

Evaluating Student Proficiency in Programming Courses: A Case Study of the Computer and Information Technology Faculty

Mohammed Fadhl Abdullah ^(1,*)

Mona Awadh Joher ⁽²⁾

Nusiba Ali Mohammed ⁽³⁾

Hussein Muhammed Almoflhy ⁽⁴⁾

Received: 08/05/2024

Revised: 07/06/2024

Accepted: 22/06/2024

© 2025 University of Science and Technology, Aden, Yemen. This article can be distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

© 2025 جامعة العلوم والتكنولوجيا، المركز الرئيس عدن، اليمن. يمكن إعادة استخدام المادة المنشورة حسب رخصة مؤسسة المشاع الإبداعي شريطة الاستشهاد بالمؤلف والمجلة.

¹ Faculty of Engineering and Computing, University of Science & Technology, Aden, Yemen.

² Faculty of Computing and Information Technology, Aden University, Aden, Yemen. moawgo90@gmail.com

³ Faculty of Computing and Information Technology, Aden University, Aden, Yemen. nusibahsirduch@gmail.com

⁴ Faculty of Computing and Information Technology, Aden University, Aden, Yemen. Alyafayay90200@gmail.com

* Corresponding Author Designation, Email: m.albadwi@ust.edu

Evaluating Student Proficiency in Programming Courses: A Case Study of the Computer and Information Technology Faculty

Mohammed Fadhl Abdullah
Faculty of Engineering & Computing
University of Science & Technology
Aden, Yemen.
m.albadwi@ust.edu

Mona Awadh Joher
Faculty of Computing
& Information Technology
Aden University
Aden, Yemen.
moawgo90@gmail.com

Nusiba Ali Mohammed
Faculty of Computing &
Information Technology
Aden University
Aden, Yemen
nusibahsirduch@gmail.com

Hussein Muhammed Almofly
Faculty of Computing
& Information Technology
Aden University
Aden, Yemen.
Alyafayay90200@gmail.com

Abstract—The process of developing computer programs using one or more programming languages is known as computer programming. These programs are collections of instructions that computers use to complete specific tasks. This study examines the hurdles and difficulties that students at Aden University's Faculty of Computer and Information Technology face when learning to program. The study involved 45 students from the College of Computer Science and Information Technology, and a questionnaire was created specifically for this investigation to ensure content validity and reliability. The study found that students experienced a variety of issues, including a lack of time for practical exercises, language hurdles hurting their skills owing to insufficient English proficiency, a lack of prior programming knowledge, and challenges connected with traditional teaching techniques. The findings highlight the need of making recommendations and offering assistance to help students improve their programming abilities, reduce dropout rates, and improve their academic achievement. This study intends to provide helpful insights to help students overcome these problems and achieve in programming courses.

Keywords — Programming skills, teaching strategies, learning difficulties.

I. INTRODUCTION

Nowadays, with the rapid development in all computer science's fields such as machine learning, artificial intelligence, deep learning, data analytic and neural network, it is important, if you want to pursue in one of these disciplines, to understand and learn computer programming. Computer programming is a process for telling a computer which tasks to proceed in order to solve problems [14]. Computer programming is an essential undergraduate course for Computer Science, Information Technology, and Software Engineering departments [8]. Learning programming is

difficult and presents major challenge for learners. It is a complex intellectual activity which is learned by a lot of practice and requires deep and accurate understanding of all of the concepts and rules involved [9].

Most of the students who registered for programming courses are considered as beginners with no knowledge, so the rate of failure becoming higher. The objective of teaching computer programming is to develop in students the competencies required of a professional software developer. Current computer programmers suffer from a wide range of difficulties. Research works conducted about failures in programming courses have most often focused on the inability or the weakness of the student to understand the concepts. Many of these students find programming to be difficult and disheartening especially when they are beginners and even when they take advanced programming courses [12]. This weakness leads to relatively high failure rates and drop-out which in years have created negative opinion and fear about taking programming courses.

The aim of this work is to study the reasons of the student weaknesses or their failure in programing courses. The research is conducted within the Faculty of Computer and Information Technology at the university of Aden and understand the real difficulties and the activities that most influence the whole teaching and learning process of the programming. In this work we investigate the factors that lead to perform poorly in programming course, we also try to identify the effective teaching strategies.

The paper is organized as follows: Section 2 offers a comprehensive review of related work. Section 3 explains the methodology used. Section 4 provides a discussion of the findings and research results. Section 5 summarizes the paper, its contributions, and outlines future work for further improvement.

II. LITERATURE REVIEW

A. Basic Programming Skills

To become a professional programmer, a student must acquire series of abilities [4]. Some of these abilities include understanding problems, making plans, and implementing them in computer programs. This involves the ability to design algorithms, write programs, understand syntax, as well as program logic [10].

According to Harimurti, Ekohariadi, Munoto, Asto, & Winanti, [5] programming involves computational thinking, which is a problem-solving approach that involves breaking down complex problems into smaller, manageable parts. This helps programmers develop a systematic and analytical mindset, enabling them to approach problems in a structured and organized manner.

Furthermore, programming requires attention to detail and precision. Programmers need to write code that is accurate and error-free, which enhances their attention to detail and ability to identify and fix mistakes.

B. Difficulties and Challenges

Some specific challenges that students encounter when trying to learn new programming languages include lack of motivation, complex concepts, keeping up with latest developments, limited resources, lack of visualizations, lack of simplicity and power [8].

Many universities undergraduate students have problems with the selection of an appropriate learning environment or programming language. In addition, there are 10 causal attributions that were either cultural or specific to computer programming. The top 10 causal attributions were 'learning strategy', 'lack of study', 'lack of practice', 'subject difficulty', 'lack of effort', 'appropriate teaching method', 'exam anxiety', 'cheating', 'lack of time', and 'unfair treatment'. [6].

According to AbdelRahman, AL-Syabi, Al Sharji, & Al Kaabi [1], Language barriers where some students struggled with programming courses due to difficulties with the English language, and lack of attention to reference materials that means some students relied solely on lecture content without referencing other sources, such as scientific books and educational programming CDs, which may have hindered their understanding and knowledge acquisition are another reason of failure.

Weaker students, lack of problem-solving skills, analytical thinking, logical reasoning, programming planning, conceptual understanding, algorithmic skills, conceptual difficulty of curriculum elements, feedback level, and study patterns identified as causes of failure in programming cited by Sarpong, Arthur, & Owusu, [12].

Inadequate time allocated for laboratories and tutorial sessions, inadequate quantity and quality of homework assignments, lack of assistance by instructors in solving homework with students, questionable competency of

instructors' abilities to motivate students for learning, and the negative impact of the absence of quality assurance and monitoring methods on the teaching process may contribute to difficulties in acquiring programming skills and may affect the performance of students in computer programming courses [2].

C. Suggestions and Recommendations

Suggested the development of visualization tools as learning aids to help students understand the implementation of program code and changes that occur in the computer memory [15]. In a paper by Othman & Abdul Wahab [11] he mentioned that encouraging students to practice answering exam questions and discussing answers with their lecturers and using systematically designed direct instruction activities with feedback and practice opportunities, exposing students to problem-solving strategies like planning, simplification, and modeling can have a high impact on students' knowledge and ability to apply it in the industry.

According to Sentance & Csizmadia, [13] provide some examples which have errors to be fixed or examples that need re-writing more efficient perhaps and get students to explain their decisions) and get them working away from the computer at times to ensure they consider the steps of the program they are undertaking rather than just hacking away on the computer. At results of Holvikivi [7] research, Educators agree that students need to learn a programming way of thinking, which is more than just knowing the syntax and semantics of a language. Innovative approaches have been developed to aid beginner programmers, such as Stein's "computing as interaction" and Radenski's "Python first, Java second" program.

III. RESEARCH METHODOLOGY

This paper explores the challenges and difficulties encountered by students at the Faculty of Computer and Information Technology, Aden University, in grasping programming courses. The study included 45 students from the College of Computer Science and Information Technology, and a questionnaire was specifically developed for this investigation, ensuring content validity and reliability.

Questionnaires were distributed to second year students of computer and information technology college. We can find three majors within the college (computer science, information systems and cyber security and networks). All the study plans of these majors have computer programming courses which are prerequisite and depend on one other.

The samples are selected as following; (i) Students are chosen randomly and their identity stays anonymous, (ii) Teachers are not involved in this survey, and (iii) For data validation purposes, all uncompleted questionnaires and those with multiple answers were excluded. The survey was divided into

four classes of questions. Three questions related to student, three to materials, two associate with teacher and two about environment (Table 1).

Table 1: Survey Main classes

No.	Class	Questions
1	Student	1-3
2	Material	4-6
3	Teacher	7, 8
4	Learning Environment	9, 10

For our analysis, we have used the schemas of Excel program, and employed descriptive analyses tools to obtain some statistical data such as Frequencies and Percentages.

IV. RESEARCH RESULTS

The findings of the student’s Questionnaire:

In Yemen, the English course starts from the seventh class of primary school where the student's ability to learn become less than when he was child, so it is normally that students face difficulties in learning English language and other courses related with. (see figure 1) .

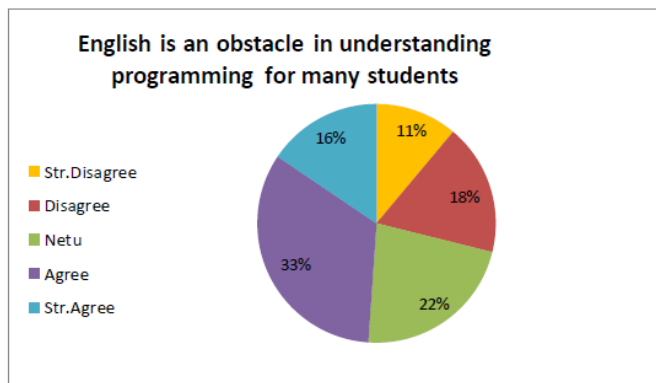


Figure 1: English language Factor.

Most programming courses in different colleges have four contact hours (two hours for theory, two hours for practical). We check the effect of this factor and we find that most of students think that two hours for practical are not enough. (see figure 2).

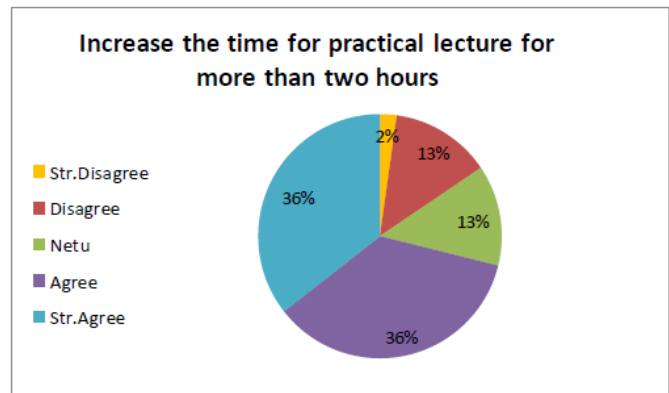


Figure 2; Practical Time Factor.

The students are divided into groups in the practical section. The number of students in each group must not exceed 20 student. The big numbers of students in the practical groups prevent the teacher to pass to every student and see what he did ,so it effects on the students' understanding of programming courses. (see figure 3).

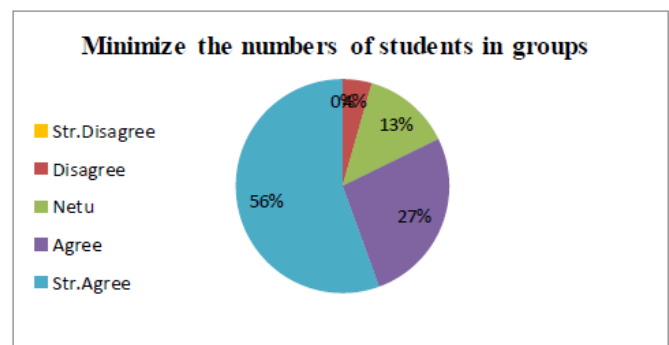


Figure 3: students’ number in groups Factor.

Sometimes, the teacher does not complete the study plan of the course due to many reasons such as a spread of epidemic, the strikes, wars or revolution ,all this causes in reduce the timetable of study. (see figure 4)

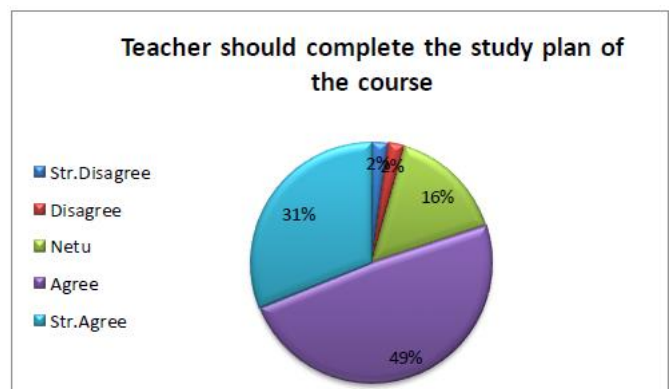


Figure 4: Study plan completeness Factor.

Some students did not review their lessons day by day. In addition, they only just lecture content without reference to other sources in order to get adequate information, this effects on their ability to adjust new programming questions. (see figure 5).

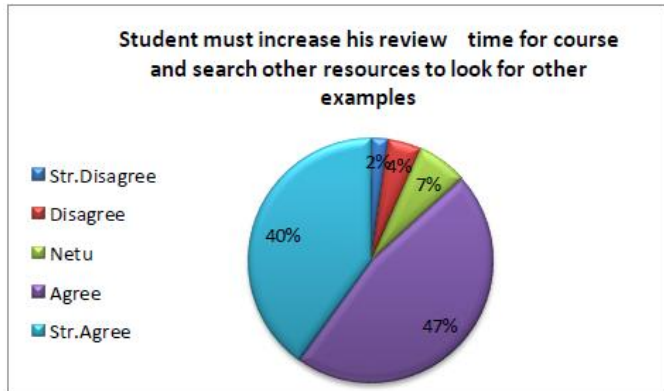


Figure 5: Extra examples and review time Factor.

Programming is fast becoming one of the most important subjects taught in schools in many countries. It preparing students for their future careers, developing their problem-solving skills, improving their creativity and building self-confidence. (see figure 6)

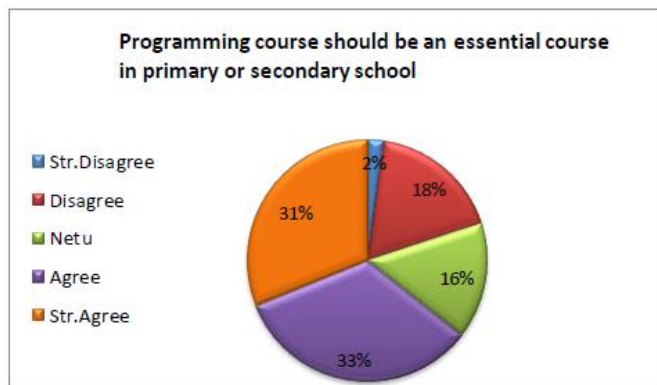


Figure 6: earlier programming course Factor.

Sometimes, computers in the lab are old or lack the capabilities to install certain large programs, such as mobile application development tools, affecting the practical aspect of educational activities. (see figure 7)

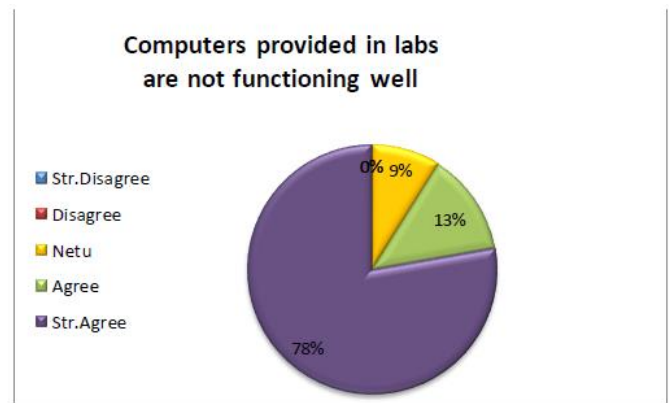


Figure 7: computer capability Factor.

Teaching methodology refers to the approach or method that a teacher uses to deliver educational content and guide students in the learning process. This includes selecting and organizing the curriculum, using teaching aids like visualization tools, and determining learning activities. (see figure 8)

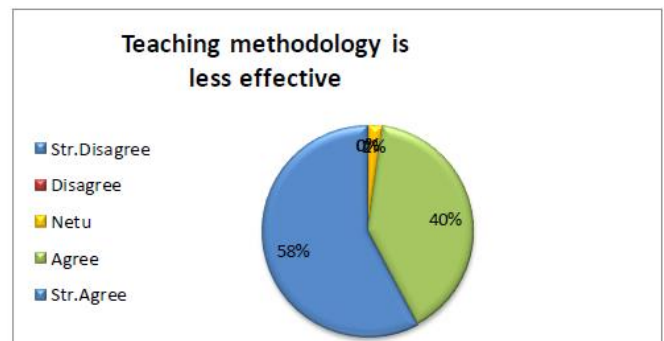


Figure 8: Teaching methodology Factor.

The most important programming language in college is the introductory programming language, as it encompasses all the concepts and skills needed for studying other languages such as problem-solving, algorithm, logic and understand syntax. Therefore, careful consideration is essential when choosing the first programming language. (figure 9)

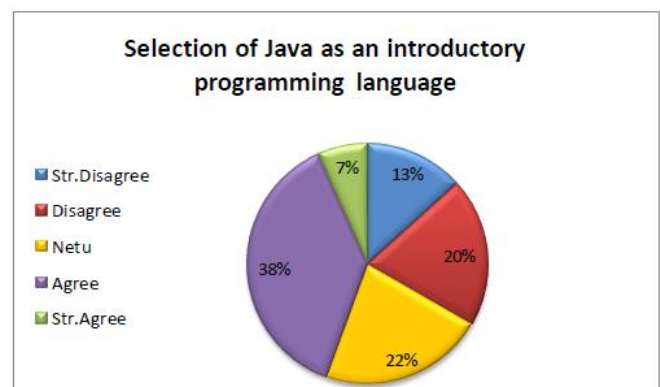


Figure 9: Introductory programming language Factor.

V. CONCLUSIONS

Programming courses are very important for the IT, IS, and AI department students. It is a technological process for telling a computer which tasks to proceed in order to solve problems. In this paper we intended to investigate and consider the reasons behind the weakness of some students in programming courses in Faculty of computer and Information Technology in Aden. We found that there are many reasons for that from student's side, course's side, environment's side and teacher's side. Some of these reasons are the short time for practical, students' English language is not good so it also affects their skills, lack of previous knowledge in programming, traditional teaching strategies, the big numbers of students in practical groups and lack of computers' capability used in laboratory.

VI. REFERENCES

- [1] AbdelRahman, S., AL-Syabi, B., Al Sharji, E., & Al Kaabi, S. S. (2016). *The Reasons behind the Weakness of some Students in Programming Courses in the College of Applied Science, Ibri*. Retrieved from MECS: <http://www.mecs-press.org/>
- [2] Alakeel, A. (2015). Investigating Difficulties of Learning Computer Programming in Saudi Arabia. *Universal Journal of Educational Research* 3(9), pp. 567-577.
- [3] Derus, S., & Ali, A. (2012). DIFFICULTIES IN LEARNING PROGRAMMING: VIEWS OF STUDENTS. *1 st International Conference On Current Issues in Education*, (pp. 74-79).
- [4] Gomes, A., & Mendes, A. (2007). Learning to program - difficulties and solutions. *International Conference on Engineering Education-ICEE*.
- [5] Harimurti, R., Ekohariadi, Munoto, Asto, G., & Winanti, E. (2019, May). ANALYSIS OF PROGRAMMING SKILLS CONCEPT IN DEVELOPING. *Jurnal Pendidikan Teknologi dan Kejuruan*, Vol. 25, No. 1, pp. 43-51.
- [6] Hawi, N. (2010). Causal attributions of success and failure made by undergraduate students. ScienceDirect.
- [7] Holvikivi, J. (2010). Conditions for Successful Learning of Programming Skills. *IFIP International Conference on Key Competencies in the Knowledge Society*, (pp. 155-164).
- [8] Khaleel, F., Ashaari, N., Wook, T., & Ismail, A. (2017, June). Programming Learning Requirements based on Multi Perspectives. *International Journal of Electrical and Computer Engineering (IJECE)*, p. 1299~1307.
- [9] Krpan, D., Rosić, M., & Mladenović, S. (2014, Jan). Teaching Basic Programming Skills to Undergraduate Students. *In Contemporary issue in economy and technology*.
- [10] Maryono, D., Budiyo, Sajidan, & Akhyar, M. (2022). The Analysis of Programming Problem Solving Skill of the students in Software Engineering Vocational High Schools. *Proceedings of the 3rd Asia Pacific International Conference on Industrial Engineering and Operations*, (pp. 13-15). Johor Bahru, Malaysia.
- [11] Othman, J., & Abdul Wahab, N. (2016). The Uncommon Approaches of Teaching the Programming Courses: The Perspective of Experienced Lecturers. *Journal of Computing Research and Innovation (JCRINN) Vol 1, No 1*.
- [12] Sarpong, K., Arthur, J., & Owusu, P. (2013, September). Causes of Failure of Students in Computer Programming Courses: The Teacher – Learner Perspective. *International Journal of Computer Applications*.
- [13] Sentance, S., & Csizmadia, A. (2015, June). Teachers' perspectives on successful strategies for teaching Computing in school. *IFIP TCS*.
- [14] Staff, c. (2023, 11 29). <https://www.coursera.org/articles/what-is-programming-in>. Retrieved from coursera.
- [15] S. Derus and A. Ali, "Difficulties in Learning Programming: Views of Students," in *1st International Conference on Current Issues in Education*, 2012, pp. 74-79.