OSCAR
PUBLISHING SERVICES

Research Article

THE SPECIFIC FEATURES OF TEACHING MODERN PROGRAMMING LANGUAGES IN PEDAGOGICAL HIGHER EDUCATION INSTITUTIONS

Rahmonov Mirzoxid Shavkatovich
Andijan State Pedagogical Institute, Uzbekistan

ABSTRACT

This article examines the relevance and necessity of teaching the Python programming language for the professional and methodological training of future computer science teachers in pedagogical universities. Python software is one of the modern programming languages and is being implemented at all levels of education. For this reason, it is necessary that a future computer science teacher has the knowledge and skills to use Python to learn the basics of programming.

KEYWORDS

Programming, programming language, Python, intellectual intelligence, professional training, learning problem, training course.

INTRODUCTION

Characteristics of the Python Programming Language

The history of the Python programming language began in the 1980s. The currently active version 3 was released to the public in late 2008 [1]. Due to its many conveniences, Python is taught as a foundational programming language at various levels of education. This includes higher pedagogical institutions, where Python is taught in the course "Modern Programming Languages." The purpose of teaching this subject is to broaden students' understanding of the theoretical and practical foundations of modern programming languages, helping them to apply these in solving various problems they may encounter in their professional activities, and to develop relevant skills and competencies.

– One of the features of the Python programming language is that, unlike other programming languages (Pascal, C, C++, Delphi, etc.), it uses dynamically typed data. The dynamic typing of data provides many conveniences when solving problems. Similarly, another feature of Python is its syntactic simplicity. Below, we list some of the advantages of Python:

– In most programming languages, a "semicolon" is placed at the end of each line of code to indicate the end of a command. However, this is not required in Python. That is, no symbol is placed at the end of the line. The "semicolon" is only used when two instructions need to be written on the same line. This feature helps reduce the number of syntax errors that are unrelated to the problem-solving process. We illustrate these points with the following examples (Figure 1).

Pascal	C++	Python
if a mod 2=0 then	if (a%2==0)	if a%2==0:
writeln("Juft")	count<< "Juft";	print("Even")
else	else	else
writeln ("Toq");	count << "Toq";	print("Odd")

Figure 1. The solution of the same problem in 3 different programming languages

– As can be seen from the above example, the way lines are terminated varies between different programming languages. In Pascal, a semicolon cannot be placed before else, whereas in C++, it is necessary to use a semicolon. In Python, there is no need to place a semicolon before else in such cases.

– Curly braces {} are not used to link operators within a block. In Python, such a block is represented by each instruction being indented, and the block is defined through indentation. This feature makes the code more concise and convenient. Additionally, the use of indentation in the structure helps make the code uniformly readable. It is well known from programming teaching practices that many students encounter difficulties aligning their code correctly and express negative opinions about this. Since it is easy and convenient to align code within a single structure in Python, it helps students develop good coding practices.

− Another feature relates to constructing logical expressions. In Python syntax, conditional expressions can be written clearly in the form a<b<c<...<n, whereas in other programming languages, logical expressions are simplified, divided into binary conditions, and the "and" operator is used to combine them. Figure 2 illustrates how a single condition is written in three different programming languages (Figure 2).

Pascal	C++	Python
if (a>0) and (a<10) then	if (a>0 && a<10)	if 0<a<10:

Figure 2. Writing a single condition in three different programming languages

The aforementioned and other specific features have made the Python programming language widely studied and taught in educational institutions. In the next decade, this language will increasingly be used in teaching, learning, and even for final certification exams in state educational institutions. Additionally, programming Olympiads are now also being held using Python. It is becoming increasingly important for future computer science teachers to master this language and understand its methodological foundations. The authors of computer science textbooks are now presenting educational materials based on the Python programming language.

Problems in Teaching Python Programming Language

Despite Python's many simplifications, there are some challenges in teaching this programming language. Some believe that Python is not suitable as the first programming language to learn [5,6].

Based on the experiences of teachers at Andijan State Pedagogical Institute and other pedagogical institutions where Python is taught, the following challenges can be highlighted:

One such challenge is directly related to dynamic typing. Many teachers acknowledge this issue. Students often forget that the input() function in Python takes string values. For instance, in Pascal, data input is performed using the universal readln operator, where the appropriate type of the variable is specified in the parameters. In Python, this can lead to a (TypeError) or a logical error. While the error in the first case seems simple, the interpreter's message regarding it is often quite complex. Even though the program runs, the output may not be correct. Here is an example of a program with a logical error due to dynamic typing:

```
gbook={input ('Фамилия: '):
    input ('Бахо: ') for i in range(10)}
gdes={}.fromkeys([1,2,3,4,5], 0)
for key in gbook:
    in gbook[key] in gdes.keys():
        gdes[gbook[key]]+=1
print(gdes)
```

This example is taken from the author's courses on Python programming: "For 10th-grade informatics students, a grading scale based on a five-point system is provided. It is necessary to calculate the number of each grade. If no grade is entered in the gradebook, the grade will not be accepted.

The issue arises because the grades in the gbook dictionary are given in string format, while the grades in the gdes dictionary are in numerical format. As a result, when trying to search for numerical grades in the gdes dictionary, the grade cannot be found due to the mismatch in data types.

Another problem arises when teaching the for loop. Unlike other programming languages, in Python, the for loop iterates over the elements of a sequence, and the value of the parameter does not change with each iteration. On one hand, the loop can refer to the indices of the elements in the sequence, or it can directly refer to the element itself. This makes the loop convenient and compact to use. However, on the other hand, students may not understand when to refer to the index and when to refer to the element.

The above-mentioned issues do not diminish the value of learning this language. We believe it is important for teachers to consider the following points when teaching programming languages.

Teaching the Programming Course in Pedagogical Higher Education Institutions

The relevance of teaching subjects in pedagogical higher education institutions stems from their traditions. To develop the syllabus and program for the "Modern Programming Languages" course, it should be linked to learning the Python programming language.

The "Modern Programming Languages" course is taught to bachelor's students in the 60110600 – Mathematics and Informatics program. Teaching future computer science teachers using modern teaching methods and digital technologies improves the effectiveness of education.

In modern education, the HEMIS platform has been introduced for the digital management of higher education institutions. This system allows students to engage in distance learning, providing access to educational materials at any time and in a convenient format. Below are the topics of the 2nd module of the "Modern Programming Languages" course:

1. Introduction to Python

2. Python programming language and syntax

3. Data types in Python

4. Python data sets and types

5. Conditional operators in Python

6. Loop operators in Python

7. Functions and modules in Python

8. Working with files and exceptions in Python

9. Object-oriented programming in Python

10. Graphics in Python

Based on our experience, when teaching programming languages, using mathematical examples to teach programming in other languages proves to be quite effective. School students spend much time solving equations in mathematics courses. Therefore, when learning branching algorithms, the focus shifts to

solving mathematical blocks. Here, Python's excellent feature of constructing conditions is demonstrated. When entering logical expressions like a<b<c<…<n, there is no need to divide them into simple conditions. In Python, when referring to the bool type, any non-empty and non-zero object is considered True, while any empty or zero object is considered False.

The next section focuses on teaching cyclic algorithms. There is no consensus on whether to teach the for or while loop first. For example, some educators believe it is better to start with the for loop, while others argue that it is necessary to begin with the while loop when teaching future informatics teachers. In Python, loop operators function similarly to functions, whereas in other programming languages like Pascal or C++, they operate differently. Some expert scholars believe that since the while loop works based on a condition, it should be taught first. They suggest teaching the range() function, which generates an arithmetic progression sequence, after that. Once students become familiar with the range() function, they can then move on to mastering the for loop. Unlike other programming languages, Python has break and continue operators within loops. Additionally, it's important to note that the else statement can also be used within loops.

In Python, strings are a basic type and form an iterable sequence. Therefore, the topic of "Working with Strings" is taught after cyclic algorithms and loop operators. When studying this topic, attention is paid to string functions and methods, as well as cyclic string processing. At this point, it is emphasized that string types in Python are immutable data types.

When teaching the topic of tuples and lists in Python, many experts recommend teaching both objects together: list (list) and tuple (tuple). This is because both data types are similar and represent ordered sequences. The main difference between them is that a list is a mutable sequence, while a tuple is immutable. The focus is primarily on teaching the methods used with lists. This is because some of the methods used for lists are also applicable to tuples. In Python, lists are used in a similar way to standard arrays in Pascal and C++. Since programming courses usually cover one-dimensional and two-dimensional arrays, this course considers using nested lists as an alternative to two-dimensional arrays.

Next, the dict (dictionary) data type is studied. A dictionary is essentially an associative array, meaning an array where data is stored in key-value pairs. In a dictionary, access to values is done not by element number, but by key (name). This section covers methods of creating dictionaries, as well as functions and methods for working with dictionaries. Particular attention is given to situations where using a list instead of a dictionary may be more beneficial and vice versa.

The Importance of Artificial Intelligence for Future Informatics Teachers

Today, artificial intelligence (AI) is rapidly developing in the field of information technology. In the scientific works of national and foreign pedagogical scholars [7-9], it is acknowledged that AI is becoming increasingly widespread in nearly all areas of society, including education. In pedagogical higher education institutions in our country, a separate module in specialized subjects is traditionally dedicated to introducing future informatics teachers (students) to the basics of AI. This is because a modern informatics teacher must be knowledgeable about digital

technologies, artificial intelligence, and the trends in their development.

As we all know, the Python programming language is widely used by professional programmers in the field of information technology. Knowledge of Python allows informatics teachers to easily learn the fundamentals of AI through specialized libraries (modules) that simplify the use of these modern technologies. With this knowledge, tasks such as computer vision, natural language processing, and machine learning can become accessible even to ordinary people without having to master programming languages. As digital technologies and AI become more deeply integrated into society, future informatics teachers can incorporate AI into the educational process, find programs in this field, use them in their pedagogical research projects, and inspire students to engage with promising technologies.

Survey of Students' Opinions on the Python Programming Language

We conducted a survey (in electronic form using iSpring Suite 8) among third- and fourth-year students who completed the "Modern Programming Languages" course at the pedagogical higher education institution. The survey was conducted in the form of a questionnaire. The questionnaire contained nine questions, to which the students could answer "yes" or "no," and they are as follows (Figure 3):
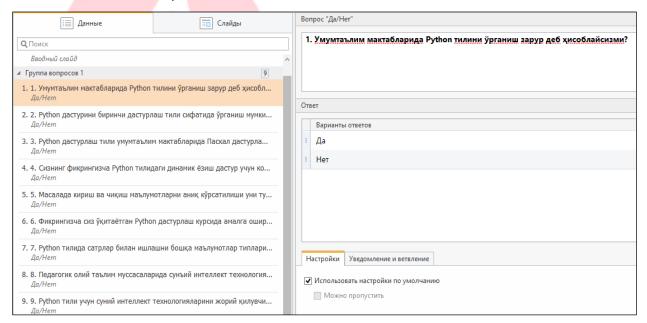

Figure 3. Survey Questionnaire Questions

Questions 2 and 3 provided an opportunity to express your opinion about the answer choices you selected.

The results of the questionnaire survey are shown in the diagram (Figure 4).
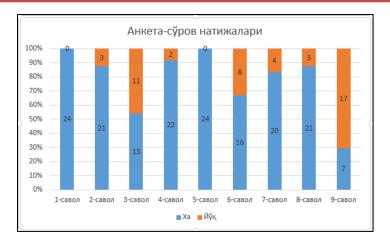
Figure 4. Representation of the Survey Results in the Diagram

Participants' Feedback

100% of participants acknowledged the relevance of teaching the Python programming language in general education schools. 87% of participants believe that Python can be learned as the first programming language. They note that Python is easier to learn and has a simpler syntax compared to Pascal, making it easier to understand. Those who chose "No" suggest that transitioning to other programming languages after learning Python can be difficult. From our observations, almost all participants answered the survey questions from the perspective of a programmer, rather than as a teacher, and paid less attention to the problems of teaching programming languages.

Looking at the diagram of the survey results, it can be noted that the responses to question 3 are almost evenly distributed among participants. That is, 54.17% believe that Python can replace Pascal when teaching the basics of programming in general education schools. This suggests that Pascal, which was previously taught, is becoming outdated, and teaching Python is becoming more relevant. However, 45.83% of

participants are skeptical about fully replacing Pascal with Python and suggest the importance of learning both languages. In our opinion, the necessity of learning two languages lies in expanding students' understanding of programming languages with varying levels of strictness.

According to 91.67% of students, dynamic typing simplifies writing code. In the course assignments, specifying input and output data precisely has proven effective, as 100% of participants indicated that this method of presenting the task makes it easier to understand. Moreover, 66.67% of participants agreed that the while loop should be taught before the for loop.

83.33% of participants confirmed that learning to work with strings earlier, before studying other data types, enhances teaching effectiveness.

87.5% of students who participated in the survey believe that studying artificial intelligence technologies in pedagogical higher education institutions is beneficial. This confirms the relevance of teaching the Python programming language in the "Modern

Programming Languages" course. Additionally, 29.17% of participants indicated that they are aware of the libraries (modules) in Python that support artificial intelligence.

CONCLUSION

Students recognize the relevance of teaching Python as the first programming language in general education schools, but they also face the challenge of "knowledge superiority over methodology." The question of whether learning Python is effective in school education and whether it should be recognized as the first programming language remains open for broad discussion among educators. Most students accepted the basis for constructing the programming course presented in our article. For pedagogical higher education institutions, the "Modern Programming Languages" course, which was developed to provide detailed coverage of the fundamentals of programming languages, holds great significance, especially with its emphasis on the integration of Python with modern digital technologies.

REFERENCES

1. Python 3.0 Release. https://www.python.org/download/releases/3.0/
2. Замонавий дастурлаш тиллари фанининг ўқув дастури. Низомий номидаги Тошкент давлат педагогика университети томонидан ишлаб чиқилган ва университет Кенгашининг 2022 йил "30" 08 1/3 даги қарори.
3. Информатика ва ахборот технологиялари. Ўрта таълим муассаларининг 11-синф ва ўрта махсус касб-хунар таълими муассаларининг ўқувчилари учун дарслик. "Extrumum-press". Тошкент. 2018, 133 б.
4. D.Ziyadullayev, M.Raxmonov, G.Temirova. Multimediyali elektron o'quv qo'llanma yaratishda Ispring suite 8 dasturni qo'llash. Toshkent davlat pedagogika universiteti. Ilmiy axborotlari. Ilmiy-nazariy jurnal. № 1 (14) 2018 y.
5. Иванов А. А., Рожкова Л. Искусственный интеллект как основа инновационных преобразований в технике, экономике, бизнесе // Известия СПбГЭУ. 2018. № 3. С. 112–115. https://unecon.ru/sites/default/files/izvestiya_3-2018.pdf
6. Поляков К. Ю. Язык Python глазами учителя //Информатика. 2014. № 9. С. 4–16.
7. Васильев Д. А. К вопросу изучения языка программирования Python в школе // Актуальные исследования в области математики, информатики, физики и методики их изучения в современном образовательном пространстве. Курск: КГУ, 2016. С. 80–83.
8. Самсонович О. О., Фокина Е. А. Искусственный интеллект — новые реалии // Международный журнал прикладных и фундаментальных исследований. 2018. № 5-1. С. 257–263. https://applied-research.ru/ru/article/view?id=12253
9. Уваров А. Ю. Технологии искусственного интеллекта в образовании // Информатика и образование. 2018.№ 4. С. 14–22.