



Methods for teaching programming languages Phyton, C ++ and their comparison

Kabira MIRZOEVA¹

Azerbaijan State Pedagogical University

ARTICLE INFO

Article history:

Received April 2022
Received in revised form
20 April 2022
Accepted 15 May 2022
Available online
15 June 2022

Keywords:

Phyton,
C++,
conditional operator,
programming languages,
lesson,
students,
teacher,
study,
knowledge,
decision,
learning.

ABSTRACT

If we look at the scientific, methodological, regulatory documents, literature for teaching students the basics of programming, we see that the state standard does not define any programming language, but only what knowledge and skills students should have.

Phyton is a great language in terms of student time at the Olympics and other competitions.

In Phyton programming, the max, min, and sort functions are used to find the maximum and minimum of an array, as well as to sort the array elements, which makes the program much more compact.

The program saves a lot of time as there is no need to compile in Phyton. An interpreter is used to convert a program into machine code. An interpreter can be used interactively to test functions during program development. For those who are new to Phyton, it is more appropriate to use the interactive mode, as it provides a visualization of the programming process.

A comparative analysis of the Python and C++ programming languages shows that Python has a simpler syntax. The concise writing of codes and the presence of an interactive mode once again prove that learning Python is more convenient.

In the presented work:

1. The advantages of Python are shown by comparing Python with C++.
2. A summary plan of an open lesson on the topic “Programming branching algorithms in the Phyton language” is presented and students are shown a methodology for studying the topic.
3. Evaluation of the course of lessons is shown by experts
4. An example of a program in C ++ and Phyton using the for period and if conditions for solving the following problem is considered:

¹ Candidate of physical and mathematical sciences, Senior Lecturer, Computer Department Science of Azerbaijan State Pedagogical University.

Find the number of integers in a one-dimensional array that are completely divisible by 3 and not divisible by 7.

2181-1415/© 2022 in Science LLC.

DOI: <https://doi.org/10.47689/2181-1415-vol3-iss3-pp126-133>

This is an open access article under the Attribution 4.0 International (CC BY 4.0) license (<https://creativecommons.org/licenses/by/4.0/deed.ru>)

Python, C++ dasturlash tillarini o'rgatish usullari va ularni taqqoslash

Kalit so'zlar:

Python,
C++,
shartli operator,
dasturlash tillari,
dars, talabalar,
o'qituvchi,
o'qish,
bilim,
qaror,
o'rganish.

ANNOTATSIYA

Talabalarga dasturlash asoslarini o'rgatish bo'yicha ilmiy, uslubiy, me'yoriy hujjatlar, adabiyotlarni ko'rib chiqsak, davlat standartida biron-bir dasturlash tili belgilanmaganligini, faqat o'quvchilar qanday bilim va ko'nikmalarga ega bo'lishi lozimligi qayd etilganligini ko'ramiz.

Python dasturlash tili olimpiada va boshqa musobaqalarda qo'llash mumkin bo'lgan ajoyib tildir. Python dasturlash tilida massivning maksimal va minimumini topish, shuningdek, massiv elementlarini saralash uchun max, min va sort funksiyalaridan foydalaniladi, bu esa dasturni ancha ixchamlashtiradi.

Dastur ko'p vaqtni tejaydi, chunki Pythonda kompilyatsiya qilishning hojati yo'q. Tarjimon dasturni mashina kodiga aylantirish uchun ishlatiladi. Interfaol dastur ishlab chiqish jarayonida funksiyalarni sinash uchun tarjimondan foydalanish mumkin. Python bilan yangi tanish bo'lganlar uchun interaktiv rejimdan foydalanish maqsadga muvofiqdir, chunki u dasturlash jarayonining vizualizatsiyasini ta'minlaydi.

Python va C++ dasturlash tillarining qiyosiy tahlili shuni ko'rsatadiki, Python soddaroq sintaksisga ega. Kodlarning qisqacha yozilishi va interaktiv rejimning mavjudligi yana bir bor Python tilini o'rganish qulayroq ekanligini isbotlaydi.

Taqdim etilgan ishda:

1. Python tilining afzalliklari Pythonni C++ bilan solishtirish orqali ko'rsatiladi.

2. "Python tilida tarmoqlanuvchi algoritmlarni dasturlash" mavzusidagi ochiq darsning umumlashtirilgan rejasi taqdim etilib, talabalarga mavzuni o'rganish metodikasi ko'rsatib o'tiladi.

3. Darslarning borishini baholash mutaxassislar tomonidan olib boriladi.

4. C++ va Python tillarida "for" davri va quyidagi masalani yechish shartlari ko'rib chiqilsa, foydalaniladigan dasturga misol tariqasida quyidagi masala keltiriladi:

Bir o'lchovli massivdagi 3 ga to'liq bo'linadigan va 7 ga bo'linmaydigan butun sonlar miqdorini toping.

Методика обучения языков программирования Python, C++ и их сравнение

Ключевые слова:

Python,
C++,
условный оператор,
язык программирования,
уроки,
учащиеся,
учитель,
изучение,
знания,
решение,
обучение.

АННОТАЦИЯ

Если посмотреть научно-методические, нормативно-правовые документы, литературы для обучения учащихся основам программирования, мы видим, что госстандарт не определяет какой-то один язык программирования, а только то, какие знания и умения должны быть у учащихся.

Python – отличный язык с точки зрения времени, отведенного студентам на Олимпийских играх и других соревнованиях.

В программировании на Python функции max, min и sort используются для нахождения максимума и минимума массива, а также для сортировки элементов массива, что делает программу намного более компактной.

Программа экономит много времени, так как нет необходимости компилировать в Python. Интерпретатор используется для преобразования программы в машинный код. Интерпретатор может использоваться в интерактивном режиме для проверки функций во время разработки программы. Для новичков в языке Python более целесообразно использование интерактивного режима, поскольку он обеспечивает визуализацию процесса программирования.

Сравнительный анализ языков программирования Python и C++ показывает, что у Python более простой синтаксис. Лаконичное написание кодов и наличие интерактивного режима лишней раз доказывают, что обучать Python удобнее.

В представленной работе:

1. Преимущества Python показаны при сравнении Python с C++.

2. Представляется сводный план открытого урока по теме «Программирование ветвящихся алгоритмов на языке Python» и демонстрируется учащимся методику изучения темы.

3. Оценка курса проведения уроков показывается экспертами.

4. Рассмотрен пример программы на C++ и Python с использованием операторов for период и условия для решения следующей задачи:

Найти количество целых чисел в одномерном массиве, которые полностью делятся на 3 и не делятся на 7.

Сегодня программирование стало очень важным инструментом в жизни человека. Самые популярные языки программирования – Java, C, C++, Python, C#. Их обучение, как правило, начинается в школе, учащиеся приобретают минимальный уровень знаний на каком-либо языке и готовы применить имеющиеся знания для решения задач.

В связи с актуальностью профессий, связанных с информатикой, робототехникой и программированием, язык программирования для обучения в школах следует выбирать таким образом, чтобы полученные знания соответствовали требованиям будущих работодателей и запросам общества. В этом смысле Python можно считать таким языком.

Python, в отличие от языков программирования семейства C, синтаксически понятен, прост в изучении и универсален в использовании.

Хотя Бейсик и Паскаль подходят для знакомства и навыков знания, несмотря на их популярность, их нельзя назвать современными. Эти языки заменяются продуктами с более высокой ценностью. Java и Ruby рассчитаны на опытных пользователей, умеющих прогнозировать и находить ошибки при написании программного кода и создавать программы, не охваченные школьным курсом.

Таким образом, школьникам лучше изучать Python. Дополнительное и углубленное изучение Python также поможет подготовить учащихся к изучению языка более высокого уровня, если ребенок заинтересован в обучении. Конечно, отношение к Python разное. Есть мнение, что люди, которые начинают изучать языки программирования с Python, не хотят переходить на низкоуровневые языки или не понимают их.

Выучить Python очень легко, на Python активно пишут различные приложения, игры, сайты и реализуют индивидуальные проекты в школах.

Из-за ограниченного времени, требуемого на конкурсах знаний, ученик должен тратить как можно меньше времени на написание программного кода, поэтому Python побеждает. Например, в Python, чтобы найти минимальный элемент в строке, нужно набрать тысячу слов, например, в Pascal нужно сравнить элементы между собой, введя дополнительную переменную.

Важно отметить, что при изучении языка программирования для школьников, на разных языках, при написании программы используются разные конструкции, которые были отмечены выше, но применяются одни и те же алгоритмы. Python экономит время ученикам для решения задач и позволяет им писать больше программ.

В целом можно обсудить вопрос о всех уроках, откуда они берутся и почему используется такая заметка. Это один из основных вопросов методики преподавания Python. Задача преподавателя – рассказать об этом перед тем, как давать какой-либо метод, и убедиться, что учащиеся понимают, как он работает.

Тогда не будет проблем с переходом на низкоуровневые языки как макс, мин, сортировка и т. д. Использование некоторых конструкций, например, наоборот, упрощает работу учащихся и избавляет их от возможных ошибок при написании. В школе работать на Python можно как на сайте repl.it, так и установив специальную программу. Для работы с Python достаточно иметь любую операционную систему Windows. Также будет полезно поработать с pythontutor.ru, где можно шаг за шагом следить за текстом программы.

Особенно важно показать это на первых уроках, чтобы учащиеся понимали, что происходит в программе. Сайт содержит как теоретическую информацию, так и возможность решить некоторые практические задачи для обучения.

Главный недостаток Python, как признают сами разработчики, заключается в том, что программы, написанные на нем, работают медленно, чем программы, написанные на компилируемых языках (C, C++). Но иногда скорость разработки

важнее скорости программы. И, как упоминалось выше, писать на Python быстрее, чем на других языках. Учеников нужно учить правильно писать программный код с первых уроков, потому что при обнаружении ошибок в большой и объемной задаче их исправление может занять много времени.

Так как, данное время язык программирования прост и доступен для использования, он широко распространяется. Повысится спрос людям, знающих язык Python. Мы надеемся изучение программирования о языке Python будет первым шагом.

Сравнительный анализ языков C++ и Python

В языке программирования Python нет необходимости объявлять переменную заранее. Это облегчает понимание типов данных, а также эффективно сокращает текст программы. В следующей таблице показан сравнительный пример разработки программы на C++ и Python с использованием операторов for период и условия если:

C++	Python
<pre>include<iostream> #include <math.h> using namespace std; int main() {int a[100],n,i,k; cin>>n; k=0; for (i=0;i<n;i++){ cin>>a[i]; if(a[i]%3==0 && a[i]%7!=0) {k++;}} cout<<"k="<<k<<endl; return 0;}</pre>	<pre>from array import * a=array('i',[-1,3,24,21]) s=0 for i in a: if (i%3==0) and (i%7!=0): s=s+1 print(s)</pre>

Как видите, язык Python очень прост с точки зрения синтаксиса. Однако C++ мощнее с точки зрения скорости выполнения. C++ подходит практически для всех платформ, а также для установленных систем, в то время как Python работает только на избранных платформах, поддерживающих языки высокого уровня. В Python нет скобок и точек с запятой. Там используются пробелы.

Программирование разветвляющихся алгоритмов на языке Python

Тема урока: Условный оператор

Класс: 8

Тип урока: комбинированный

Цель: Обучение условному оператору на языке Python

Задачи, поставленные на уроках:

Когнитивное: Формировать знания учащихся о языке Python, в условном операторе.

Инструктор:

Формировать знания учащихся о языке Python, в условном операторе.

Инструктор: Формировать знания учащихся о языке Python, в условном операторе.

Инструктор:

Научите условный оператор

Поучительный:

Помочь развить познавательный интерес к программированию, повысить культуру программирования.

Основные понятия: Операция, функция, выражения

Форма работы: Фронтальный и индивидуальный

Метод работы: Инновативный (устный), очный, иллюстративный, практический

Оборудование: Доска, компьютерная презентация, проектор, компьютеры, программное обеспечение: интерпретатор Python.

Структура урока:

Организационная часть, постановка урока, мотивация

Актуализация деятельности

Презентация нового материала

Завершение урока

Домашнее задание

Рефлексия

Ход урока

Организационная часть

Учитель здоровается с учениками и проверяет их готовность к уроку. Учащиеся здороваются учителем и демонстрируют свои готовности к уроку.

2. Представление нового материала в виде лекции.

Прежде всего, давайте вспомним, что такое ветвление. В линейных программах операции выполняются одна за другой. В ветвящихся алгоритмах управление передается соответствующему блоку в зависимости от того, истинно условие вернее или нет. Теперь давайте посмотрим на общее написание условного оператора в Python. Написание условного оператора в полном объеме будет следующим:

if <логическое выражение>:

<оператор 1>

else:

<оператор 2>

В неполной форме это будет так:

if <логическое выражение>:

<оператор 1>

В Python то, что в алгоритмах называется условием, называется логическим выражением, которое вычисляется первым. Если true <оператор 1> (после знака:), если false <оператор 2> выполняется (в полной форме), в неполной форме выполняется часть сразу после условия, если не иначе). Отметим три момента относительно условного оператора if-else;

1. Необходимо написать символ (:), который отделяет выражение (или последовательность выражений) для выполнения заголовкам оператора.

2. Строки после символа (:) должны быть с отступом. В skript файлах это расстояние по стандарту определено равным 4 пробелам.

Все строки в одном диапазоне после символа (:) выполняются при заданном условии.

Помимо полной и неполной формы условного оператора существует еще одна форма выделения. Если необходимо проверить более двух условий, это называется множественным ветвлением, и в этом случае используется оператор `elif`, состоящий из комбинации операторов `else` и `if`. Формат этого оператора следующий:

```
If < логическое выражение>:
<оператор1>
elif:
<оператор 2>
else:
<оператор 3>
```

3. Самостоятельная работа

Давайте рассмотрим пример с применением этого оператора.

Рассмотрим решение линейного уравнения $Ax = B$:

```
a=int(input('Введите a'))
b=int(input('Введите b'))
if (a==0) and (b==0):
print("beskonechnoe reshenie")
elif (a==0) and (b!=0):
print("net resheniy")
elif (a!=0):
x=b/a
print("x=",x)
```

Ученики выполняют работу самостоятельно.

4. Итоги урока

Чему вы научились на уроке? Есть ли у вас вопрос к учителю?

Учащиеся отвечают на вопросы учителя.

5. Учитель называет номер домашнего задания из учебника.

6. Рефлексия.

Спасибо вам всем. Я очень рада, что вы внимательно послушали меня и помогли мне. Теперь самое главное хорошо подготовиться дома к следующему уроку.

Учащиеся благодарят учителя.

Теперь приведем программу для решения линейного уравнения на C++:

```
#include <iostream>
#include<iostream>
#include <math.h>
using namespace std;
int main()
{int a,b;float x;
cin>>a>>b;
if(a==0 && b==0) {cout<<" бесконечное решение"<<endl;}
if(a==0 && b!=0) {cout<<" нет решения "<<endl;}
if (a!=0 && b==0) {cout<<"x=0"<<endl;}
if(a!=0 && b!=0) {cout<<"x="<<b/a<<endl;}
return 0;}
```

4. При оценке эффективности занятия было отмечено следующее:

1) Поставленные цели и задачи нашли отражение в обзоре и в ходе урока в соответствии с заданной темой.

2) Учебный материал удачно адаптирован к языку программирования Python и соответствует учебнику, по которому занимаются ученики. Ход урока построено правильно.

3) Обзор урока предполагает постоянное общение и взаимодействие между учителем и учениками.

4) На уроке используются мультимедийные презентации, что делает урок более интересным для учащихся и дает им более четкое понимание материала урока.

5) Обзор курса оформлен в соответствии с образовательными стандартами, что позволяет легко найти нужную информацию в обзоре.

БИБЛИОГРАФИЧЕСКИЕ ССЫЛКИ:

1. С. Шапошникова «Основы программирования на Python», Вводный курс, версия 2, Лаборатория юного линуксоида, 2011.

2. Доусон М. «Программируем на Python». СПб: Питер, 2014, – С. 416.

3. Бьерн Страуструп – Язык программирования C++ Специальное издание., 2011.

4. Алексей Васильев. Программирование на C++ в примерах и задачах., 2016.

5. Стефан Р. Дэвис – C++ Для чайников, 2014.

6. Литвиненко Н.А. – Технология программирования на C++, 2014.

7. Романов Е.Л. Си++. От дилетанта до профессионала 2014.

8. Чистый Python. Тонкости программирования для профи. – СПб.: Питер, 2018. – С. 288: ил. – (Серия «Библиотека программиста»).

9. Лейси Никола, Python, например. – СПб.: Питер, 2021. – С. 208.

10. Седер Наоми Python. Экспресс-курс. 3-е изд. – СПб.: Питер, 2019.

11. Билл Любанович «Простой Python, современный стиль программирования» Питер, 2019.

12. Пол Бэрри, Изучаем программирование на Python, 2017.

13. Васильев А.Н. Программирование на Python в примерах и задачах. Бомбора, 2022.

14. Коплиен Дж. Программирование на C++. – СПб: ПИТЕР, 2005.

15. Алексей Васильев – «Python на примерах. Практический курс», Наука и техника, 2019.