ISNN 3030-3907

2026
MAY
ISSUE - 5
VOLUME - 4

DEVELOPMENT OF SCIENCE

scientific journal



Development of science

Ilmiy jurnal

MAY 2026
Issue - 5, VOLUME - 4

ISSN 3030 -3907

Ilmiy jurnal OAK rayosatining 2024 yil 27 sentabrdagi 361-son qaroriga asosan 02.00.00 kimyo fanlari hamda 2025 yil 12 fevral 367- son qaroriga asosan 05.00.00 texnika fanlari bo‘yicha falsafa doktori (PhD) va fan doktori (DSc) ilmiy darajasiga talabgorlarning dissertatsiya ishlari yuzasidan dissertatsiyalari asosiy ilmiy natijalarini chop etish tavsiya etilgan milliy ilmiy nashrlar ro‘yxatiga kiritilgan.

BOSH MUHARRIR:
Temirov Alisher Hoshim o‘g‘li – texnika fanlari falsafa doktori (PhD),Dotsent.
TAHRIRIYAT KENGASH RAISI:
Fozilov Sadriddin Fayzullayevich – texnika fanlari doktori, professor
TAHRIRIYAT KENGASH A’ZOLARI

TEXNIKA FANLARI/ TECHNICAL SCIENCES/ ТЕХНИЧЕСКИЕ НАУКИ

Bozorov G‘ayrat Rashidovich
Buxoro Davlat Texnika Universiteti, professor
Abdurahmonov Olim Rustamovich
Buxoro Davlat Texnika Universiteti, professor
Adizov Bobur Zamirovich
O‘zbekiston Respublikasi Fanlar akademiyasi Umumiy va noorganik kimyo instituti, professor
Hayitov Ruslan Rustamjonovich
Buxoro Davlat Texnika Universiteti, professor
Jumayev Qayum Karimovich
Buxoro Davlat Texnika Universiteti, professor.
Berdiyev Dorob Murodovich
Islom Karimov nomidagi Toshkent Davlat Texnika Universiteti, professor
Maxmudov Rafiq Amonovich
Buxoro Davlat Texnika Universiteti, professor.
Tilloyev Lochin Ismatilloyevich
Buxoro Davlat Texnika Universiteti, dotsent.
Raxmonov Qahramon Sanakulovich
Buxoro Davlat Texnika Universiteti, dotsent.
Barakayev Nusratullo Rajabovich
Islom Karimov nomidagi Toshkent Davlat Texnika Universiteti, professor.
Jo‘rayev Xayrullo Fayziyevich
Buxoro Davlat Texnika Universiteti, professor.
Panoyev Erali Rajabboyevich
Buxoro Davlat Texnika Universiteti, dotsent.
Ataullayev Sherzod Nabiullyevich
Buxoro Davlat Texnika Universiteti, dotsent.
Murodov Malik Negmurodovich
Buxoro Davlat Texnika Universiteti, dotsent.
Axmedova Ozoda Bahronovna
Islom Karimov nomidagi Toshkent Davlat Texnika Universiteti.
Safarov Bahri Jumayevich
Buxoro Davlat Texnika Universiteti, dotsent.
Xaydarov Axtam Amonovich
Buxoro Davlat Texnika Universiteti, dotsent.
Safarov Jasur Alijon o‘g‘li
Buxoro Davlat Texnika Universiteti, dotsent.

KIMYO FANLARI/ CHEMICAL SCIENCES/ ХИМИЧЕСКИЕ НАУКИ

Axmedov Voxid Nizomovich
Buxoro Davlat Texnika Universiteti, professor
Umarov Baqo Bafoyevich
Buxoro Davlat Universiteti, professor
Kodirov Abduaxad Abduraximovich
Qarshi davlat Universiteti, professor
Qurbonov Mingniqul Jumagulovich
Qarshi davlat Universiteti, dotsent
Turemuratov Sharibay Naurizbaevich
O‘zbekiston Respublikasi Fanlar akademiyasi Qoraqalpog‘iston bo‘limi Bosh ilmiy kotibi, professor
Tursunov Kaxor Shonazarovich
Qarshi davlat Universiteti, professor
Do‘stov Hamro
Buxoro Davlat Texnika Universiteti, professor
Panoyev Nodir Shavkatovich
Buxoro Davlat Texnika Universiteti, dotsent.
Olimov Bobur Bahodir o‘g‘li
Buxoro Davlat Texnika Universiteti, dotsent.
Kasimov Sherzod Abduazimovich
Termiz Davlat Universiteti, professor.
Aliqulov Rustam Valiyevich
Termiz Davlat Universiteti, professor
Tropov Xamza Tursunovich
Samarqand Davlat Universiteti, professor

IQTISODIYOT FANLARI/ ECONOMICS/ ЭКОНОМИЧЕСКИЕ НАУКИ

Nizamov Aslitdin Badreddinovich
Buxoro Davlat Texnika Universiteti, professor.
Boboyev Akmal Choriyevich
Buxoro Davlat Texnika Universiteti, professor.
Ergashxojayeva Shaxnoza Djasurovna
Toshkent davlat iqtisodiyot universiteti, professor
Saidaxmedova Nodira Ilxomovna
Toshkent davlat iqtisodiyot universiteti, professor
Muxsinov Bekzod Toxirovich
Buxoro Davlat Texnika Universiteti, dotsent.
Shakarov Qulmat Ashirovich
Renessans ta'lim universiteti,dotsent.
Jumaeva Zulfiya Qayumovna
Buxoro Davlat Texnika Universiteti, dotsent.
Norova Salomat Yusupovna
Buxoro Davlat Texnika Universiteti, dotsent.
Avezova Shaxnoza Maxmudjonovna
Buxoro Davlat Texnika Universiteti, dotsent.
Jalilov Jamshid G‘anijonovich
Toshkent davlat iqtisodiyot universiteti, dotsent
Xasanova Gulruh Djumanazarovna
Buxoro Davlat Texnika Universiteti, dotsent.

PEDAGOGIKA FANLARI/ PEDAGOGICAL SCIENCES / ПЕДАГОГИЧЕСКИЕ НАУКИ

Siddiqova Sadoqat G‘afforovna
Buxoro Davlat Texnika Universiteti, dotsent.
Zoyirov Erkin Xalilovich
Buxoro Davlat Texnika Universiteti, DSc, professor
Gaffarov Feruz Hasanovich
Puchon Universiteti, *professor*
Ruziyev Davron Yuldashevich
Buxoro Davlat Universiteti, professor
Muradova Firuza Rashidovna
Buxoro Davlat Texnika Universiteti, professor.
Yunusova Gulandon Samiyevna
Buxoro Davlat Texnika Universiteti, professor.
To‘xtayeva Zebo Sharifovna
Buxoro Davlat Texnika Universiteti, professor.
Navro‘zova Gulchixra Nigmatovna
Buxoro Davlat Texnika Universiteti, dotsent.
Uzoqova Laylo Polvonovna
Buxoro muhandislik – texnologiya instituti, dotsent.
To‘raqulova Marjona Qiyom qizi
Buxoro Davlat Texnika Universiteti, dotsent.
Muradova Zarina Rashidovna
Buxoro Davlat Texnika Universiteti, dotsent.
Murodov Sanjar Aslonovich
Buxoro Davlat Texnika Universiteti, dotsent.
Shodiyev Jahongir Jo‘raqulovich
Buxoro Davlat Texnika Universiteti, dotsent
Gaffarov Laziz Hasanovich
Buxoro muhandislik – texnologiya instituti, dotsent.

Tahririyat a’zolarning to‘liq ro‘yxati jurnalning www.devos.uz saytida keltirilgan.

Tahririyat hay’ati a’zolarining tarkibini o‘zgartirish hamda qo‘shimcha kiritish yoki tahririyat a’zoligidan chiqarish bosh muharrir tomonidan amalga oshiriladi.

ISSN 3030-3907

Development of science ilmiy jurnali 2024 yil 15 aprel kuni №256445 sonli guvohnoma bilan Davlat ro‘yxatidan o‘tgan.

To‘plam elektron ko‘rinishda (pdf formatida) mualliflarga taqdim etiladi. To‘plamga kiritiladigan maqolalarning mazmuni, ulardagi statistik ma’lumotlar va me’yoriy hujjatlarning to‘g‘riligiga hamda har qanday fikr - mulohazalarga mualliflarning o‘zlari mas’ul hisoblanadilar. Talablarga javob bermaydigan maqolalar to‘plamga kiritilmaydi. Tashkiliy qo‘mita maqola matnini qisqartirish, qisman tuzatish kiritish, seksiyalarga joylashtirish huquqiga ega.

ISSN 3030-3907

ACTUAL PROBLEMS OF TEACHING PROGRAMMING LANGUAGES IN MODERN CONDITIONS

Otahanov Nurillo Abdumalikovich,
Namangan state university, professor, DSc.
E-mail: nurillo_otaxanov@list.ru

Abstract. In the context of the digitalization of society and the economy, the demand for rapid processing of large volumes of information is steadily increasing. This trend intensifies the need for training adaptable and competitive specialists who possess in-depth knowledge, skills, and competencies in the field of programming. Therefore, teaching modern programming languages plays a significant role in enhancing the professional competence of specialists. Educational programs related to information processing include a number of disciplines focused on programming languages; however, their effective instruction requires a reconsideration of teaching goals and objectives from a modern perspective, as well as the search for solutions to existing challenges. This article substantiates the role of programming languages in the training of specialists whose professional activities are associated with large-scale data processing and revisits approaches to their instruction in higher education. Furthermore, the goals and objectives of teaching programming languages in a digitalized society are analyzed from a contemporary standpoint, key problems are identified, and practical recommendations for addressing these issues are proposed. The research findings will be useful for researchers and educators dealing with problems related to programming language education.

Keywords: programming language; problems of teaching programming; programming activity; specifics of educational directions; teaching programming language.

Annotatsiya. Jamiyat va iqtisodiyotning raqamlashtirilishi sharoitida katta hajmdagi axborotni tezkor qayta ishlashga bo‘lgan talab izchil ravishda ortib bormoqda. Mazkur tendensiya dasturlash sohasida chuqur bilim, ko‘nikma va kompetensiyalarga ega bo‘lgan moslashuvchan hamda raqobatbardosh mutaxassislarni tayyorlash zaruratini kuchaytiradi. Shu bois zamonaviy dasturlash tillarini o‘qitish mutaxassislarning kasbiy kompetentligini oshirishda muhim ahamiyat kasb etadi. Axborotni qayta ishlash bilan bog‘liq ta’lim dasturlari dasturlash tillariga oid qator fanlarni o‘z ichiga oladi, biroq ularni samarali o‘qitish ta’lim maqsad va vazifalarini zamonaviy nuqtayi nazardan qayta ko‘rib chiqishni hamda mavjud muammolarga yechim topishni talab etadi. Mazkur maqolada kasbiy faoliyati katta hajmdagi ma’lumotlarni qayta ishlash bilan bog‘liq bo‘lgan mutaxassislarni tayyorlashda dasturlash tillarining o‘rni asoslab berilgan hamda oliy ta’lim tizimida ularni o‘qitish yondashuvlari qayta

ISSN 3030-3907

ko‘rib chiqilgan. Shuningdek, raqamlashtirilgan jamiyat sharoitida dasturlash tillarini o‘qitishning maqsad va vazifalari zamonaviy nuqtayi nazardan tahlil qilinib, asosiy muammolar aniqlangan hamda ularni hal etish bo‘yicha amaliy tavsiyalar ishlab chiqilgan. Tadqiqot natijalari dasturlash tillarini o‘qitish muammolari bilan shug‘ullanuvchi tadqiqotchilar va pedagoglar uchun foydali bo‘lishi mumkin.

Kalit so‘zlar: dasturlash tili; dasturlashni o‘qitish muammolari; dasturlash faoliyati; ta’lim yo‘nalishlarining o‘ziga xosligi; dasturlash tilini o‘qitish.

Аннотация. В условиях цифровизации общества и экономики устойчиво возрастает потребность в оперативной обработке больших объёмов информации. Данная тенденция усиливает необходимость подготовки адаптивных и конкурентоспособных специалистов, обладающих глубокими знаниями, умениями и компетенциями в области программирования. В связи с этим обучение современным языкам программирования играет важную роль в повышении профессиональной компетентности специалистов. Образовательные программы, связанные с обработкой информации, включают ряд дисциплин, посвящённых языкам программирования, однако их эффективное преподавание требует переосмысления целей и задач обучения с современных позиций, а также поиска решений существующих проблем. В данной статье обоснована роль языков программирования в подготовке специалистов, профессиональная деятельность которых связана с обработкой больших массивов данных, а также пересмотрены подходы к их преподаванию в системе высшего образования. Кроме того, с современных позиций проанализированы цели и задачи обучения языкам программирования в условиях цифровизированного общества, выявлены ключевые проблемы и предложены практические рекомендации по их решению. Результаты исследования могут быть полезны исследователям и преподавателям, занимающимся проблемами обучения языкам программирования.

Ключевые слова: язык программирования; проблемы обучения программированию; программная деятельность; специфика образовательных направлений; обучение языкам программирования.

1. INTRODUCTION.

Nowadays, there is an increasing immersion of society and the economy into the digital world. This world possesses a powerful potential for further modernizing society and the economy and integrating into all spheres of human activity; allows for rapid response to everything happening in the world; enhances efficiency, speed, effectiveness, economy, and quality of fulfilling job responsibilities; provides comfortable living and working conditions; ensures the

ISSN 3030-3907

mobility of all kinds of activities. In the context of digitalization, low-skilled specialists inevitably face the prospect of losing their jobs, as modern digital technologies acquire more and more capabilities for performing their functional duties quickly, efficiently, and effectively.

The growing importance of the speed of processing large volumes of informational data in a digital society has increased the demand for multidisciplinary specialists who possess competencies in various disciplines, including computer science and ICT. Therefore, the formation and improvement of the process of training well-rounded and professionally competent specialists, depending on the specifics of the educational direction, become the main task for any university. At the forefront is the degree of professional flexibility, competitiveness, and mobility of future specialists.

The job duties of most specialists whose activities are related to processing large volumes of information involve performing a range of functions, each of which entails either processing large amounts of data or promptly responding to requests and events happening in the world and society. In the context of the digitalization of society and the economy, the role of computer and software tools is increasing, as they collectively contribute to performing the aforementioned functions.

Currently, there are many software tools that allow improving the quality and conditions of specialists' work. However, new tasks that require reflection in computer and software tools continue to arise. There is a wide range of tasks for which corresponding software was created at one time, which nonetheless requires modernization considering new realities. It is no secret that software tools, being one of the important driving forces of the digital society, are created either through the joint efforts of programmers and experts in the field of programmed tasks or through the sole efforts of highly qualified specialists of various professions possessing sufficient programming skills. In the joint creation of software tools, there are often numerous disagreements between programmers and experts due to programmers not being sufficiently informed about the specifics of the programmed tasks and experts about the intricacies of programming. Ultimately, these disagreements can lead to a decrease in quality, prolonged development time, and increased costs associated with creating the software product. When creating software products, specialists who are highly qualified in the field of programmed tasks and possess sufficient programming skills can easily eliminate such problems.

In this regard, it can be said that teaching modern programming languages (PL) is of paramount importance for increasing the level of professional competence of specialists. Considering this phenomenon, higher education

ISSN 3030-3907

institutions have begun to implement educational policies aimed at training highly qualified, competitive, and quickly adapting personnel to modern conditions. In line with this policy, several disciplines related to the teaching of computer science, ICT, and PL have been included in the curricula of educational directions, whose future activities are related to processing large volumes of information. Modern PLs (such as C#, Python, etc.) support many programming paradigms. They are multifunctional, allow developing software tools for a wide range of tasks, have a vast set of tools, and are widely used in global programming practice. Therefore, many educational directions in universities include the training of these languages.

In the process of teaching PLs, several problems arise related to the large number of tools, usage options, and forms of data and instruction representation. Another problem arises from the lack of scientifically and methodologically justified educational technologies that contribute to increasing the efficiency of the educational process.

Object of Research: The process of teaching programming languages in higher education institutions.

Subject of Research: Defining the goals and objectives of teaching programming languages, identifying the most relevant problems arising in the process of learning programming languages and studying them, as well as developing recommendations for solving these problems.

2. LITERATURE REVIEW.

The analytical review is based on existing scientific-methodological literature and research works dedicated to general education problems and the teaching of programming languages (PL) in higher education institutions. In the vast majority of the existing scientific-methodological literature, authors used a distinctive style of presenting materials on PL. An analysis based on sources such as “C#: The Complete Reference” by H. Deitel [4], “C# 4: The Complete Reference” by H. Schildt [24], and “Fundamentals of Programming with C#” by S. Nakov [14], and “Python 3: Standard Libraries with Examples” by D. Hellmann [10] showed that the didactic principles “from simple to complex” and “from known to unknown” remain out of the authors' sight. A large amount of the proposed material should be recognized as morally outdated. Despite the grand titles and impressive volume, [4, 14, 24] lack information about the graphical capabilities of the C# language. In our opinion, [10] on Python and the IDLE environment lacks information for beginner programmers, resulting in insufficient useful information. For these reasons, in modern conditions, using these sources in the process of teaching programming languages cannot ensure high efficiency.

ISSN 3030-3907

E. Troelsen, in his book "C# and the .NET Platform. Programmer's Library," provided more detailed information about the capabilities of the C# language. It is worth noting that the author, using the example of working with images, described in detail the namespaces GDI and System.Drawing, the classes Point, Rectangle, Size, Region, and working with colors and fonts, as well as with ready-made images [28, 1197-1215]. However, the didactic principles "from simple to complex" and "from known to unknown" are also not implemented here, complicating the process of mastering the proposed materials.

In open sources, there are numerous scientific studies dedicated to general education problems. In our opinion, the following works deserve special attention. M. Forrath examined the international context of the quality issue of higher education and its adaptation specifics in Russia. He conditionally divides the problems of higher education quality into three parts: the first part consists of the general context, the second part touches on the analysis of the representations of university teachers and administrators, and the third part highlights the basics of state policy in the field of education quality [7]. A.V. Lubkov researched modern problems of pedagogical education and considered the content and essential factors of modernizing pedagogical education [13]. A.V. Tomiltsev and A.V. Maltsev studied the problems of assessing the professional preparedness level of graduates based on methodological approaches. They described various approaches to assessing education quality and the evaluation procedure based on basic tools, as well as developed criteria, parameters, indicators, and norms for use in selecting evaluation benchmarks [26].

Many scientists conduct research in identifying and addressing problems related to teaching informatics, among which we highlight the following. E.K. Khenner considers IT education a factor influencing gender disparities in computing. Comparing the experience of Russia and the USA, he notes significant differences in the distribution of men and women across professions among university teachers: women are much more numerous among professors in the humanities and fewer in technical, mathematical, and natural sciences, including information technology and programming [12]. R.A. Abramov and other scientists studied issues of increasing the professional demand for university graduates and identified factors important for the successful employment of graduates and the practical application of acquired knowledge and skills [1].

Many scientists conduct research on identifying and addressing problems related to teaching programming languages. Among them, the works of such scientists as A.A. Horovik, A.I. Zagitova, I.A. Oblomov [15], V.Yu. Pirogova [18], and D.Kh. Fayziyeva can be highlighted. In particular, A.A. Horovik considers the lack of motivation and technical equipment in classes as a serious

ISSN 3030-3907

obstacle to the effectiveness of classes [11]. A.I. Zagitova researches programming education problems and ways to improve programming courses in universities and believes that the problem-oriented method has a positive impact on the academic performance results of students studying programming courses [31]. D. Fayziyeva associates the problems of teaching PL with the individual abilities of students and believes that "beginners in programming often do not realize their own shortcomings. Also, the personal qualities of students affect their performance" [5]. M. Foychik believes that there are several problems in teaching PL for both teachers and students. Teachers' problems lie in the need to choose the most effective approach among various approaches to the learning process, programming skills, and teaching qualifications. Students' problems lie in forming motivation and approaches to mastering PL [6]. R.S. Shahamiri analyzed problems arising at each stage of the software creation lifecycle and believes that they are serious issues in learning PL [23]. Saad M. A. investigated the factors affecting students at the College of Applied Sciences in Ibri and believes that due to poor English proficiency, students face some problems while taking programming courses [22].

Many scientists conduct research on improving the methodology of teaching PL, contributing to the quality of programming language education. Among them, the studies of such scientists as autor [16], K.V. Rozov [21], X. He [9], P. Patil [17], A.A. Grishin [8], P. Szlavi [25], T. Chen [3], and B. Qarshiyev [20] can be noted. In particular, B. Wang and Y. Liu developed and implemented a practical course on studying PL C# case studies to improve students' ability to analyze and solve applied tasks [29]. Autor recommended the approximate content of PL Python education, considering the specifics of bachelor's educational directions in the Republic of Uzbekistan [16].

Based on the above analysis, it can be concluded that research only touches on individual questions of the examined issues. Research that unites the whole complex of problems of teaching PL is currently not presented in open information sources. Therefore, serious scientific-methodological research is needed that combines the entire spectrum of organizational and methodological problems of improving the effectiveness of PL teaching.

3. METHODOLOGY, MATERIALS, AND METHODS.

3.1. Goals and Objectives of Teaching Programming Languages. In the context of the digitalization of society and the economy, especially in those sectors where employees' professional duties are related to processing large volumes of information, the demand for professionally mobile, highly qualified, competitive, and quickly adapting specialists possessing a set of knowledge, skills, and abilities in various disciplines, including programming, increases. In

ISSN 3030-3907

this regard, programming languages (PL) as one of the most important sections of computer science become one of the priority disciplines that contribute to increasing the readiness level of future specialists for professional activities. PL as an object of study and as a means of training should be included in the preparation process for specialists whose job duties will be related to processing large amounts of information or developing new and modernizing existing software applications. The content of disciplines involving the use of programming methods and tools should be professionally oriented, allowing students to gain practical experience and prepare them for the independent improvement of programming skills and abilities.

Therefore, rethinking the goals and objectives of PL in modern conditions, considering the specifics of educational directions, will contribute to improving the quality and effectiveness of the educational process.

In our opinion, the goal of teaching PL in universities should be defined as follows: to contribute to preparing specialists who, in addition to general professional competencies, have sufficient systematic knowledge, skills, and abilities for the targeted use of the tools and possibilities of PL in accordance with their job duties.

To achieve this goal, the main focus should be on solving the following organizational and educational-methodological tasks:

– rethinking the role and place of PL in the modern world and in the personnel training process, and making corresponding changes to the regulatory documents of educational institutions;

– creating printed and electronic (including cloud-based and visualized) educational-methodological resources on modern PL, considering the specific features of educational directions;

– analyzing the possibilities of modern educational technologies and adapting the most suitable ones, considering the features of teaching PL;

– studying advanced methodologies of prestigious foreign universities and integrating them into domestic universities while harmonizing existing teaching traditions;

– teaching students the basic structural elements of modern PL;

– mastering the basic programming paradigms by students;

– developing students' skills and abilities in the principles of software development;

– developing students' skills and abilities in the basics of algorithmizing the processing of information of various forms and content, corresponding to the specifics of educational directions;

ISSN 3030-3907

– improving students' knowledge for the targeted and rational use of the tools and possibilities of PL;

– mastering methods for creating new and modernizing existing software tools by students, considering changes in regulatory acts and modern requirements.

To effectively solve the set goals and objectives, the following are required:

1) a deep understanding by students that programming skills and abilities will contribute to a clear understanding and penetration into the essence of the studied objects, phenomena, and processes and will become an essential tool in professional activities;

2) equipping university classrooms with modern technical, software, and didactic teaching tools;

3) prompt resolution of difficulties that create serious problems in the process of teaching and learning modern PL.

3.2. Problems of learning Programming Languages. It is known that the history of modern programming began in the 1950s. Over this period, around 9000 programming languages (PL) have been developed, and about 50 of them are used in modern programming practice. In the mid-20th century, a new scientific discipline—cybernetics—emerged and developed, and in the 1960s-1970s, informatics branched out from cybernetics as an independent scientific discipline. The development of teaching informatics and PL, as one of its important sections, faces numerous problems caused by changes in attitudes towards informatics, paradigm shifts, and changes in the demand for specialists in specific PL.

In our opinion, the following problems are the key obstacles to improving the effectiveness of the modern PL teaching process.

3.2.1. Demand for Specialists with Programming Skills. Firstly, in the context of the digitalization of society and the economy, the requirements for the overall readiness level of graduates for professional activities are changing. Secondly, the sectors requiring specialists with programming skills are changing. Thirdly, in global programming practice, the status of PL changes quickly depending on their demand. Fourthly, the content and volume of useful information on PL increase due to frequent updates of their versions. The educational policy simply does not keep up with such phenomena.

It is known that the employment of graduates plays a special role in determining the prestige of universities. For graduates with skills in specific PL, employment directly depends on the demand for specialists in that category. The demand for specialists in the field of programming can be determined by the following parameters:

ISSN 3030-3907

availability of job vacancies in specific PL in the labor market;
– the number of jobs requiring programming competencies;
– the expansion of the application area of programming knowledge;
– salary offerings;
– changes in the dynamics of PL relevance.

The variable nature of the above parameters does not allow for unequivocal decisions calculated for the long term.

Recommendations:

1) Establish constant monitoring of the domestic and global labor market and programming practices in accordance with the above parameters.

2) Based on regularly collected data, develop annual specific proposals to improve the regulatory documents of educational directions and take necessary measures to integrate these changes into the educational process, planned for the next 3-4 years of training.

3.2.2. Provision of Methodically Developed Literature. As known, educational literature on PL is created either by leading programmers or pedagogical researchers. Books written by leading programmers contain a large amount of information aimed at revealing the full potential of PL. Examples of such books are [4, 10, 14, 24, 28]. Modern languages (especially C#) have expanded to such an extent that covering detailed information about them in one book has become a challenging task. Usually, such books are released as monographs, reference books, or manuals. Learning PL based on such materials is challenging for beginner programmers, as they ignore the basic didactic principles of assimilating educational information. These resources provide good opportunities for expanding the knowledge of programmers of medium and high qualification. However, beginner programmers can easily get lost in the informational "jungle." Therefore, without denying the overall significance of the works of the mentioned authors, we believe that such resources are more beneficial for the first two categories of programmers rather than beginners.

It should also be noted that educational literature created by pedagogical researchers usually does not cover a large volume of data on PL (for example, [2, 19]). However, it is more accessible to students and beginner programmers since educational materials are presented in a systematic manner, based on the principles of "from simple to complex" and "from known to unknown." Such textbooks and educational manuals are usually created based on a specific methodology and in a language understandable and accessible to young programmers. Nowadays, there is generally insufficient educational literature that considers all methodological aspects of learning PL.

ISSN 3030-3907

Recommendations:

1) We consider the following structure of educational literature on PL appropriate:

1.1. A brief annotation;

1.2. Basic language constructs (general program view; input and output operators, branching, looping; data arrays; functions; graphics; file handling) for beginners;

1.3. Exception handling, libraries, working with new data types and references for the intermediate level;

1.4. Classes, containers, interfaces, and object-oriented programming for professionals.

2) Information should be presented in strict logical sequence and comply with all didactic principles (especially the main rules of the principle of accessibility: "from known to unknown" and "from simple to complex").

3) The volume of literature should be minimized.

4) Since creating and releasing printed resources takes more time, effort, and resources, preference should be given to developing educational-methodological literature based on cloud technologies and visualization.

3.2.3. Complexity of Programming Language. Competitor companies are steadily working on improving and expanding the capabilities of PL based on modern requirements. As a result, new versions of these languages constantly appear. N. Wirth, in his article "Computing science education: The road not taken" [30] compares many modern PL to "machines packed with various necessary and unnecessary devices" and writes that "in luxurious cars, packed with the latest technologies, gadgets, and attractions, it is already difficult to see the difference – as difficult as determining the point at which the technology, whose purpose was to assist the driver, becomes a hindrance to driving... These things do not contribute much to the true purpose of the car, but they add complexity, complicate maintenance, and increase cost." N. Wirth transfers this example to the field of PL and says that "they have become immensely complex not so much because all their wonderful tools and capabilities were really needed, but simply because they were possible. Therefore, manufacturers hope that these tools and capabilities will provide an advantage over competitors. However, the advantage leads to excessive bulkiness, difficulty in use, and reduced reliability." This leads to an increase in the volumes of information on PL, the emergence of difficulties and uncertainties in their practical application, as well as to the increasing confusion of the material, which seriously complicates the learning process. N. Wirth asserts that the "wonderful" capabilities of many languages are nothing more than advertising and writes that "unnecessary, artificial complexity generates many problems. People mistakenly take complexity for sophistication".

ISSN 3030-3907

The complexity of a language directly depends on the variability of data and construct representations. In other words, a PL with a large number of tools and possibilities for expressing the same instruction is considered more complex (according to N. Wirth's logic).

Recommendations:

1) The volume of transmitted educational information should be minimal and necessary but sufficient to study the basics of PL.

2) Avoiding unnecessary complexity in the language, the educational material should be built on the basis of the simplest and most widely used variants of instructions, while other variants can be offered for independent study.

3.2.4. Selection and Content of the Programming Language for Learning. As known, the content and volume of each planned subject (including disciplines related to teaching PL) are determined based on state standards and qualification requirements for the professional training of future specialists, depending on the specifics of educational directions. The knowledge acquired in these disciplines guarantees the graduate's full readiness for independent professional activity. In particular, programming skills and abilities will help future specialists solve professional tasks arising within their job responsibilities.

As we remember, around 50 programming languages, supporting various paradigms, are used in modern global programming practice. Some of them, as universal languages, are designed to create software tools for a wide range of tasks. Others, as domain-specific languages, aim to solve a narrow range of tasks. Therefore, the question of how to choose PL for different educational directions remains open.

Each modern PL has immense possibilities and a wide set of tools for creating software applications. Therefore, teaching the full potential of a language is considered an insurmountable task. In light of this, another problem arises: determining the minimally necessary but sufficient content of educational material on PL. The urgency of this problem is intensified by the set of competencies that graduates of a particular educational direction must possess.

Recommendations:

1) The choice of PL should be based on the following parameters: specificity of the educational direction; ease of mastering; universality; practical orientation; popularity; independence [27].

2) For educational directions whose curricula include only one subject related to PL, we propose Python. For educational directions involving the development of complex software complexes, we suggest studying several languages supporting different programming paradigms (e.g., C#, SQL, and Python).

ISSN 3030-3907

3) The content of educational material (including the set of tasks used in practical, laboratory, and independent classes) should correspond to the specifics of educational directions [16].

3.2.5. Insufficiency of Research Works. An analysis of open sources, the example of the website dissercat.com showed, that showed that there are few dissertation works dedicated to improving the quality of modern PL education, and the number of defended works after 2020 is even smaller. In our opinion, the following factors are serious obstacles to conducting research aimed at improving the quality of programming language education: excessive bureaucracy; a small number of funded works; changes in the requirements for modern classes; shifts in the paradigms of organizing classes; trends in the development of methodology, educational technologies, and technical and software tools; parameters mentioned in point I; frequent updates of language versions, and others.

Recommendations:

1) Limit bureaucracy to reasonable limits, considering the specifics of teaching PL.

2) Increase the share of funded research works and projects.

3) Conduct monitoring of promising educational technologies and advanced foreign methodologies with further adaptation and implementation in the process of teaching programming languages.

3.2.6. Inability to Use Programming Language Tools. The main source of this problem is not the syntax of programs or the understanding of programming concepts but the ability to create a logical chain of language tools and events guaranteeing the expected results. A student may know the types and rules of data conversion, the basic system of operators, PL concepts (e.g., creating new data types), but still not be able to use them purposefully, skillfully, and correctly in the process of creating new programs. Students may understand the syntax and semantics of individual expressions but not know how these expressions integrate into the programs they create. As noted by A. I. Zagitova, "To help students effectively study both programming concepts and strategies for their application, a 'problem-oriented approach' was developed. According to this approach, students are introduced to PL functions only in the context of solving specific tasks. It has been shown that this method positively impacts the academic performance of students studying programming courses and increases their confidence in their knowledge" [31].

Recommendations:

1) It is necessary to pay more attention to teaching students the basic methods of algorithmization, designing the software creation process, and effective debugging and testing of code.

ISSN 3030-3907

2) Introduce new PL concepts and tools in the context of real-world tasks they may face within their future job responsibilities.

3.2.7. Insufficient Awareness of Instructors about All Capabilities of the Programming Language. Changes in the prestige of PL, frequent updates of versions, the vast range of tools and capabilities, and trends in paradigms can lead to insufficient awareness of instructors in disciplines related to teaching PL. The history of teaching PL shows that until the 1960s, the primary language of instruction was ASSEMBLY, in the 1970s-1980s it was ALGOL and FORTRAN, in the 1990s it was BASIC, at the end of the 20th century it was PASCAL, at the beginning of the 21st century it was C (or C#), and today it is Python. There is a noticeable trend: almost every decade, the language changes, but the teaching staff changes insignificantly. This means that instructors of "old PL" adapt to new languages. Since the main advantages of previous languages are preserved in the creation of new ones, the range of possibilities expands, and the ways of presenting data and instructions change, instructors generally have the opportunity to promptly stay informed about new transformations in PL.

Our analysis of the PL process in three universities showed that after announcing a new topic, about 15% of students are already partially familiar with the new material and only need to expand their knowledge on this topic. Such students can put instructors in an awkward position.

Recommendations:

1) Organize short-term professional development courses involving leading programmers aimed at expanding instructors' knowledge of new PL.

2) Require PL instructors to regularly undergo professional development courses with subsequent certification.

In addition to the above, it should be noted that there are methodological problems directly affecting the productivity of each class, the level of which completely depends on the implementation of a whole set of requirements. Among them are the following:

a) Planning appropriate forms and methods of teaching;

b) Stimulating students to assimilate new material;

c) Creating and using methodically developed lecture texts;

d) Adhering to didactic, ethical, and aesthetic principles;

e) Presenting educational material in a systematic, accessible, and simple form with strict logical sequence;

f) Presenting educational materials considering the preparedness level of the student audience for accepting new educational resources;

g) Reasonably distributing resources (time, educational materials, technical and software tools, etc.) and managing them.

ISSN 3030-3907

Violation of any of these requirements can lead to serious problems in teaching PL. As a result, these problems can hinder the achievement of educational, training, and developmental goals set for the current class. Furthermore, such situations can lead to a loss of enthusiasm in students, loss of motivation, and interest in studying the material.

4. EXPECTED RESULTS.

In the course of the research, we proceeded from the fact that the process of teaching PL would be organized at a proper level using modern technical and interactive teaching tools. Meanwhile, the above-mentioned problems negatively affecting the degree of efficiency in learning and teaching processes are discovered. Solving these problems and providing sufficient motivation to the participants of the educational process allows achieving the following results:

– Increasing the efficiency of the educational process;
– Enhancing the readiness of students for independent professional activities;
– More complete achievement of educational, developmental, and instructional goals of the educational process;
– Mastery of all competencies specified in the educational standards and qualification requirements of educational directions;
– Forming systematic knowledge about modern methods and programming languages, their place, and role in the training of specialists;
– Expanding horizons and deepening concepts and programming skills;
– Mastery of the main forms and methods of creating and using software tools;
– Forming students' skills in algorithmizing typical tasks within their future professional activities;
– Forming skills of interdisciplinary integration competence;
– Ability to correctly and reasonably select language tools and methods that contribute to the quick and effective creation of software applications;
– Developing students' ability for systematic, logical, and abstract thinking;
– Improving students' computational, algorithmic, and informational culture;
– Ability to find optimal ways to solve problems;
– Identifying primary and secondary factors in constructing processes and phenomena;
– Ability to create optimal plans;
– Possessing competencies in individual and collective work, and so on.

Assessing the scientific and practical significance of the problems identified in the field of teaching programming languages and the recommendations developed to address them is a challenging task when relying solely on narrowly focused studies or the research of a single scholar. The

ISSN 3030-3907

expected outcomes of this study depend on a range of factors, including the specific programming language, approaches to teaching programming languages, the characteristics of educational programs, the pedagogical competence of instructors, and many other parameters. To ensure an objective evaluation of the results, it is necessary to implement the proposed recommendations in long-term and diverse pedagogical research conducted by multiple scholars, followed by a comprehensive analysis of the outcomes.

Nevertheless, considering the generalized nature of numerous studies conducted by researchers in the field of programming language education and the successful mitigation of typical problems encountered in the teaching process, it can be reasonably assumed that the proposed recommendations are capable of ensuring the expected results with a high degree of probability.

5. DISCUSSION AND CONCLUSION.

In the digitalization era, employers highly value graduates who not only can work on a computer but also have programming skills. Such specialists actively engage in processing large volumes of information, where their programming knowledge clearly manifests, considering the entire spectrum of their job responsibilities. Therefore, universities must actively integrate PL into the process of training specialists and address the aforementioned problems related to it.

Only then will students, as future specialists, acquire not only general cultural and professional competencies but also the ability to understand the patterns of economic, social, and other processes, identify their cause-and-effect relationships, create economic-mathematical models for typical professional tasks, and develop software tools for their implementation.

Thus, teaching programming languages should become one of the fundamental parts of the process of training specialists whose professional activities involve processing large volumes of information and who are considered the driving force of the digital society and economy.

REFERENCES

1. Abramov R.A., Pronina I.V., Khalatenkova E.Yu. Increase of demand for graduates of higher education institutions. *The Education and science journal.* (10) 2016:91-106. DOI: 10.17853/1994-5639-2016-10-91-106. (In Russian.)
2. Aripov M.M., Otahanov H.A. *Basics of programming.* Tashkent: Tafakkur gulshani. 2016. P. 240. (In Uzbek.)
3. Chen T., Hsiao T., et al. Learning Programming Language in Higher Education for Sustainable Development: Point-Earning Bidding Method. *Sustainability.* 12(11). 2020. 4489: P. 14. DOI: 10.3390/su12114489.
4. Deitel H.M. *C# 6 for programmers.* Boston, USA, Pearson Education Inc. 2017. P. 771.

ISSN 3030-3907

5. Fayzieva D.H. Analysis of the problems of studying and teaching programming in computer science lessons. *Universum: technical sciences: electron scientific journal*. 12-1(93). 2021. 83-85. (In Russian.)
6. Fojcik M., Fojcik M. K., Hoyland S., Hoem I. O. Challenges in teaching programming. *Education and New Developments.* 3(4). 2022. 160-163. DOI: 10.36315/2022v1end 034.
7. Forrath N. The problem of the quality of higher education: global challenges and their Russian transformations. *Educational Studies Moscow.* 2(December). 2010. 121-38. URL: https://vo.hse.ru/article/view/15257. (In Russian.)
8. Grishin A.A., Sokolov A.N., Mindorov N.I., Romashkina T.V. Information system to support the process of learning the C# programming language for the discipline "Computer Science and basics of programming". *International research journal.* 3(45). 2016. 8-11. DOI: 10.18454/IRJ.2016.45.063.
9. He X., Song Y., Du Y., Huang Y., Yin X., Long X. Research on Teaching Reform of C Programming Language Course for Deep Integration of Professional Fields. *Open Journal of Social Sciences.* 10(1). 2022. 267-275. DOI: 10.4236/jss.2022.101022.
10. Hellmann D. *The Python 3: Standard Library by Example*. Boston, USA, Addison-Wesley. 2017. P. 1454.
11. Holmes, W., Bialik, M., & Fadel, C. *Artificial Intelligence in Education: Promises and Implications for Teaching and Learning.* Boston, The Center for Curriculum Redesign. 2019. P. 243.
12. Horovik A., Makhamajonov H. Actual problems of teaching programming in universities. *Online scientific forum for young scientists, doctoral students and researchers*. TATUFF-EPAI. 277. 2023. 275-277. (In Russian.)
13. Khenner E.K., Frieze C., Zane O. IT education as a factor to influence gender imbalances in computing: Comparing Russian and American experience. *The Education and science journal.* 22(8). 2020, 189-206. DOI: 10.17853/1994-5639-2020-8-189-206.
14. Lubkov A.V. Modern Problems of Pedagogical Education. *The Education and science journal.* 22(3). 2020. 36-54. DOI: 10.17853/1994-5639-2020-3-36-54. (In Russian.)
15. Nakov S., Kotlev V. *Fundamentals of Computer prog-ramming with C#.* Sofia, Bulgaria, Baso. 2013. P. 1122.
16. Oblomov I.A., Rzhavin V.V., Krasnov V.I., Fadeeva K.N. Teaching declarative programming languages. *Bulletin of the Chuvash State Pedagogical University named after I.Y. Yakovlev*. 1(106). 2020. 204-210. DOI: 10.37972/chgpu.2020.69.53.026. (In Russian.)
17. Autor. Teaching the Python programming language in higher educational

ISSN 3030-3907

institutions. *Informatics and Education.* 38(5). 2023. 78-86. DOI: 10.32517/0234-0453-2023-38-5-78-86. (In Russian.)

18. Patil P., Jadhav P., Kanase M. Imparting effective Teaching Learning Methods for teaching C programming course to First Year Non-IT Students. *Journal of Engineering Education Transformations.* 36 (Sp.Is.2). 2023. 133-140. DOI: 10.16920/jeet/2023/v36is2/23019.
19. Pirogov V.Ju., Popova E.I. Some aspects of teaching programming. World of Science. *Pedagogy and psychology.* 7(6). 2019. 1-11. (In Russian.)
20. Prokhorenok, N.A., Dronov V.A. *Python 3. The most necessary.* St. Petersburg: BHV-Petersburg. 2019. P. 608. (In Russian.)
21. Qarshiyev N.B. The content of the course of modern programming languages in the context of higher education and the technology of its teaching. *International scientific journal.* 1(7). 2020. 1607-1610. DOI: 10.5281/zenodo.7316541.
22. Rozov K. V., Posadnikov A. V. The Python programming language in a pedagogical university: from the basics to artificial intelligence. *Informatics and Education.* 6. 2019. 26-33. DOI: 10.32517/0234-0453-2019-34-6-26-33. (In Russian.)
23. Saad M. A. et al. The Reasons behind the Weakness of some Students in Programming Courses in the College of Applied Science, Ibri. *International Journal of Modern Education and Computer Science.* Vol.8, No.1. 2016. 48-54. DOI: 10.5815/ijmecs.2016.01.07
24. Shahamiri R. S. The Challenges of Teaching and Learning Software Programming to Novice Students. *Encyclopedia of Information Science and Technology*. 2018. 7 p. DOI: 10.4018/978-1-5225-2255-3.ch643.
25. Shidt G. *The Complete Reference: C# 4.0.* Moscow: Wilyams; 2011. (In Russian.) P. 1056.
26. Szlavi P., Zsako L. Methods of teaching programming. *Teaching Mathematics and Computer Science*. ½. 2003. 247-257. DOI: 10.5485/TMCS.2003.0023.
27. Tomiltcev A.V., Maltsev A.V. The problems of professional training assessment: methodological approaches. *The Education and science journal*. 20(4). 2018. 9-33. DOI: 10.17853/1994-5639-2018-4-9-33. (In Russian.)
28. Tretyakov O.A., Fedorkevich Y.V. Choosing the first language for learning programming. *The world of science. Pedagogy and psychology*. 5(8). 2020. 24-29. (In Russian.). https://mir-nauki.com/PDF/64PDMN520.pdf.
29. Troelsen A., Japikse P. *Pro C# 10 with .NET 6. Foundational Principles and Practices in Programming C#*. New York, USA, Apress. 2022. P. 1680. DOI: 10.1007/978-1-4842-7869-7.
30. Wan B., Liu Y. Research of Case Teaching Practice in C# Programming Language. *Proceedings of the 2019 International Conference on Advanced*

ISSN 3030-3907

Education Research and Modern Teaching (AERMT 2019). Available Online, October. 2019. DOI: 10.2991/aermt-19.2019.44.

31. Wirth, N. Computing science education: The road not taken. *ITiCSE '02: Proceedings of the 7th annual conference on Innovation and technology in computer science education*. Denmark. 2002. P. 1 – 3. Doi.org/10.1145/544414.544415.

32. Zagitova A. I. Problems of teaching programming in universities and advancement of programming courses forstudents. *Economy and society*. 6(37). 2017. 779-782. (In Russian.)

ISSN 3030-3907