MUHAMMAD AL-XORAZMIY
NOMIDAGI TATU FARG'ONA FILIALI

FERGANA BRANCH OF TUIT
NAMED AFTER MUHAMMAD AL-KHORAZMI

"AL-FARG'ONIY AVLODLARI"

ELEKTRON ILMIY JURNALI | ELECTRONIC SCIENTIFIC JOURNAL

TA'LIMDAGI ILMIY, OMMABOP VA ILMIY TADQIQOT ISHLARI

3-SON 1(3)
2023-YIL

TATU, FARG'ONA
O'ZBEKISTON

ISSN 2181-4252

O‘ZBEKISTON RESPUBLIKASI RAQAMLI TEXNOLOGIYALAR VAZIRLIGI

MUHAMMAD AL-XORAZMIY NOMIDAGI TOSHKENT AXBOROT TEXNOLOGIYALARI UNIVERSITETI FARG‘ONA FILIALI

Muassis: Muhammad al-Xorazmiy nomidagi Toshkent axborot texnologiyalari universiteti Farg‘ona filiali.
Chop etish tili: O‘zbek, ingliz, rus.
Jurnal texnika fanlariga ixtisoslashgan bo‘lib, barcha shu sohadagi matematika, fizika, axborot texnologiyalari yo‘nalishida maqolalar chop etib boradi.

Учредитель: Ферганский филиал Ташкентского университета информационных технологий имени Мухаммада ал-Хоразми.
Язык издания: узбекский, английский, русский.
Журнал специализируется на технических науках и публикует статьи в области математики, физики и информационных технологий.

Founder: Fergana branch of the Tashkent University of Information Technologies named after Muhammad al-Khorazmi.
Language of publication: Uzbek, English, Russian.
The magazine specializes in technical sciences and publishes articles in the field of mathematics, physics, and information technology.

2023 yil, Tom 1, №3
Vol.1, Iss.3, 2023 y

ELEKTRON ILMIY JURNALI **ELECTRONIC SCIENTIFIC JOURNAL**

«Al-Farg‘oniy avlodlari» («The descendants of al-Fargani», «Potomki al-Fergani») O’zbekiston Respublikasi Prezidenti administratsiyasi huzuridagi Axborot va ommaviy kommunikatsiyalar agentligida 2022-yil 21 dekabrda 054493-son bilan ro’yxatdan o’tgan.

Tahririyat manzili:
151100, Farg‘ona sh., Aeroport ko‘chasi 17-uy, 201A-xona
Tel: (+99899) 998-01-42
e-mail: info@al-fargoniy.uz
Qo‘lyozmalar taqrizlanmaydi va qaytarilmaydi.

FARG’ONA - 2023 YIL

TAHRIR HAY'ATI

Jurnal quyidagi bazalarda indekslanadi:

MUNDARIJA | ОГЛАВЛЕНИЕ | TABLE OF CONTENTS

USING VISUAL LEARNING ENVIRONMENTS IN TEACHING OBJECT-ORIENTED PROGRAMMING

Abdurakhmonov Sultonali Mukaramovich,
TUIT Fergana branch, Faculty of Telecommunication Technologies and Vocational Education, Head of the Department of Information and Educational Technologies, Candidate of Physical and Mathematical Sciences, Associate Professor.
sulton59_15@inbox.ru

Ibragimov Shavkat Mamirovich,
Ferghana State University, Faculty of Mathematics - Informatics, Senior Lecturer of the Department of Information Technologies.
shavkat70@bk.ru

Abstract: The article deals with the problem of teaching object-oriented programming. The paper proposes guidelines for teaching students object-oriented programming using the capabilities of various programming languages, and developing computer programs based on them. Teaching the basics of object-oriented programming makes it possible to bring the teaching of a programming course to a more modern level and expands the possibilities of future specialized courses, as well as eliminates the contradictions between the current state of programming and the content of teaching the discipline in higher educational institutions.

Keywords: object-oriented programming, algorithm, visual learning environment, methods, programming language, analysis, teaching methodology.

Introduction. Modern society, which has entered the information age, is characterized by the active use of information technology in all areas of its activity. These technologies, providing human information activity, are based on the use of a variety of computer technology and software. At the same time, it is software tools that determine the algorithms and logic of working with information, allow the use of standard computer technology to solve a wide variety of problems, and most significantly affect the capabilities and quality of implemented information technologies. In this regard, the problem of teaching the creation of software for specialists in the field of information technology is relevant, which involves the formation of students' competencies to solve professional problems in the development of projects for automation and informatization of applied processes, as well as the creation of information systems in applied areas. The implementation of this part of the training is carried out within the framework of certain disciplines, where knowledge of software tools and programming languages is formed, the ability to use various strategies for developing and implementing computer programs, experience and skills in solving problems in the field of programming.

The purpose of the study is to develop and scientifically substantiate a methodology for teaching the basics of object-oriented programming for bachelors using visual learning environments.

To achieve the goal of the study, the following research objectives were formulated:

1. Clarify the role and place of teaching object-oriented programming in the system of bachelor's education.
2. Identify the essential characteristics of the bachelor's competence in the field of object-oriented programming.
3. To develop the target, content and procedural components of the methodology for teaching the basics of object-oriented programming for bachelors using visual learning environments.
4. To experimentally substantiate the effectiveness of the methodology for teaching the

https://doi.org/10.5281/zenodo.8319524

https://journals.indexcopernicus.com/search/article?articleId=3712793

https://al-fargoniy.uz/

basics of object-oriented programming for bachelors using visual learning environments.

Literature analysis and methods. Traditional teaching of object-oriented programming usually includes complex tasks and questions that are difficult for students to perceive, which leads to further misunderstanding of the educational material. Learning object-oriented programming by traditional methods is based on learning the syntax of a particular programming language that is not of interest to students or is not clear to master. At the same time, such training at the stage of university training is based on the knowledge and skills of students obtained in the study of the relevant sections of computer science in a secondary school. As shown by the data of many studies (A.N. Bobrov [1], O.G. Nelzina [2], M.A. Pavlichenko [3]), as well as the results of our ascertaining experiment, these knowledge and skills are not enough for the subsequent learning professional programming languages at the appropriate level. In this regard, there is a need to study the basics of object-oriented programming at the stage before learning professional programming languages in order to provide fundamental training of students in the field of the methodology itself, the basic concepts of object-oriented programming. This requires the development of an appropriate methodology for teaching the basics of object-oriented programming, based on the results of research on the process of teaching object-oriented programming, the use of design technologies and the creation of modern computer programs.

There are a number of studies that are of particular value in this area, which include studies in the field of professional training of information technology specialists (V.V. Andreeva [4], I.E. Vostroknutov [5] and others), the formation of professional qualities in students in the areas of training in applied informatics (V.Yu. Bodryakov [6], A.A. Bykov [7]), research that reveals the essence of teaching programming (J. McConnell [8]).

Simultaneously with the theoretical prerequisites, the practical prerequisites for teaching the basics of object-oriented programming as the initial stage of professional training of specialists in the field of computer program development were formed. In particular, these are practical developments related to the use of visual learning environments for such training. Among these environments, Alice and Scratch occupy a special place, created as educational products for teaching programming. The results of the practical application of these environments, presented in the works of Randy Pouch, Wand Dunn, Stephen Cooper, Mitchell Resnick, convincingly show that their use allows you to avoid many of the problems of initial programming learning. Visual learning environments allow students to be motivated, to focus on the development of thinking, understanding the very essence of the programming paradigms being studied.

To achieve the goal, you can use the following methods:

- *theoretical* (analysis, comparison, classification, systematization, generalization) - analysis of scientific, psychological, pedagogical and methodological literature on the problem, study of the experience of teaching programming and features of the practical training of future bachelors;
- *empirical* (questionnaire, observation, interviewing, method of expert assessments) - to determine the level of professional training of university students;
- *pedagogical experiment* - to test the effectiveness of the proposed methodological system;
- *methods of mathematical statistics* - for analyzing the data obtained, determining quantitative indicators for the phenomena and processes under study.

Results and discussion. Modern society is characterized by a number of features, among which one of the most significant places is the assertion of the role of information technology as a leading means of improving intellectual, labor, leisure and many other types of human activity. In the most general form, information technologies are understood as processes, methods for searching, collecting, storing, processing, providing, distributing information and methods for implementing such processes and methods using computer technology.

The task of training citizens in the field of information technology as one of the main ones is solved in the education system. In our country, the solution of this problem is connected with the implementation of the educational field of informatics at all levels of education.

Thus, ensuring computer literacy has consistently transformed into the task of forming an information culture, and now - the information competence of students.

Programming training is included in the educational programs of various areas of training in the

https://doi.org/10.5281/zenodo.8319524
https://journals.indexcopernicus.com/search/article?articleId=3712793

field of information technology according to the standards of higher education.

So, after analyzing the undergraduate areas "Applied Mathematics and Informatics", "Information Technologies in Education", we can say that in all of them, in the structure of tasks of professional activity, there are tasks related to algorithmization and programming - the study of programming languages, the development and analysis of algorithms , software development. Depending on the areas of training, the emphasis on teaching programming can be placed to a greater or lesser extent, but in any case, in the structure of professional competence in the general professional or professional part, there are competencies related to the development of computer programs.

The problem of training specialists in the field of information technology is also relevant in European countries, where it is controlled by the European organization Career Space. Career Space was created with the support of the European Commission (European Commission), nine leading European companies in the information technology industry. The activity of this organization is aimed at formulating requirements for the competencies of university graduates in the relevant educational standards. These requirements are formulated on the basis of an analysis of the results of an expert survey of leaders of leading companies operating in the information and communication technology sector.

Among the world projects proposed for teaching computer science, one can also single out the Computing Curricula project, which is used in the field of training IT specialists of various levels.

The programming language is for the programmer the basis of his activity. However, knowledge of one language is not enough, a specialist must understand various programming paradigms.

Teaching programming with elements of *imperative programming* is a traditional approach that involves learning a single procedural programming language. But the developers of the program note that an object-oriented programming language can also be used - if training is carried out using an object-oriented programming language, then at the first stage, attention should be focused on the imperative aspects of this language (expressions, control structures, procedures and functions, and others). key elements of the traditional procedural model), and object-oriented programming technologies are studied in the next stage.

The *Object-Oriented Programming* Education Orientation also focuses on programming, but emphasizes the principles of object-oriented design and programming from the very beginning.

The *functional programming* approach involves the use of functional programming languages. This model is aimed at studying and getting acquainted with the programming language only in higher educational institutions and does not depend on the student's prior training. The clear syntax of functional languages allows the teacher to focus on the fundamental principles and issues of programming. In functional programming, the entire set of sequential states of the computational process is represented explicitly, which makes it possible to study these issues at the very beginning of learning to program.

Learning programming with elements of *maximum coverage of the material* allows students to gain a broader understanding of programming, which enables them to move on, confidently and purposefully, mastering other computer disciplines. Students during training have the opportunity before the start of the course to evaluate the variety of topics in the curriculum of the discipline and only after that proceed to the traditional cycle of studying programming.

The *algorithm-oriented approach* to learning programming minimizes the effort spent on learning the specific syntax constructs of a particular programming language by using a graphical way of describing algorithms and pseudocode. This enables students to work with a wide range of data types and control logic structures. After students master the basic types of algorithms and data types, they can begin to use the programming language in practice.

Learning to program with elements of a *hardware component* is learning from the machine level. Only after the students have formed an understanding of the structural features of the hardware component, machine logic and mathematics, the course proceeds to consider programming in high-level languages.

An analysis of modern strategies for teaching programming shows that the orientation of teaching to object-oriented programming differs from all the considered models and provides a number of advantages that other models do not provide. Object-oriented programming is characterized by a combination of fundamentality and deep practical orientation, which allows students to develop an object-

https://doi.org/10.5281/zenodo.8319524
https://journals.indexcopernicus.com/search/article?articleId=3712793
https://al-fargoniy.uz/

oriented style of algorithmic thinking, a clear sequence of actions, imaginative and logical thinking, as well as qualities and skills that allow direct development of software applications.

Teaching the basics of object-oriented programming, introducing the basic concepts and concepts of the object-oriented approach are important problems in the preparation of bachelors. In this regard, there is a need to develop an appropriate teaching methodology aimed at the formation of professional competence in the field of object-oriented programming and realizing one of the meaningful lines of informatics in the professional training of bachelors. Such a technique should be based on the use of specialized software and, as a result, assume the study of a professional language.

The choice of a programming language from which to start training in the system of vocational education is an important task facing educational institutions in our country. The task is not simple, because, on the one hand, it is necessary to take into account the incomplete theoretical and practical base of a first-year student, and on the other hand, the need for further study of more complex disciplines that are part of a particular specialty in the field of computer science.

According to the frequency of queries in search engines according to the TIOBE index, the most popular languages in this ranking include Java, C, C++, Python, C#, PHP, JavaScript. At the same time, in the rating of programming paradigms, also compiled by TIOBE, the object-oriented approach is the undisputed leader. Another programming language ranking project is Ohloh, which is determined by how often a programming language is mentioned on the Internet. According to this rating, the leading languages are C (C++), Java, JavaScript, Objective-C, PHP, Python.

The software tools for teaching object-oriented programming should include not only professional tool environments, but also specialized visual learning environments, which, being an integrated development environment with a simple interface, allow you to visualize the model of the program being created, execute methods of object classes in the process of writing them, test object classes, setting various parameters for methods as they are written. Such environments, most famous in the academic environment, include Alice and Scratch.

Alice and Scratch allow you to work with objects and methods, as well as develop programs in terms of objects, classes and their interactions. These environments are focused on mastering the fundamental concepts of object-oriented programming, fixing them in the practice of programming using a specific language, such as C ++. This allows you to choose exactly these environments for the initial training in object-oriented programming for bachelors of applied computer science.

Alice and Scratch allow you to manipulate objects with their properties, functions, and methods, both built-in and user-constructed. These visual learning environments minimize keyboard input. The program code is not text in the usual sense: within one method, it is a set of nested blocks highlighted in color depending on the type (loops, conditional jumps, etc.), they can be collapsed, dragged, deleted, reordered, etc. .

The use of a visual programming environment in the learning process allows you to:

- imagine and imagine what exactly can be done and obtained as a result of programming;
- create a project based on your own ideas;
- share the results of their activities;
- reflect and discuss their own results;
- to develop more complex and interesting projects.

In addition, visual environments for teaching programming form the skills of a future programmer to work in a team - students in teams discuss the problem of solving a problem, determine the level of their own knowledge and readiness to solve a problem, and show a lack of knowledge in certain issues.

In the process of teamwork using visual learning environments, students learn to assess the problem and their own level of preparation, analyze the problem, look for possible solutions to problems, formulate and express their thoughts, convince colleagues, share knowledge with each other and work collectively to solve the problem, plan their own activities, coordinate their actions with team members, present the results of their work. Practical experience in this study shows that students, using visual programming learning environments in practice, are able to independently and quite effectively acquire knowledge and form skills and abilities. This contributes to the formation of an integral system of professional competencies for bachelors of applied informatics.

Conclusions. From the foregoing, we can conclude that programming is a key component of the bachelor's training system, the main task of which is to

https://doi.org/10.5281/zenodo.8319524
https://journals.indexcopernicus.com/search/article?articleId=3712793
https://al-fargoniy.uz/

provide a theoretical basis and practical skills in the professional training of students. Therefore, knowledge of programming is a prerequisite for the assimilation of most professional disciplines of bachelor's programs in applied computer science, because:

- this area of knowledge consists of those concepts that are important for the practice of programming, regardless of the programming paradigm that is used in the educational process;

- successful assimilation by students of the content of professional disciplines and, as a result, their future professional activity depends on the quality of assimilation of the educational material of the programming course;

- programming should be studied in junior courses, before studying all disciplines of the professional cycle;

– teaching programming for bachelors of applied informatics is advisable to start with learning the C ++ language, using special environments for teaching programming.

Such training from the very beginning should focus on the basics of object-oriented programming, as this makes it possible to bring the teaching of a programming course to a more modern level and expands the possibilities of future specialized courses, and also eliminates the contradictions between the current state of programming as a science and the content of teaching this discipline in higher educational institutions.

REFERENCES

1. Бобров, А.Н. Проблемы выбора языка программирования в школьном курсе информатики / А.Н. Бобров // Молодой ученый. – 2015. – №24. – С. 61–64. URL: https://moluch.ru/archive/104/24471/

2. Нельзина, О.Г. Проблемы обучения программированию по курсу информатики в системе «школа-вуз» / О.Г. Нельзина // RELGA – научно-культурологический журнал (№13 [135]. – 2006. URL: http://www.relga.ru/Environ/WebObjects/tgu-www.woa/wa/Main?textid= 1087&level1=main&level2=articles.

3. Павличенко, М.А. Проблема обучению программированию в школе / М.А. Павличенко // Социальная сеть работников образования nsportal.ru. – 2014. URL: https://nsportal.ru/shkola/obshchepe dagogicheskie-tekhnologii/library/2014/10/19/problema-obucheniyu-programmirovaniyu-v

4. Андреева, В.В. Проектирование и реализация системы многоуровневой подготовки специалистов в области информационных технологий: дис. ... д-ра пед. наук : 13.00.08 / Андреева Валентина Владимировна. – Новгород, 2005. – 375 с.

5. Вострокнутов И.Е., Калинина О.С. Применение метода программных фрагментов при обучении визуальному программированию на базовом уровне в средней школе / И.Е. Вострокнутов, О.С. Калинина // Педагогическая информатика. – 2014. – № 1. – С. 22–31.

6. Бодряков, В.Ю. Практический опыт формирования исследовательских компетенций студентов, обучающихся по направлению «01. 03. 02 – Прикладная математика и информатика» / В.Ю. Бодряков, Л.Р. Ушакова // Педагогическое образование в России. – 2015. – №7. URL: https://cyberleninka.ru/article/n/ prakticheskiy-opyt-formirovaniya-issledovatelskih-kompetentsiy-studentov-obuchayuschihsya-po-napravleniyu-01-03-02 prikladnaya.

7. Быков, А.А. Особенности реализации компетентностного подхода при подготовке специалистов прикладной информатики и вычислительной техники на базе смоленских вузов / А.А. Быков // Электронный научно-практический журнал «Современная педагогика». – 2014. – №10. URL: http://pedagogika.snauka.ru/ 2014/10/2753.

8. Макконелл, Дж. Анализ алгоритмов. Вводный курс / Дж. Макконелл. – Москва: Техносфера, 2002. – 304 с.

https://doi.org/10.5281/zenodo.8319524

https://journals.indexcopernicus.com/search/article?articleId=3712793