RESEARCH ARTICLE | FEBRUARY 24 2025

Methods of teaching the basics of programming to bachelors of computer science at Bukhara State University

Gulsina Atayeva ✉; Gulbahor Turayeva; Fazilat Norova

AIP Conf. Proc. 3268, 070015 (2025)
https://doi.org/10.1063/5.0257421

Methods of Teaching the Basics of Programming to Bachelors of Computer Science at Bukhara State University

Gulsina Atayeva[a)], Gulbahor Turayeva[b)] and Fazilat Norova[c)]

Department of Information Systems and Digital Technologies, Bukhara State University, M. Iqbol street, 11, Bukhara, 200100, Uzbekistan

[a)]Corresponding author: g.i.ataeva@buxdu.uz
[b)]g.h.turaeva@buxdu.uz
[c)]f.f.norova@buxdu.uz

Abstract. In recent years, IT education has become the most in-demand in the labor market in all countries of the world. This actualizes the problem of training and attracting highly qualified specialists, which is one of the most difficult tasks not only for the employer but also for the state as a whole. Today, the whole world has been captured by the process of digitalization, which requires highly qualified specialists in the field of software technologies. Since teaching programming is a rather complicated process, the search for effective methods of teaching programming has been going on for many years. The article proposes a methodology for teaching bachelor's programming with the introduction of online platforms and artificial intelligence tools. Since these technologies have now received their deep development among other digital technologies they should be turned into faithful teaching assistants.

INTRODUCTION

In the era of building a New Uzbekistan, at the initiative of the President of Uzbekistan, the necessary conditions are being created for the widespread introduction of the digital economy and e-government. Thus, the Strategy "Digital Uzbekistan-2030" approved in 2020 sets priorities and defines measures to implement digital transformation: foreign investments are attracted, the quality of the national information and communication technologies (ICT) infrastructure is improved, the concept of a "smart" city in the regions is being implemented, the skills and competencies of specialists in the field of digital technologies are being improved. In this regard, professions related to the ability to program are becoming more in demand. Higher education institutions are tasked with providing high-quality programming training to graduates.

Programming is a specific area of professional activity. The training of IT specialists in this field requires a significant amount of time and differs significantly from any other activity. Like teaching any other practical activity, programming training requires significant real work in this field. A feature of programming is the fact that a future programmer must not only know the syntax of a programming language but also understand the possibility of implementing specific algorithms to solve specific problems. The constant connection between theory (the possibility of implementing an algorithm) and practice (writing a solution in a specific programming language) complicates the construction of the learning process.

According to a study by [1-5], teachers identified four main issues in computer programming education:

1. Students lack the skills needed to analyze problems and understand programming concepts. This is the first issue encountered in teaching programming, as students lack the necessary skills such as a course in Discrete Mathematics and Logic Programming.
2. The inefficient use of presentation techniques when solving problems also contributes to a poor understanding of programming concepts. Traditional methods, such as Pseudocode and Flowcharts, are only suitable for structured programming but cannot provide adequate explanations and understanding in object-oriented programming.

Proceedings of the IV International Conference on Advances in Science, Engineering, and Digital Education
AIP Conf. Proc. 3268, 070015-1–070015-6; https://doi.org/10.1063/5.0257421
Published under an exclusive license by AIP Publishing. 978-0-7354-5128-5/$30.00

3. Most programming languages are currently taught based on object-oriented principles, which can make it difficult for students to understand the concepts effectively [5].
4. There is a lack of practical experience for students in programming, as they do not have the opportunity to apply their knowledge in real-world situations. This can lead to a lack of confidence and motivation in the subject. To provide students with the opportunity to mentally visualize the problem, it is necessary to adopt approaches that ensure greater clarity in explanation [6].

Another challenge arises from the inefficiency of programming learning strategies. This learning strategy must be tailored to long-term objectives. Traditional teaching methods are no longer suitable for teaching object-oriented programming, as teachers agree that different paradigms (imperative and declarative) should be employed to promote cognitive strategy.

In addition to providing support for spatial and visual abilities, learning materials are necessary to help students understand the process of control and data flow. Furthermore, the lack of student participation and engagement during practical sessions further exacerbates this problem, leading to students' inability to comprehend computer programming during the learning process. The last reason why students fail to learn programming is because they do not understand or master programming syntax [7], [8]. This problem is further compounded when they are also unable to program complex structures. This can be explained by their lack of understanding of the concept of programming and their inability to create efficient program code [6].

Currently, various online courses, video tutorials, and code validation systems can be used to teach programming. If relevant queries are searched in search engines, a significant portion of the resources will also refer to mobile software products. Therefore, programming training can take place within the framework of web-based platforms (online courses, Massive Open Online Courses (MOOCs)), mobile applications, collections of instructional videos, and other thematic resources.

The article discusses the feasibility of teaching programming using an interactive approach at the initial stages, using online platforms that enhance the learning experience. The digitalization of the education process offers the potential for the utilization of appropriate cutting-edge technologies, allowing students to have complete control over their search and processing of information. The foundation of digital transformation lies in technological progress, which drives the changes in the epochs of civilization. A special responsibility rests with the education system, which we need to systematically develop and enhance.

MATERIALS AND METHODS

The article discusses the methodology used for teaching programming at the initial stages of higher education in Bukhara State University, in the fields of information technology. It is based on the principles:

1. Programming training within the context of professional training for bachelor's degrees should aim to develop a holistic set of competencies, including motivation and needs for professional growth, knowledge and skills related to key concepts in object-oriented programming, and a specific object-oriented language.
2. Training should consist of two parts: the study of an object-oriented language, which provides a basic understanding of the concepts of object-oriented programming.
3. The study of basic concepts of object-oriented programming should be conducted within the framework of a separate discipline at the beginning, using the C++ programming language as it supports the imperative programming paradigm. The foundation for establishing such a discipline would be the exploration of visual programming environments that utilize the C++ language.
4. Teaching methods and techniques should promote student engagement, involving the creation and implementation of challenging tasks, as well as the organization of training sessions through specialized online platforms. Students would solve problems using artificial intelligence and then analyze the results to make necessary adjustments to the program code. While doing so, students would comment on the compiled output.
5. Assessment of students' programming skills should be based on the writing of small programs in memory and then running them on specialized online platforms, such as HackerRank and Codecademy. Students can also evaluate their skills by analyzing completed projects throughout the course.

In 2020-2023, we developed a new methodology at Bukhara State University. This method was used in programming classes for bachelor's students in the specialty "Information Systems and Technologies" as part of the "Fundamentals of Programming" course (1-2 semesters). This course aims to study the basics of both procedural and object-oriented programming. It is based on modeling a real-life process or object. Since this course is called "Fundamentals of Programming", the first semester begins with simple programming concepts, such as variables and

operators. In the second semester, we had a smooth transition into the study of the conceptual issues of object-oriented programming. This included the description and identification of objects, classes, and methods. We studied the basics of object-oriented programming by presenting code on an electronic whiteboard and then parsing the code. After explaining the topic, students used online compilers for the C++ programming language to type the code themselves and parse the program.

From the very beginning of the course, we modeled objects and their relationships by integrating elements of object design. This was reflected in the course's representation of objects as program code. This technique has been used since the beginning of the academic year 2020/2021. In the academic year 2021/2022, changes were made to the curriculum because the incoming generation of students had little knowledge of programming. As a result, the topics in the curriculum were further clarified so that students could understand the subject more deeply.

The lectures were conducted using interactive presentations that explained the meaning of programming codes and the HackerRank online platform. This platform provides solutions to tasks at the initial level. HackerRank is a popular online platform for programming that offers a wide variety of tasks and challenges. It is a valuable resource for anyone wanting to improve their programming skills and become a professional.

When studying object-oriented programming in the second semester, basic concepts were covered using the Codecademy online platform.

The main advantage of using an online programming platform is that it allows you to not only learn the basics of programming but also get an assessment of your programming skills, which is very motivating for students. Codecademy's ease of use is also a great advantage. All the lessons are divided into logical units that help you clearly structure your learning. You can select lessons on specific topics, or take a complete course in a particular programming language. Every lesson is accompanied by an explanation of the theory, as well as practical exercises that will help you consolidate your knowledge. To date, artificial intelligence chatbots like GPT have been actively used in programming studies. Modern students enjoy the possibilities of modern information technologies and, as a result, have a strong desire to use AI. However, this also has advantages. Students who are tasked with creating a specific program must necessarily explain how it works, so using a pre-made program allows students to figure out the ins and outs. The author believes it is not worthwhile to hinder such programming studies, and indeed it is impossible to do so! To analyze the methodology described above, an experiment was conducted with students from 1st to 2nd year of the Information Systems and Technology program. An experimental group of students were selected who studied using the new methodology, while a control group studied using the traditional method.

RESULTS

The results were recorded every month. It was found that students in the experimental group had the highest rate of success, reaching 62.3%. In the control group, the rate was 59.8%.

To compare the academic performance of both groups, a certification exam was given at the end of every month for 6 months. The average score was then calculated as a percentage and plotted on a graph. The scores on the exam as a percentage represented the coordinates for the graph in Fig. 1. The starting value was set to zero.

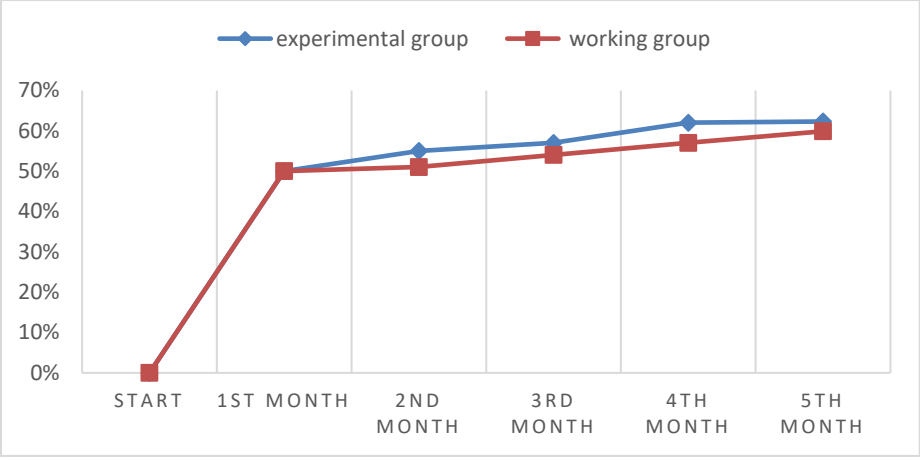

FIGURE 1. The Results of the Experiment on Academic Performance in Programming.

When comparing the results of programming mastery in the experimental group with those of the control group, it was observed that changes to the teaching methodology resulted in positive outcomes.

To determine the efficacy of the novel components incorporated into the teaching methodology using online platforms, a study was conducted – teaching programming using HackerRank, which produced satisfactory results in enhancing students' academic achievement in programming at initial training stages. A comparison was made with the certification grades of first-year students in the academic years 2022–2023 and 2023–2024 (Fig. 2). The progression graph of students in programming is given in Fig. 3.

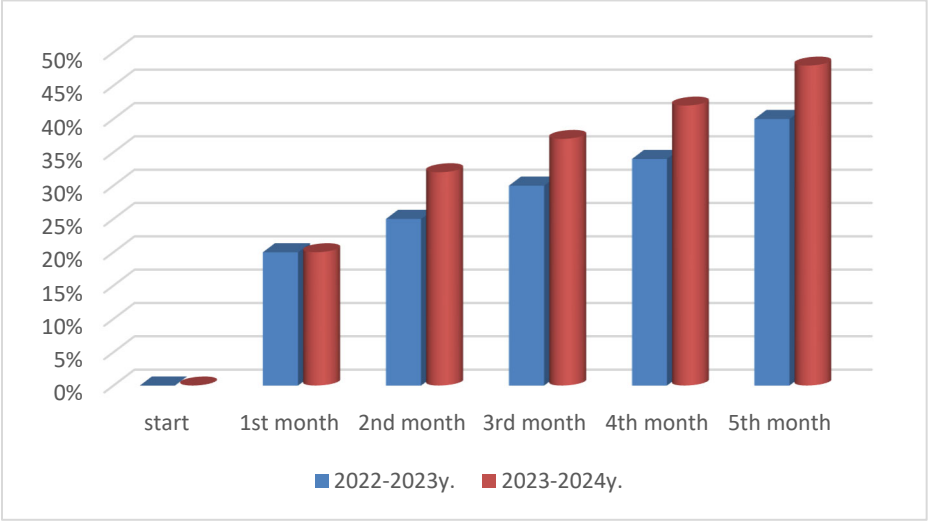

FIGURE 2. Assessment assessments for the 2022-2023 and 2023-2024 academic years.

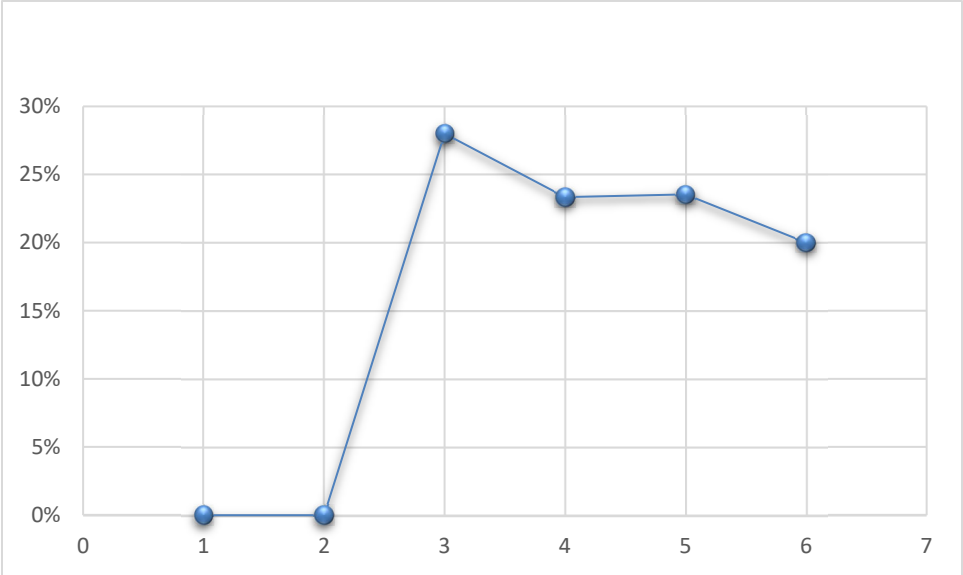

FIGURE 3. Graph of students' academic performance in programming.

The calculations were carried out according to the following formula:

$$PG = (PY - TY)/ PY,$$

here:
PG - percentage of growth,
PY - previous year,
TY - this year.

As the graph shows, this online platform provides better learning at the initial stages of programming, but with continued use, student academic performance does not decrease, only the rate of improvement declines. This suggests that it may be worthwhile to look for another platform with a greater impact, as the programming course becomes more complex due to the transition to object-oriented coding. Thus, the implementation of a developed methodology for teaching basic programming and basic object-oriented programming in the educational process for bachelor's degree students in the specialty of information systems and technology using online programming learning platforms will ensure an increase in the competency level of students in this field.

Experiments conducted to improve this methodology have confirmed that modern students simply need to learn programming through innovations in the learning process. Additionally, the implementation process should be gradual, starting with simple and gradually moving on to more complex topics, which will help increase the professional training level of students studying computer science.

DISCUSSION

Training in software development is included in the educational programs in various fields of training in computer science according to higher education standards. After analyzing the curricula of Bukhara State University for bachelor's degrees in the areas "Information Systems and Technologies," "Computer Science and Programming Technology," and "Software Engineering," we found that in all three, tasks related to algorithmic and programming-learning programming languages, developing and analyzing - are highlighted as part of the structure of professional activities. Algorithms, software development, and mobile application development are all highlighted in this structure. Depending on the field of training, emphasis on programming training can be given more or less, but in either case, there are competencies in programming related to professional competence in general or in a specific field. At the same time, despite considerable attention paid to programming training in various educational programs, there are still problems in this area. These include the lack of consistency between educational methods and the current level of development in programming technologies, which leads to poor quality graduate training. Our experiment has shown that the overall level of training for students at the end of their academic course can only be described as satisfactory. Students in the IT field do not demonstrate a high level of knowledge in professional programming and lack sufficient motivation to learn programming languages and develop new software systems. In many ways, this situation is caused by the low level of school knowledge in programming, which is not enough to study professional programming languages at university (82.7% of students surveyed mentioned this problem). Therefore, teaching programming at university boils down to forming basic ideas about concepts and approaches to programming while learning a professional language at the same time. This makes the programming course extremely difficult for students, and it doesn't contribute to the development of motivation, a deep and systematic understanding of programming techniques and professional languages.

The solution to this problem lies in the development of a new approach to teaching programming at the beginning of university training. When analyzing existing methods, several authors were mentioned. S.D. Karakozov argues that advanced training means learning modern programming technologies, specifically web development, so that it meets the requirements of the modern labor market and international standards in IT curricula, as well as the recommendations of modern theories and research in computer programming teaching. In addition, this is done at every stage in the educational process: from curriculum development to assessing students' competencies [2].

From a pedagogical perspective, teachers often focus on teaching the syntax of programming languages rather than promoting problem-solving methods. Additionally, the choice of programming language for teaching often depends on popularity rather than pedagogical suitability. Selecting an inappropriate programming language for education not only reduces students learning of computer programming but also makes it more difficult for them to understand the material. Programming requires a high level of abstract thinking and analytical skills to create effective solutions. These skills are best developed through the application of specific programming concepts and algorithms, which is why choosing programming languages based on their industrial popularity is not ideal. The chosen programming language should be easy to learn and understand, with fewer complexities and easier self-explanation to facilitate the learning process at the initial stage [4].

CONCLUSION

Thus, the introduction into the educational process of the developed methodology for teaching the basics of programming and the basics of object-oriented programming to bachelors of the specialty information systems and

technologies using online programming learning platforms ensures an increase in the level of competencies of bachelor of the specialty Information Systems and technologies in the field of programming. The experiments conducted to improve the methodology of teaching programming confirm that modern students simply need to master programming with the help of innovations in the learning process. Moreover, the process of implementing training using online platforms should be phased, from simple to complex, which will increase the level of professional training of students studying in the field of computer science.

REFERENCES

1. C. S. Cheah, Contemporary Educational Technology **12(2)**, 272 (2020).
2. S. D. Karakozov et al., European Proceedings of Educational Sciences **1** (2020). https://www.doi.org/10.15405/epes.20121.9
3. G. Atayeva, Universum: technical sciences: electron. Scientific magazine **3(120)** (2024). https://7universum.com/ru/tech/archive/item/17019
4. F. V. Shkarban, Development of a methodology for teaching the basics of object-oriented programming using visual learning environments. Scientific notes of the Crimean Federal University named after In And Vernadsky. Sociology. Pedagogy. Psychology **4(70)**, 52-58 (2018).
5. M. N. Ismail, N. A. Ngah and I. N. Umar, TOJET: The Turkish Online Journal of Educational Technology **9(2)** (2010). http://tojet.net/articles/v9i2/9214.pdf
6. A. Robins, The Cambridge Handbook of Computing Education Research, Cambridge Handbooks in Psychology, 327-376 (2019). Retrieved from https://www.otago.ac.nz/computer-science/otago706761.pdf
7. Y. Bosse and M. A. Gerosa, ACM SIGSOFT Software Engineering Notes **41(6)**, 1-6 (2017). https://doi.org/10.1145/3011286.3011301
8. Y. Qian and J. Lehman, ACM Transactions on Computing Education (TOCE) **18(1)**, 1-24 (2017). https://doi.org/10.1145/3077618