A Phenomenological Study on the Feelings, Challenges and Difficulties Experienced by Information Technology Students in Learning Computer Programming

Cris Norman Olipas [1]

[1] *Nueva Ecija University of Science and Technology*
Sumacab este, Cabanatuan City, 3100, Philippines

DOI: 10.22178/pos.83-3

LCC Subject Category: : L7-991

Received 21.06.2022
Accepted 25.07.2022
Published online 31.07.2022

Corresponding Author:
olipas.cris@gmail.com

Abstract. This study aims to identify the experiences of information technology (IT) students in learning computer programming. Specifically, the feelings towards learning programming and the challenges and difficulties encountered by IT students were determined. The researcher used a phenomenological approach to determine the commonality of lived experiences among the students. The study was conducted during the second semester of Academic Year 2020–2021. The participants were first-year students enrolled in a computer programming I course. Open-ended questions were employed to gather responses. Since the data gathering happened during the coronavirus 2019 pandemic, the researcher used Google Form to collect responses. Participants were informed that the responses gathered were treated with the utmost confidentiality, security, and anonymity, following ethical research guidelines. Results revealed that participants have both positive and negative views about learning programming. As to the feelings towards learning computer programming, analysis shows that positive emotions include excitement and good happy feelings towards writing programs and finding them fun, challenging, and engaging. Negative feelings include being nervous, anxious, confused, and having difficulty learning how to write programs because it is hard. The challenges and problems encountered by IT students include difficulties relating to the program's syntax, errors and debugging processes; lesson-related; algorithm, analysis, resources, teaching, time-management, and personal factors. The researcher suggests recommendations that would improve the quality of teaching computer programming and offer new possible topics for future studies.

Keywords: challenges and difficulties; computer programming; lived experiences; phenomenological; qualitative.

INTRODUCTION

The field of information technology (IT) continues to grow, and the demand for IT professionals increases as humans realize the importance of IT in various fields and disciplines. Expertise in the information and communications technology field has become necessary to address the demands of different industries [1]. Information technology involves storing, retrieving, and sending information, which has become a vital component of everyday lives. IT is the "study of systematic approaches to select, develop, apply, integrate, and administer secure computing technologies to enable users to accomplish their personal, organizational, and societal goals" [2].

IT allows access to any information [3]. With the growing number of integrated systems and modern computing techniques and infrastructure, such as the cloud, access to information has become more accessible. IT also saves time in complex tasks and fosters innovations [3] in different sectors like education, business, healthcare, transportation, agriculture, and governance.

In recent years, IT has facilitated communication [3]. With improved and efficient communication tools through IT, opportunities and improvements have been achieved in different areas of society. Social media, for instance, has allowed people from other parts of the world to be easily connected. IT also evidently helps the education sector, especially during pandemics. With differ-

ent communities' much-needed strict restrictions to control the spread of Coronavirus 2019 (COVID-19), teaching and learning processes have still been conducted. The shift to online and flexible learning allowed students to continue learning in the comfort of their homes while ensuring safety and security.

The business sector has also benefited from IT. Operations and business processes have been improved using different computerized systems and tools. Opportunities have been created, and improvements have been achieved, which have benefited society. Marketing goods and services have dramatically improved with the use of different IT platforms. Thus, economic progress and a better quality of life for everyone are continuously achieved.

Writing computer programs is one of the essential skills for individuals wanting to pursue a career in information technology and related computing disciplines. In the Philippines, the Commission on Higher Education (CHED) laid guidelines for the higher learning institutions intending to offer or currently offering information technology and related computing programs through the CMO 25 series of 2015. This document sets the policies, standards, and guidelines for undergraduate programs in information technology, computer science, and information systems [4]. Programming is at the core of these computing programs. Thus, learning and developing this skill is a must for computing professionals.

Computer programming is writing instructions in a language machine understand to perform the desired process. The written machine instructions are called computer programs [5]. This set of codes, which the computer understands, performs functions and procedures to provide meaningful output relevant to end-users. A group of skills must be developed to write a computer program effectively. Skills in designing the algorithm, writing the program, understanding the syntax of the programming language, applying the correct logic, and the ability to use an integrated development environment (IDE) in constructing programs are necessary skills [6][7].

The mastery of a programming language is necessary for writing a program [8]. Today, various programming languages have been made available for professionals to develop computer programs effectively. Programming languages are classified into object-oriented programming languages, logic programming languages, procedural programming languages, functional programming languages, and scripting programming languages [9]. Another critical skill is debugging and problem-solving [8]. An individual needs to possess these skills because writing solutions to real-world problems is one of the ultimate goals of programming. Finding answers and translating them into codes is easier for an individual with good problem-solving skills. Logical and abstract thinking [8] is another essential programming skill. In writing programs, one must understand how to think logically to solve problems. Also, looking at issues and solutions in an abstract state helps create practical outputs. In addition, communication skills [8] are essential programming skills. Programmers must be able to effectively express their thoughts and ideas to others to effectively find solutions to existing problems and open opportunities for collaboration. Another skill one must develop is an eye for detail [8]. Writing programs require individuals to look into more information clearly to provide the expected output and solutions. Lastly, individuals must be good at memorizing [8]. Since many programming languages are available today, programmers must be very familiar with their language. While syntax may differ from one programming language to another, it is essential to deeply understand the similarities in structure, concepts, and use.

Learning programming languages and writing programs can be considered challenging tasks by students [10], especially for those with a weak logic and problem-solving foundation and those who lack materials to use, such as computers. Students must understand the problem, formulate the solution using standard problem-solving procedures and techniques, and write the solution using a programming language for computers to understand and execute to produce the expected output [11]. However, writing computer programs has been one of the challenges that students are experiencing, especially those with a weak foundation in logic, mathematics, critical thinking, and problem-solving. Many novice students have difficulty writing programs, which may lead to programming anxiety. In a study [12], researchers found that females tend to be more anxious about programming than males. However, in the same study, they also found out that the anxiety of females positively affects their academic performance. Thus, an increase in their

programming anxiety may also contribute to the improvement in their academic performance.

Previous studies have presented challenges and difficulties experienced by students [1][6][11], and the present study aims to contribute to the growing body of literature. The study focuses on the feelings, challenges and difficulties encountered in learning programming by first-year college students enrolled in the information technology undergraduate program. The study results may become a basis for crafting a plan of action to improve the quality of the teaching and learning process.

This study seeks to answer the following research questions:

1. How to describe the feelings toward learning programming IT students?

2. How do you explain IT students' challenges and difficulties writing computer programs?

The study focused on the feelings towards challenges and difficulties the selected first-year students encountered in learning programming from the course Computer Programming I during the first semester of Academic Year 2020–2021. Results were only drawn and analyzed from the responses of student-participants using open-ended questions.

METHODS

The study used a phenomenological type of qualitative research to identify the participants' lived experiences in learning programming. Phenomenological research seeks commonality within a particular group to arrive at a description of specific expertise or phenomenon [13]. The lived experiences of the IT students gathered using this approach provide an understanding of the feelings, challenges and difficulties encountered. This may result in developing a plan of action to mitigate the negative impact of such experiences in learning to program. Further, the results can also be a basis for improving instructional strategies in teaching programming.

The study was conducted during the second semester of Academic Year 2020–2021. Three hundred forty-eight (348) first-year students enrolled in computer programming I served as participants in this study. In the data gathering, the researcher employed a data gathering tool with open-ended questions about their feelings towards learning programming and the challenges and difficulties they encountered in writing programs. The conduct of data gathering happened at the time of the coronavirus 2019 pandemic. Thus, the researcher used Google Forms to collect online responses from the participants. The researcher ensured that the participants' answers to open-ended questions were treated with the utmost confidentiality.

RESULTS AND DISCUSSION

IT Students' Feelings Towards Programming. The results of the analysis on the most common feelings towards programming are shown in Figure 1.

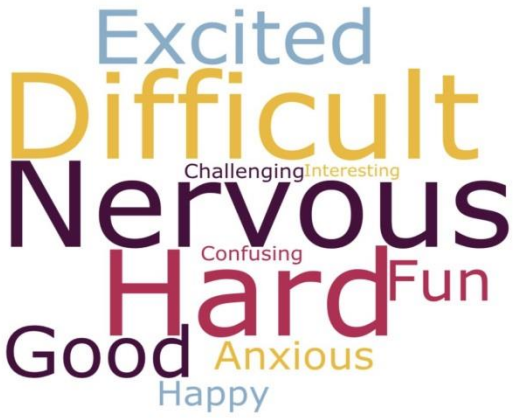

Figure 1 – Word cloud of the most common keywords mentioned about the feelings of IT students towards programming

Answers to open-ended questions revealed that participants had 55% positive and 45% negative feelings. The researcher extracted these positive and negative feelings expressed, and the results showed that in terms of negative emotions, students said that learning programming brought them feelings of being nervous (19.80%), anxious (4.60%), and confused (2.30%). In addition, participants expressed that learning programming is hard (17.5%) and difficult (14.66%). Described below are some of the responses of the student-participants:

"I felt programming is difficult and requires us to study hard".

"I feel nervous about writing programs; some parts are hard to understand".

"It is not easy for me. I feel so stressed sometimes when I don't get the right code running".

"Programming is not easy. Whenever our professor gives us an activity, I feel nervous because I am not good at programming".

"For me, at first, programming is not easy to understand. I feel uncomfortable and nervous at the same time".

"I felt uncomfortable writing programs because I think it is not easy to understand and I am not fully equipped with the things I needed to possess when programming".

"At first, I thought it was easy, but when problems were given to us, I feel like I cannot answer them".

"Palagi pong kinakabahan sa takot na magkamali" (Feeling nervous because of the fear of making mistakes).

On the other hand, student-participants positively expressed that learning programming brings them feelings of excitement (8.90%), fun (6.61%), and happiness (4.02%). Also, students felt good (8.33%) about learning how to write a program because they thought it was challenging (2.30%) and interesting (2.01%). The following statements are some of the responses of the student-participants to the open-ended questions.

"I feel excited to learn how to write programs even if it may be difficult".

"Programming boosts my confidence when I know that my program works. Though there is a feeling of nervousness, being proud is much greater".

"Learning how to write programming is a challenging and fun experience for me." "I feel happy, curious, and excited when we are talking about programming. It is fun when we are doing it".

"I felt inspired thinking about successful programmers and others who have created their programs".

"I feel good about programming. It is exciting to learn".

The expressed feelings about learning programming by the student-respondents are familiar to novice programmers. Authors [14] explain different emotions that novice programmers experience during their first computer programming learning session. Primary emotions include flow and engagement, confusion, frustration, and boredom. On the other hand, curiosity, happiness, anxiety, surprise, anger, disgust, fear, and sadness are rare feelings experienced. The perception of the student-participants that learning to program was difficult is supported by the studies of [15-17], expressing that programming in higher education requires problem-solving and higher-order thinking skills. Thus, students must develop these skills to excel in a computing discipline.

The feelings of anxiety, panic, and stress when unable to solve programming errors [18] have been reported as experienced by novice programmers. Negative emotions may lead to sorrow and despair [19], and giving up may lead to transferring to other degree programs when students cannot overcome them. These feelings toward learning programming should be addressed by appropriate intervention activities and programs that can lead to a more effective teaching-and-learning experience.

Challenges and Difficulties Encountered in Learning Programming. Figure 2 presents the most common challenges and difficulties in learning programming through a word cloud.

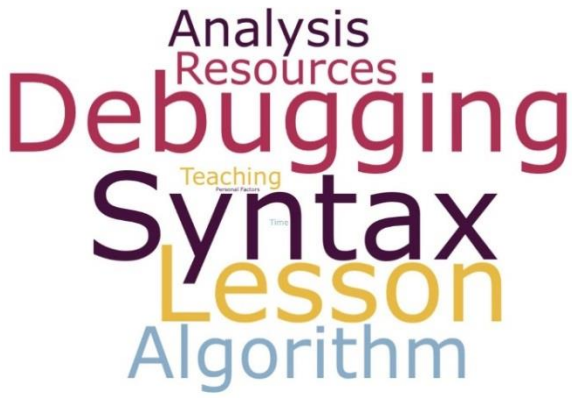

Figure 2 – Word cloud of the most common keywords mentioned about the challenges and difficulties encountered by IT students in learning programming

The top-most challenge and difficulty experienced by student participants involved writing the syntax (28.16%). They expressed that writing computer programs alone is already challenging because they need to be familiar with the programming language, its syntax, operators, and ways to write them. This is followed by debugging (11.78%). Student-participants assert that

finding errors and solving them is the next challenging task when learning to program. Meanwhile, lesson-related difficulties got 11.49% of the responses of the participants. It is found that understanding the topics such as decision statements, iteration statements, and other essential information needed in programming is challenging.

On the other hand, writing algorithms (7.76%) was also expressed as a challenging activity for students. Applying appropriate algorithms to programs requires in-depth analysis. Thus, analyzing (5.46%) followed after this. For some, their resources are a significant concern. Students have difficulty learning programming because of the lack of resources such as computers (4.89%). Others find it difficult to understand how instructors discuss (2.58%), manage their time (.86%), and other personal factors (.57%). The student-respondents expressed the following statements about the challenges and difficulties experienced in learning programming.

"I have no computer to use to practice writing programs".

"It is difficult for me to write the syntax. Most of the time, I forget to use symbols like semicolons properly, which results in errors in my program".

"Some lessons are hard to understand, like looping".

"Applying the correct logic for the program or proper algorithm is difficult".

"Finding errors in my program is challenging. Sometimes I don't know why my program doesn't run even if the code is correct".

The challenges and difficulties identified and analyzed from the responses of the student-participants provide a clearer view of how to improve the teaching and learning process to have a more effective and quality IT education.

CONCLUSIONS

Based on the findings of this study, the following conclusions were drawn:

1. Feelings toward learning to program by IT students are classified into positive and negative. Positive emotions include excitement and good happy feelings towards writing programs and finding them fun, challenging, and engaging. Negative feelings include being nervous, anxious, confused, and having difficulty learning how to write programs because it is hard.

2. The challenges and difficulties IT students encounter include problems relating to the program's syntax, errors and debugging processes; lesson-related; algorithm, analysis, resources, teaching, time-management, and personal factors.

3. Identified feelings towards programming and the challenges and difficulties experienced by IT students can be a basis for a plan of action in the college.

The following recommendations are presented based on the conclusions made:

1. Instructors teaching programming courses may consider employing classroom activities to boost the students' motivation and interest to lessen the discipline's negative perception. This will also lead to the effective development of programming skills.

2. Innovation in teaching strategies and materials may be considered to lessen the challenges and difficulties experienced by students.

3. An intervention plan of action may be crafted to improve the teaching of programming courses.

REFERENCES

1. Rosminah, M. D., & Zamzuri, M. A. (2012). *Difficulties in Learning Programming: Views of Students*. Retrireved from https://www.researchgate.net/publication/267338258_Difficulties_in_learning_programming_Views_of_students
2. Task Group on Information Technology Curricula. (2017). *Information Technology Curricula 2017*. Retrieved from https://www.acm.org/binaries/content/assets/education/curricula-recommendations/it2017.pdf
3. Basit, N. (2021, December 23). *15 Reasons Why Information Technology is Important*. Retrieved from https://curiousdesire.com/reasons-why-information-technology-is-important

4. Policies, standards, and guidelines for Bachelor of Science in Computer Science (BSCS), Bachelor of Science in Information Systems (BSIS), and Bachelor of Science in Information Technology (BSIT) programs (Philippines), No 25, S 2015. Retrieved June 1, 2022, from https://ched.gov.ph/wp-content/uploads/2017/10/CMO-no.-25-s.-2015.pdf

5. Wilkins, J. (2021, October 21). *What is Computer Programming?* Retrieved from https://freecodecamp.org/news/what-is-programming

6. Baist, A., & Pamungkas, A. S. (2017). Analysis of Student Difficulties in Computer Programming. *VOLT : Jurnal Ilmiah Pendidikan Teknik Elektro, 2*(2), 81. doi: 10.30870/volt.v2i2.2211

7. Rahman, T., Anuar, N., & Said, M. (2018, January 6). How the nature of programming and learning materials affects novice learner's motivation and programming ability. *Proceedings of the 6th International Conference on Information and Education Technology*. doi: 10.1145/3178158.3178184

8. Designveloper (2022, January 13). 6 Best Computer Programmer Skills That You Should Focus On. Retrieved from https://www.designveloper.com/blog/computer-programmer-skills

9. Kumar, S. (2021). 5 Different Types of Programming Languages. The Crazy Programmer. Retrieved from June 10, 2022, from https://www.thecrazyprogrammer.com/2021/09/types-of-programming-languages.html?twitter=@aneeshnair&@aarushinair_

10. Kadar, R., Wahab, N. A., Othman, J., Shamsuddin, M., & Mahlan, S. B. (2021). A Study of Difficulties in Teaching and Learning Programming: A Systematic Literature Review. *International Journal of Academic Research in Progressive Education and Development, 10*(3), 591–605

11. Moström, J. E. (2011). *A Study of Student Problems in Learning to Program*. Retrieved from http://umu.diva-portal.org/smash/get/diva2:447104/FULLTEXT02

12. Olipas, C., & Luciano, R. (2020). Understanding The Impact Of Using Countdown Timer On The Academic Motivation And Computer Programming Anxiety Of IT Students: The Case Of A State University In The Philippines. *International Journal of Scientific and Technology Research, 9*(3), 235–240.

13. Creswell, J. W. (2013). *Qualitative inquiry and research design: Choosing among five approaches*. London: Sage Publications.

14. Bosch, N., D'Mello, S., & Mills, C. (2013). What Emotions Do Novices Experience during Their First Computer Programming Learning Session? *Artificial Intelligence in Education*, 11–20. doi: 10.1007/978-3-642-39112-5_2

15. Amnouychokanant, V., Boonlue, S., Chuathong, S., & Thamwipat, K. (2021). A Study of First-Year Students' Attitudes toward Programming in the Innovation in Educational Technology Course. *Education Research International, 2021*, 1–10. doi: 10.1155/2021/9105342

16. Mladenović, M., Rosić, M., & Mladenović, S. (2016). Comparing elementary students' programming success based on programming environment. *International Journal of Modern Education and Computer Science, 8(8)*, 1–10. doi: 10.5815/ijmecs.2016.08.01

17. Robins, A., Rountree, J., & Rountree, N. (2003). Learning and Teaching Programming: A Review and Discussion. *Computer Science Education, 13*(2), 137–172. doi: 10.1076/csed.13.2.137.14200

18. Rogerson, C., & Scott, E. (2010). The Fear Factor: How It Affects Students Learning to Program in a Tertiary Environment. *Journal of Information Technology Education: Research, 9*, 147–171. doi: 10.28945/1183

19. Shuhidan, S., Hamilton, M., & D'Souza, D. (2009). *A Taxonomic Study of Novice Programming Summative Assessment*. Retriebved from https://www.researchgate.net/publication/228984735_A_taxonomic_study_of_novice_programming_summative_assessment