

OPEN ACCESS

Python to learn programming

To cite this article: A Bogdanchikov *et al* 2013 *J. Phys.: Conf. Ser.* **423** 012027

View the [article online](#) for updates and enhancements.

You may also like

- [ng: What next-generation languages can teach us about HENP frameworks in the manycore era](#)
Sébastien Binet
- [CodeToProtect©: C++ Programming Language Video Game for Teaching Higher Education Learners](#)
Mazeyanti Mohd Ariffin, Nurshazlyn Mohd Aszemi and Mohammad Syazran Mazlan
- [Implementation of python source code comparison results with Java using bubble sort method](#)
E Insanudin



244th Electrochemical Society Meeting

October 8 – 12, 2023 • Gothenburg, Sweden

50 symposia in electrochemistry & solid state science



Deadline Extended!
Last chance to submit!

New deadline:
April 21
submit your abstract!

Python to learn programming

A Bogdanchikov¹, M Zhaparov¹ and R Sulyev¹

¹ Suleyman Demirel University, 1/1 Abylaikhan St., Kaskelen, Almaty, Kazakhstan

E-mail: meirambek.zhaparov@gmail.com

Abstract. Today we have a lot of programming languages that can realize our needs, but the most important question is how to teach programming to beginner students. In this paper we suggest using Python for this purpose, because it is a programming language that has neatly organized syntax and powerful tools to solve any task. Moreover it is very close to simple math thinking. Python is chosen as a primary programming language for freshmen in most of leading universities. Writing code in python is easy. In this paper we give some examples of program codes written in Java, C++ and Python language, and we make a comparison between them. Firstly, this paper proposes advantages of Python language in relation to C++ and JAVA. Then it shows the results of a comparison of short program codes written in three different languages, followed by a discussion on how students understand programming. Finally experimental results of students' success in programming courses are shown.

1. Introduction

Python programming language is most suitable as a first language to learn for newbie programmers, because it has powerful tools that reflect the way people think and the way they implement the code. [1] Moreover it minimizes extra keywords that are necessary to write syntactically correct program. This approach seems to be more productive than teaching C++ or Java languages, which have a lot of terms and elements that are related to the specifics of a language rather than to an algorithm realization. In addition, instructors in over a dozen universities, such as MIT, UC Berkeley, UC Davis, Sonoma State University, the University of Washington, the University of Waterloo, Luther College, and Swarthmore College, have used it for teaching the introductory programming course to the students of computer science department. [2]

Today, for a computer scientist, it is important to learn at least one programming language, because all innovations and technologies are based on thorough understanding of computers, operating system, software API or some hardware peripherals. All of which are created by programmers that use specific way of thinking. And to gain that way of thinking, one has to get used to one of the programming languages and become qualified in software development. [6]

For any person, who starts to learn programming, it is important to concentrate on programming concepts rather than on language specifics, because they may be different for various programming languages. But Python provides the highest level of programming. So the student does not have to think about memory management, which is unavoidable in C++, or class hierarchy, that is unavoidable in Java, or variable types and declarations, that exist in almost each programming language.

2. Advantages

The easiest thing in Python is to write “Hello World” program. Most programming languages require writing a lot of specific methods, or functions, class or program declarations, etc. But Python gives ability to start programming without those requirements. You can check examples we gave in “Comparison” chapter.

Python is interpreted language, so with the use of command-line interpreter student can easily check how operators or functions work. [1] Python interpreter has built-in help module, which can significantly improve the process of understanding of different aspects of language.

In order to understand programming languages, freshmen (students which have no experience in programming) need to learn how to think like computer scientist, and that requires a great effort and a complete shift in their paradigm of thinking. [5] Implementation of Python code is easy enough, so that a person who passed a course of elementary mathematics will find such tools as ‘variables’ and ‘functions’ easy-to-use.

Whenever a programmer needs a prototype of software, Python, with its rich library, can be used. [3] Then the software may be rewritten in lower language if needed. Advantages of Python are significant, so using it as primary language to learn programming can highly affect the speed of learning computer science in general.

3. Comparison

Fangohr H in his paper compared C, MATLAB and Python [7], and here we are showing the differences between Python, JAVA and C++.

As first and very primitive example of simplicity of Python language, we will demonstrate the code of “hello world” program written in three different languages: Java, C++ and Python as shown in tables 1, 2 and 3 respectively.

Table 1. JAVA code of ‘hello world’ program

```

1. public class Hello{
2.     public static void main(String args[]){
3.         System.out.println("Hello, world!");
4.     }
5. }
```

Table 2. C++ code of ‘hello world’ program

```

1. #include <iostream>
2. using namespace std;
3. int main(){
4.     cout<<"Hello, world!"<<endl;
5. }
```

Table 3. Python code of ‘hello world’ program

```

1. print "Hello, world!"
```

So to explain the first program to a student, an instructor will need to explain many other useful, but unnecessary terms like *public static void main* or *using namespace std*, but in case of Python there is only single line of code and nothing else.

Another example is class declaration and inheritance. Tables 4, 5 and 6 display the code that is necessary to create small hierarchy of classes: *Person* and *Student* with constructors.

Table 4. JAVA code of Person hierarchy

```

1. public class Person{
2.     protected String name;
3.     protected String surname;
4.     public Person(String n, String s){
5.         this.name = n; this.surname = s;
6.     }
7. }
8. class Student extends Person{
9.     protected String group;
10.    public Student(String n, String s, String g){
11.        super(n,s);    this.group = g;
12.    }
13. }
```

Table 5. C++ code of Person hierarchy

```

1. #include <string>
2. using namespace std;
3. class Person
4. {
5.     public:
6.         Person(string &n, string &s);
7.     protected:
8.         string name, surname;
9. };
10. Person::Person(string &n, string &s)
11. {
12.     name = n; surname = s;
13. }
14. class Student: public Person
15. {
16.     public:
17.         Student(string &n, string &s, string &g);
18.     protected:
19.         string name, surname, group;
20. };
21. Student::Student(string &n, string &s, string &g):Person(n, s)
22. {
23.     group = g
24. }
```

Table 6. Python code of Person hierarchy

```

1. class Person:
2.     def __init__(self, name, surname):
3.         self.name = name; self.surname = surname
4. class Student(Person):
5.     def __init__(self, name, surname, group):
6.         Person.__init__(self, name, surname)
7.         self.group = group

```

4. Discussion

Python do not have security attributes (like: public/private/protected), so the program become simpler and shorted, more strict and understandable. Also Python is very dynamic so fields/attributes can be created on the fly, which cannot be done in JAVA or in C++. Polymorphism is nature of Python functions and class methods, unlike C++ with its virtual or non-virtual functions. Operator overloading gives extra power for Python objects, because it can be used for any natural expression, unlike JAVA's restricted syntax.

Programming indentation in Python plays a great role in program structure, because of that any program written in Python is easily read and understood. But in the case of JAVA or C++ every student is trying to make his/her program shorter, fitting it in one line, or trying to write every statement with same indentation, which makes the program difficult to be read and understood not only for the teacher but for the student himself.

Python includes wonderful algorithms in its native libraries. [3] So the student does not need to understand *long* arithmetic to make huge calculations. It already exists in native *long* data type in Python. There are good tools to sort, find, slice, and join any sequence of data. Sometimes they may not be so effective, like those which are written in C++ for a specific purpose. But for the beginners it is not the primary priority.

Python has its specific way to store and use variables, so the programmer no longer needs to define types of the variables, they will be defined by the values that variables store. Of course it is important to know how different types of variables are stored in memory, this can be discussed in lectures, but for the practice it is easier to skip the information about data types.

Also Python has adaptive and intuitive set of keywords and commands that impressively help students to learn programming.

5. Experiment

We compared results of midterm examinations of the course given in Java programming language (fall 2011) and the same course taught in Python programming language (fall 2012). Total midterm score of students, who took the course in Python language, increased by 16% in relation to previous year students, who took the same course in JAVA. Three groups of students (EN1-A-04, EN1-B-04 and EN1-C-04) were taken into consideration while comparison. Total of 76 students in three groups participated in the course "CS101 Algorithms & Programming (JAVA)" in fall 2011, and 65 students from first three groups are taken into account for course "CS101 Algorithms & Programming (Python)" in fall 2012. Results of their midterm marks are shown in Figure 1 below.

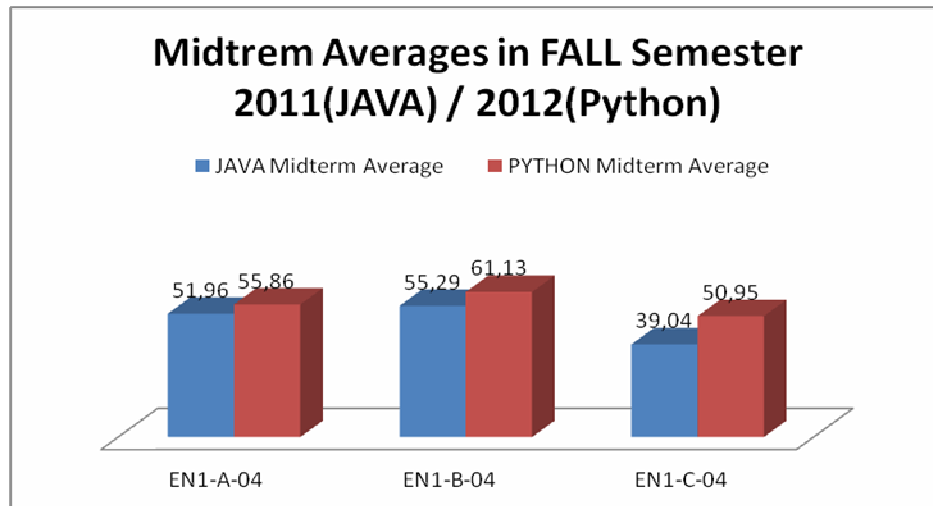


Figure 1. Midterm results for 2011(JAVA) / 2012 (Python)

6. Conclusion

In the conclusion, we would like to say that the initial understanding of programming is essential for a computer scientist. And because of complexities of advanced programming languages it appears that learning them is not so easy. First programming language imprints students' understanding and desire to learn programming and computer science in whole. So it is important to apply precise and adequate strategy to the process of learning the language.

Today some of leading universities either use Python to teach introduction to programming to their students or develop and teach their own simple compiler that can be easily understood by students. [2]

As a result we can say that students understand programming well with the use of Python. They have command line tool which allows them to test their theories immediately. Moreover there is a very good help tool in interpreter which always reminds about the structure of specific classes. Students enjoy simple *turtle* library [3] that provides very simple interface to draw on canvas. That helps them to understand simple statements like cycles and conditions. But the major result, of course, depends on students' desire to learn programming and computer science.

In experiment section we compared the results of midterm marks of the course taught using Java language with the results of the midterm marks of the same course taught using Python. We observed that results increased by 16%.

References

- [1] Downey A, Elkner J and Meyers C 2008 "*How to think like Computer Scientist` Learning with Python*"
- [2] <http://www.pythontutor.com/>
- [3] <http://docs.python.org/3/library/>
- [4] Blank D, Kay JS, Marshall JB, O'Hara K and Russo M "*Calico: A Multi-Programming-Language, Multi-Context Framework Designed for Computer Science Education*"
- [5] Anshul KJ, Manik S and Manu SG "*Algorithm Building and Learning Programming Languages Using a New Educational Paradigm*"
- [6] Sanders ID and Langford S "*Students' Perceptions of Python as a First Programming Language at Wits*"
- [7] Fangohr H "*A comparison of C, MATLAB, and python as teaching languages in engineering*"