bosse & Gerosa, 2017). A quantitative evaluation of different approaches to teaching programming has been presented in (Koulouri et al., 2015). A triangulation study was performed in (Kalelioğlu, 2015) to analyze the gender difference in problem solving skills of students. N. Malik and Saad (2018) explored the social and emotional problems of undergraduate

students of Pakistan.

## Methodology

Figure 1 shows the proposed methodology. To analyze the major issues faced by programming students, a survey was conducted for the students who just completed their programming course. The total number of participants was 57. The objective of the survey was to analyze what concepts are difficult for students and when does the student feel most comfortable in learning? For conducting survey, survey monkey , Iqra university's virtual learning environment (VLE) and manual approach were used. Once the responses were gathered, they were analyzed further in Python. The descriptive statistics, association rules mining and k-mode clustering approaches were used.

**Figure 1**  
Proposed methodology

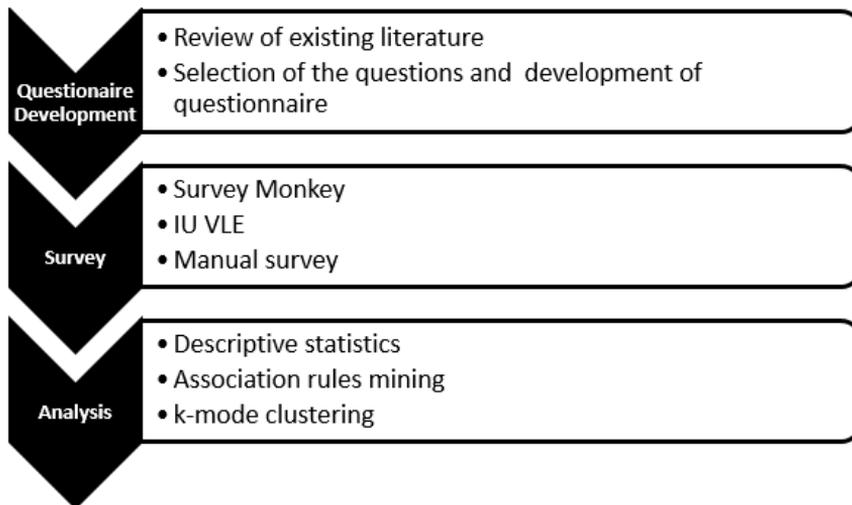


Figure 2 shows the questionnaire students were asked to fill. The responses of the students were analyzed further using various machine learning techniques. Machine learning is defined as explicitly learning a model from examples to predict outcome of unseen data (Lison, 2015). Machine learning can be divided into supervised and unsupervised learning. Supervised learning learns from labeled data where as unsupervised learning is learning from unlabelled data and is used for better understanding or analysis of data.

In this paper, first data analysis was performed using pandas library (McKinney, 2015). Then, associations among students were found out. For this purpose, Apriori algorithm was used (Yuan, 2017). Association rules mining attempts to discover hidden rules in data. If a certain number of transactions (students' response) show a pattern,

then association rules mining can be used to uncover such patterns. Finally, students' responses were clustered. K-mode clustering was applied (Goyal & Aggarwal, 2017) using the library "k-modes". Clustering is defined as the process of grouping similar data based on features (based on students' responses).

**Figure 2**

Questionnaires filled by students, analyzed further using various statistical and machine learning techniques

**Feedback about Programming Fundamentals Course**

1. Before taking this course, did you have any prior programming experience?  
a. Yes b. No

2. What type of challenges do you generally face while programming?  
a. Writing Algorithms b. Finding bugs c. Using IDE d. Understanding syntax e. Modularity/ writing functions  
If any other, specify: \_\_\_\_\_

3. Which programming concept do you feel is the most difficult to learn?  
a. Variables b. Expression c. Conditions d. Loops e. Arrays f. Pointers g. Functions h. Structures i. Filing j. Strings k. I/O  
If any other, specify: \_\_\_\_\_

4. Describe, when do you learn most about programming?  
a. In lectures b. During exercise session c. While studying alone  
If any other, specify: \_\_\_\_\_

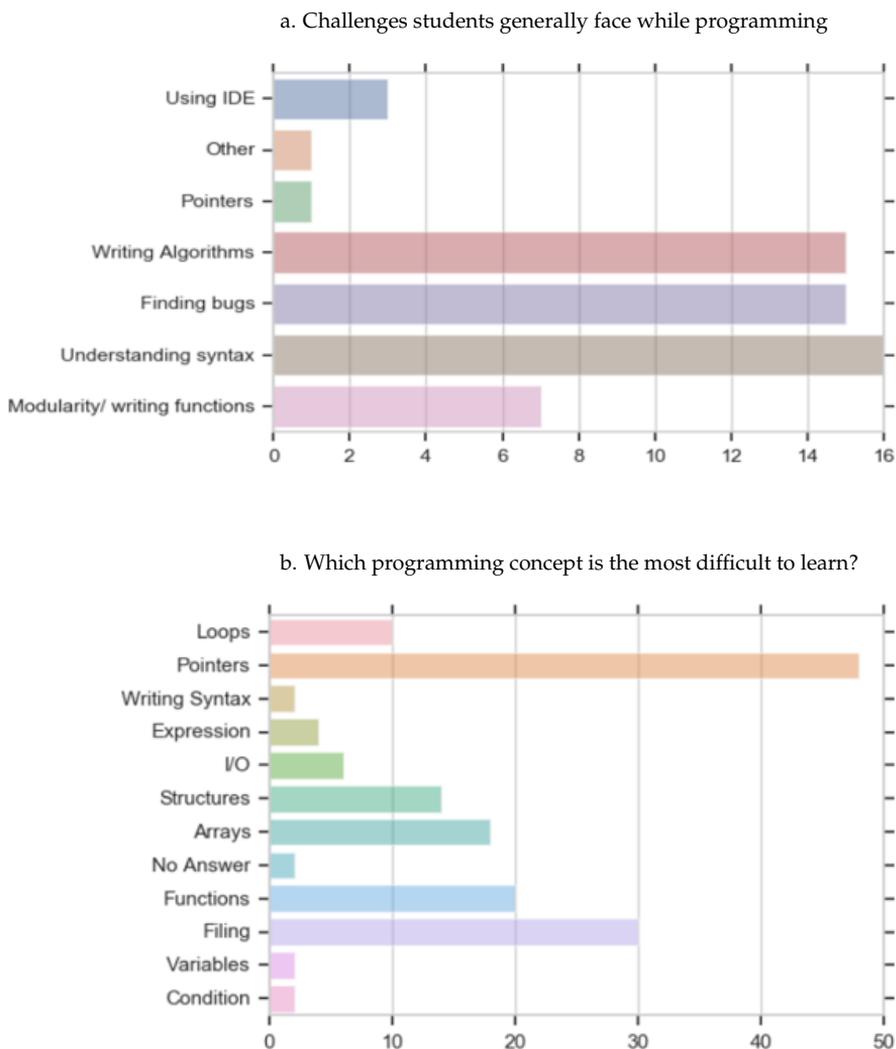
5. What kind of material helped you most in learning programming?  
a. Course book b. Lecture slides c. Programming exercise d. YouTube videos  
If any other, specify: \_\_\_\_\_

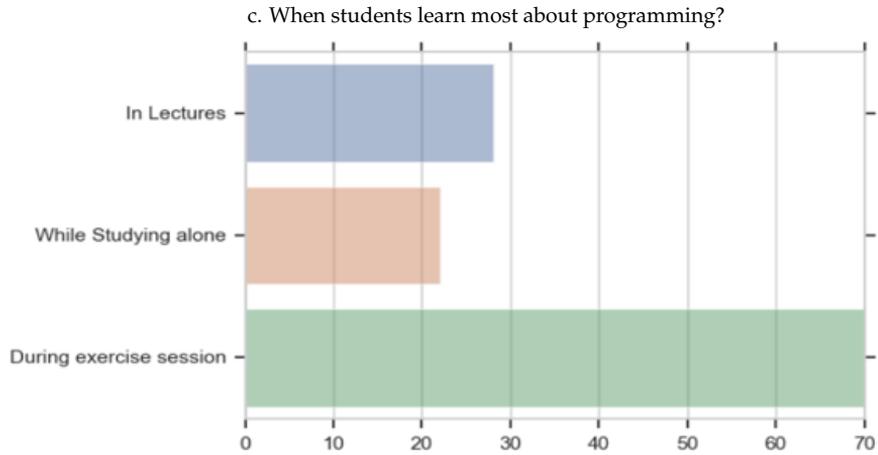
## Results

The responses gathered from students were analyzed further and results are shown in Figure 3. Figure 3a shows the challenges that students faced most while programming. Compared to the specific concepts of programming, it was found students feel that they had issues primarily in understanding syntax, debugging and writing algorithms.

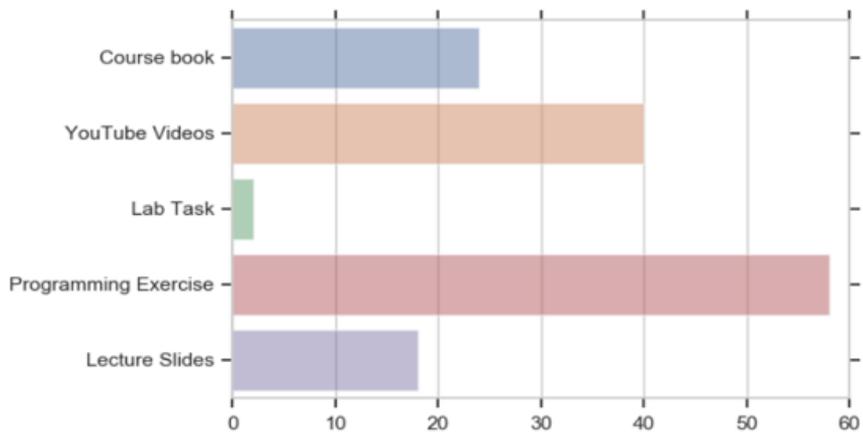
Figure 3b shows specifically which programming concept is the most difficult to learn for students. Pointers and filing are the concepts that were found to be most difficult concepts for students to learn. The questionnaire also asks when the students feel most comfortable learning the concepts. The students generally learn a lot while they are doing programming exercises. Finally, it was asked to the students what kind of materials helped them most in learning? The students response show that they learn when they are doing programming exercises or through YouTube videos.

**Figure 3**  
Analysis of students' responses to various questions of survey





d. What kind of material helped most in learning programming?



To analyze the students' response further, a correlation analysis was performed. In particular, association rules mining were applied to the responses. Figure 4 shows some of the rules identified using association rules mining.

It was found that students having issues in understanding filing, pointers, understanding syntax or debugging say that they learn a lot in exercise session or the exercises material helped them a lot in understanding the concepts. It was also found that students with problems in writing algorithms also have issues with filing, and these students learn a lot from YouTube videos.

**Figure 4**  
Results of applying association rules mining to students' response

<b>Filing</b>	→	<b>Exercise Sessions</b>
<b>Pointers</b>	→	<b>Programming Exercise</b>
<b>Understanding Syntax</b>	→	<b>Exercise Sessions</b>
<b>Finding Bugs</b>	→	<b>Programming Exercise</b>
<b>Writing Algorithms</b>	→	<b>Filing</b>
<b>Writing Algorithms</b>	→	<b>YouTube Videos</b>

To analyze further the responses, clustering was applied to students' response to see what are the major categories of respondents? Since the data is categorical, k-mode clustering was identified for clustering. Figure 5 shows the result of applying k-mode clustering. There were three types of students in the class, depending upon their previous experience of programming or the types of issues they face while programming. It was found that students with no programming experience had issues in writing algorithms and understanding syntax. Even students with some programming experience face difficulty in implementing the pointers. Irrespective of experience, all the students enjoy learning from exercises and YouTube videos.

**Figure 5**  
Results of applying k-mode clustering

<p><b><u>Group 1</u></b></p> <ul style="list-style-type: none"> <li>• No programming experience</li> <li>• Problem in understanding syntax</li> <li>• Problem in learning pointers</li> <li>• Learn during exercises</li> <li>• Learn from programming exercises</li> </ul>	<p><b><u>Group 2</u></b></p> <ul style="list-style-type: none"> <li>• Some programming experience</li> <li>• Difficulty in debugging</li> <li>• Problems with pointers</li> <li>• Learn during exercises</li> <li>• Learn from programming exercises</li> </ul>
	<p><b><u>Group 3</u></b></p> <ul style="list-style-type: none"> <li>• No programming experience</li> <li>• Difficulty in writing algorithms</li> <li>• Difficulty in filing concepts</li> <li>• Enjoy learning alone</li> <li>• Learn from YouTube videos</li> </ul>

## **Discussions and Recommendations**

It was identified writing algorithms is the most difficult challenge for students even more challenging than pointers. The other concepts that students face difficulty in learning are pointers, debugging and filing. Students find it convenient to study alone using YouTube videos. Most of the students learn during programming exercises. Also, people with even some programming experience face difficulty in pointers and debugging code.

This work is part of an ongoing study. It is concluded from this study that students prefer to watch videos and learn from programming exercises. Hence, flipped class room model is ideal for students to learn programming. As the next step, an experiment is being conducted with a batch of programming students with flipped class room model. In addition, a self access learning center (SALC) has been established, whose effectiveness can be analyzed further in future studies. One can also analyze the results with large sample of students to scrutinize the findings of this study. The analysis of the answer scripts of students' exam can also be performed to find correlation between their responses and the challenges they faced when attempting the real exam. In addition, other programming courses such as object oriented programming and data structures can be analyzed to identify the challenges that students face while learning these courses.

## **Conclusion**

In this work, a study of major programming issues (faced by students) was performed. The objective was to identify what major issues student face while programming and how can an instructor tailor their lectures or the approach towards teaching to benefit students. It was found out that flipped class room teaching is the ideal model for teaching programming courses. The paper can be extended further to analyze the effectiveness of flipped model for teaching programming course.

## References

- Arawjo, I., Wang, C.-Y., Myers, A. C., Andersen, E., & Guimbretière, F. (2017). Teaching programming with gamified semantics. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (pp. 4911–4923).
- Bosse, Y., & Gerosa, M. A. (2017). Why is programming so difficult to learn?: Patterns of difficulties related to programming learning mid-stage. *ACM SIGSOFT Software Engineering Notes*, 41(6), 1–6.
- Çakıroğlu, Ü., & Öztürk, M. (2017). Flipped classroom with problem based activities: Exploring self-regulated learning in a programming language course. *Journal of Educational Technology & Society*, 20(1), 337–349.
- Goyal, M., & Aggarwal, S. (2017). A Review on K-Mode Clustering Algorithm. *International Journal of Advanced Research in Computer Science*, 8(7), 725–729.
- Kalelioğlu, F. (2015). A new way of teaching programming skills to K-12 students: Code.org. *Computers in Human Behavior*, 52, 200–210.
- Koulouri, T., Lauria, S., & Macredie, R. D. (2015). Teaching introductory programming: A quantitative evaluation of different approaches. *ACM Transactions on Computing Education (TOCE)*, 14(4).
- Lahtinen, E., Ala-Mutka, K., & Järvinen, H.-M. (2005). A study of the difficulties of novice programmers. *Acm Sigcse Bulletin*, 37(3), 14–18.
- Lison, P. (2015). An introduction to machine learning. *Language Technology Group (LTG)*, 1, 35.
- Malik, N., & Saad, I. (2018). Exploring the social and emotional problems of undergraduates business students of pakistan: A qualitative case study. *Journal of Education and Social Sciences*, 6(2), 53–63.
- Malik, S. I., & Coldwell-Neilson, J. (2017). A model for teaching an introductory programming course using ADRI. *Education and Information Technologies*, 22(3), 1089–1120.
- McKinney, W. (2015). Pandas: A python data analysis library. Retrieved from [seehttp://pandas.pydata.org](http://pandas.pydata.org)
- Milne, I., & Rowe, G. (2002). Difficulties in learning and teaching programming-views of students and tutors. *Education and Information technologies*, 7(1), 55–66.
- Pirani, S., & Hussain, N. (2019). Technology is a tool for learning: Voices of teachers and parents of young children. *Journal of Education and Social Sciences*, 7(1), 55–66.
- Qian, Y., & Lehman, J. (2017). Students' misconceptions and other difficulties in introductory programming: A literature review. *ACM Transactions on Computing Education (TOCE)*, 18(1).
- Sheikh, G. S., & Islam, N. (2016). A qualitative study of major programming languages: Teaching programming languages to computer science students. *Journal of Information and Communication Technology*, 10(1).
- Yuan, X. (2017). An improved apriori algorithm for mining association rules. In *AIP conference proceedings* (Vol. 1820, p. 080005).
- Zakharova, I., & Zakharov, A. (2018). Key issues of low-level parallel programming in the individual projects for graduate students. *The International Journal of Engineering Education*, 34(4), 1250–1260.