Choosing the First Educational Programming Language

Vladyslav Kruglyk[1] and Michael Lvov[1]

[1] Kherson State University, 40 Rokiv Zhovtnya, 27
73000, Kherson, Ukraine
`kruglik@ksu.ks.ua, lvov@ksu.ks.ua`

Abstract. The article describes requirements to educational programming languages and considers the use of Python as the first programming language. The issues of introduction of this programming language into teaching and replacing Pascal by Python are examined. The advantages of such approach are regarded. The comparison of popular programming languages is represented from the point of view of their convenience of use for teaching algorithmization and programming.

Keywords: Educational programming language, Python, Pascal, first programming language.

Key Terms: Didactics, CompetenceFormationProcess, InformationCommunicationTechnology, TeachingMethodology

1 Introduction

The informatization of society lays new claims to teaching programming. Information technologies are strategic area of economy development in Ukraine, and the significance of their development and support is equal with the development of electronics in the last century.

At the end of the last century humanities have become very popular specialties. A school course of informatics was altered as well. As for now, a programming course has been almost totally excluded of secondary schools curriculum. The subject "Informatics" has been focused on teaching office technologies, and the proprietary software of one famous company was chosen as an example (such choice has been made historically). Therefore, the restoration of students' background in mathematics and informatics has become an urgent and important issue.

1.1 Overview of Current Situation

At schools conceptually different methodological approaches are used to teach programming languages. They are: equipped and non-equipped. The essence of the first one is in teaching programming without computers, by using so-called

educational algorithmic languages. From our point of view, such approach was reasoned by the only factor - a lack of computers at schools.

The second approach implies teaching the basics of algorithmization and programming at school with the use of industrial programming systems. Mostly BASIC, TURBO PASCAL and visual programming environments, like DELPHI and VISUAL BASIC on the basis of BASIC and OBJECT PASCAL are used as programming languages at schools.

The programming languages, named above, are developed for industrial purposes, and they are not intended for teaching, because they don't explicitly contain the means, based on the concepts of algorithmization and programming. Let's remember words of the academician A.P. Ershov, who asserted that educational programming language has to involve all main programming concepts [4].

Finally, attempts to implement and use educational programming languages at schools were not successful. From our point of view, a way out is to find an industrial programming language, which can be successfully used as an educational one.

2 Requirements to Educational Programming Languages

Let's note that the Pascal language was initially developed by N. Wirth as an educational language. This language demonstrates such conceptual peculiarities, like strict typing and availability of means of structured (procedural) programming. And, by N. Wirth, this language should facilitate forming of a good style of algorithmic thinking, and, thus, programming. In particular, the author tried to make its syntax intuitively clear even at the first acquaintance with the discipline "Programming".

During long time Pascal was fairly considered one of the best programming languages for education purposes. Unfortunately, the versions of programming environments, which were used to teach the language (Turbo Pascal, Borland Pascal), have become outdated. New programming systems, which are based on Pascal, for example, Delphi, are too expensive and industrially, not educationally, oriented. Particularly, the programming environment is poorly suitable to teach basic programming and algorithmization due to its complexity.

At the same time, modern means of interface are so well developed, that their existence should not be ignored even at the elementary stages of learning programming. From our point of view, the use of console for the input-output is nowadays insufficient.

Let's take a look at the requirements, which are necessary to be put forth to the educational programming language and to the programming system, connected with it. By A.P. Ershov, the following requirements to the educational programming languages should be marked out [4]:

− The language and the programming system, connected with it, should represent all main programming concepts in their structure.
− The language should have clear logic and should be methodically conditioned, i.e. its structure should be built by certain methodical scheme, which allows sequential introducing of new concepts and forming of necessary skills.

− The structure of educational language should be methodically conjugated with the structure of modern standard high-level programming languages, i.e. learning this language should make further transition to learning other languages (for instance, within professional training) easier or not complicated.

Let's add some more clarity to the description of educational programming language.

− Programming language should correspond to the next paradigms of programming: imperative and structural (at basic levels), and object-oriented programming (later stages).

− Educational programming language should be learnt in the way to avoid "learning in advance", i.e. using the references to the materials, which haven't been studied before.

− The syntax of programming language should be as simple and clear as possible, to make a program not only easily written, but also easily read and understood.

− The language should have sufficiently high-level and a special emphasis on algorithmization. Moreover, mechanisms of its interaction with a computer should be hidden as much as possible. Therefore, the use of extra-high level features, such as a concept of abstract data type and its explicit definition in the text of the program (for example, like class), is advisable.

− The problem of the use of explicit memory management needs individual consideration. It is very important for studying, especially when learning dynamic data structures.

3   Requirements to the Programming Systems

Let's consider requirements to the programming systems for educational purposes. From our point of view, the main requirements are the following:

3.1. Requirements to the environment:

− Cross-platform – it is advisable not to bind a student to a certain platform, because a user should choose the platform to work on his/her own.

− Open Source –programming system should be an open one, with open source codes.

− Free distribution –programming system should be free of charge.

3.2. Requirements to the subsystem of editing:

− Initial files should be stored in the file system;

− There should be a system of projects, which jointly stores related files and provides convenient access to the files within the project;

− Modern editing tools of initial texts, like autoformat, intellisense, pages numbering, etc. should be available;

− Modern debugging tools, like step-by-step running, variables watch, call stack should be available.

3.3. Requirements to the libraries of educational programming language.

− Educational programming language should contain the libraries, which provide the solutions for main tasks, like file system, sockets, Internet, localization, user interface, etc.

4   Industry Value of a Programming Language

Under the concept Industry Value we mean the ability of the language to be used and demanded in IT sphere. Let's consider such programming languages, like Pascal, C#, Java, Python, C++, ObjectPascal, PHP, Javascript from the point of view of their applicability in different areas.

In the capacity of evaluation criteria let's consider the possibilities of writing the applications in the next areas: system programming, Web-programming, Desktop–programming, Mobile–programming, Web –client development. See Table 1

Table 2. Applicability of programming languages.

	Pascal	C#	Java	Python	C++	Object Pascal	PHP	Javascript
System programming	0	0	0	0	2	0	0	0
Web – programming	0	2	2	2	1	1	2	1
Desktop – programming	1	2	2	2	2	2	1	0
Mobile – programming	0	2	2	2	2	0	0	0
Web – client development	0	2	2	2	0	0	0	2

Legend:
0 – the language is impossible to use for such purposes
1 –the language can be used for these purposes but there are better solutions
2 – the language can be used for these purposes and is recommended to

There is a need to make a conclusion that certain programming languages are aimed to solve bigger amount of tasks. Therefore, it's better to choose them for educational purposes.

4.1   Rating of Programming Languages

As programming languages constantly evolve, there are some systems of their evaluation.

Dutch company TIOBE Software BV [9] is a famous author of the popularity rating of programming languages, which is calculated on a regular basis. While making a popularity rating, Tiobe takes into account the amount of language experts, the amount of language learning courses as well as the amount of vendors, who support the language and the amount of the code, indexed by search engines. Let's take a look at the popularity rating of programming languages for February, 2012.

Table 2. TIOBE rating.

	Place in a rating
Pascal	**15**
C#	**3**
Java	**1**
Python	**8**
C++	**4**
Object Pascal	**-**
PHP	**6**
Javascript	**10**

Ohloh.net is a free, public directory of Free and Open Source Software and the contributors who create and maintain it. Ohloh represents a resembling rating, based on the activity of commits (saving) of program code.

The lines show the count of monthly commits made by source code developers. Commits including multiple languages are counted once for each language [10].

It is clear that the high rating of a programming language should be taken into consideration when choosing a programming language for educational purposes. Due to the ratings, represented at the table 2 and figure 1, it's possible to make a conclusion that any listed programming language is more popular and needed, than Pascal.

5   Choosing a Programming Language

Due to all the information represented above, let's try to choose the best programming language to replace Pascal.

As far as the only drawback of Pascal is reducing of its use in real projects, and, as a result, falling of its popularity, choosing a popular and demanded language makes sense. All programming languages, taken into the consideration, meet this condition.

The second important criterion is a level of suitableness of each programming language for teaching, including beginner level.

Let's examine the chosen languages one by one.

Pascal – is a good language to teach programming, but the use of this language in real applications is on the wane, and newer languages are replacing it. Moreover, existent programming environments are either outdated or proprietary, or experimental. The second considerable reason to refuse this language is a poor motivation of students to learn it due to the practical impossibility to use it in real projects.

C#, Java – are similar languages with resembling concepts and tasks. They both are very important in the IT field. Unfortunately, they are difficult for beginners because require the knowledge of object-oriented programming almost immediately. The same

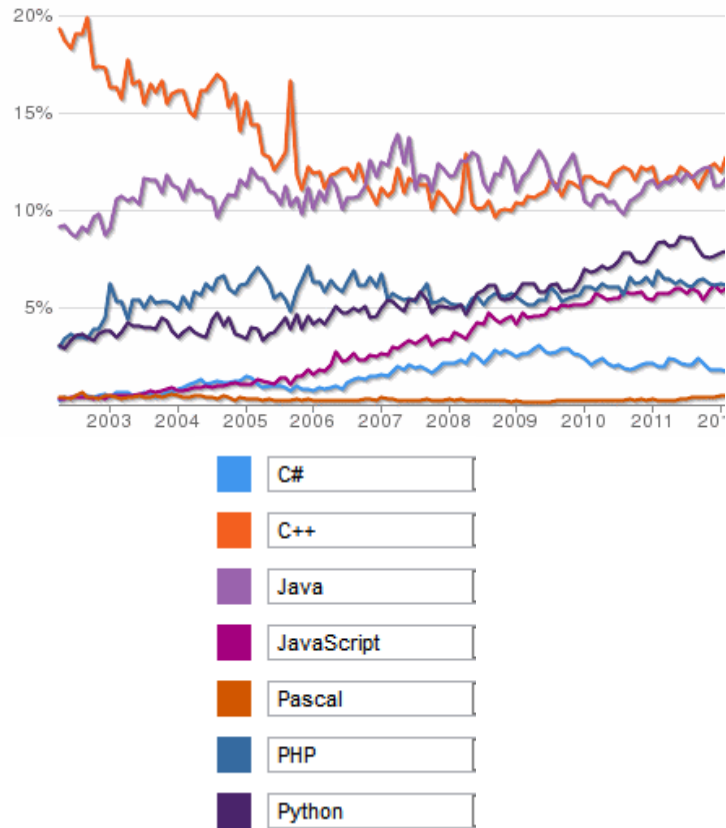drawback is characteristic of C, Java, C# languages, which is why they are not suitable to teach beginners.

Fig. 2. Dynamics of the activity of the use of programming languages.

Example of the Java code (Hello world):
```
public class helloWorld
{
  public static void main(String [] args)
  {
  System.out.println("Hello World!");
  }
}
```
C++ - is now one of the basic and the most popular languages. It is very convenient to teach basic skills of system programming to programmers. It should be used to teach specialists, but not beginners [8].

ObjectPascal – is Pascal which supports the paradigm of object-oriented programming. It is used as a basic language of Delphi programming system. Because of the price of the programming system and the necessity to operate with the concepts of object-oriented programming, it is not suitable to teach programming to beginners.

PHP – is a programming language used to build server scenarios for web sites. Despite its popularity, PHP language is not very elegant and features syntax contradictions, non-coordination in function names and a dynamic typing of variables with a possibility to change a type. PHP possesses C-like syntax, and is practically used only in web programming. Therefore, PHP is not suitable to teach programming to beginners.

Javascript – is an object-oriented scripting language. It is a dialect of ECMAScript language. JavaScript is usually used as embedding language for program access to application objects. It is most widely used as a scenario language in browsers, to add interactivity to web pages. Its main architectural features are: dynamic typing, weak typing, automatic memory management, prototype programming, has first-class functions. This language is not suitable to teach programming to beginners because of the peculiarities of its use [12].

Python – is a high-level programming language for joint purposes, with an emphasis on a developer's productivity and readability of the code. Main architectural features are: dynamic typing, complete introspection, exceptions processing mechanism, multithreaded computations support and easy-to-use high-level data structures. It will be regarded as a pretender to replace Pascal language.

6   Python as a Programming Language for Beginners

Python is a modern scripting language, which supports main programming paradigms. [3, 5].

The philosophy of the language «The Zen of Python» requires elegant style of writing the programs from programmers:

'Beautiful is better than ugly.
Explicit is better than implicit.
Simple is better than complex.
Complex is better than complicated.
Flat is better than nested.
Sparse is better than dense.
Readability counts.' [11]

As a quality of introduction of the language into teaching process depends on teachers, one of the main tasks during the introduction will be teachers' retraining to teach the cycle of disciplines with the use of Python. Notably, there should be no problems with mastering the language for the beginner level of programming, because the syntax of Python is very simple. In addition, Python's syntax is not C-like, which makes understanding of programs and algorithms easier.

Let's compare main algorithmic constructions of Pascal and Python.

Example of the code in Pascal (Hello world):

```
Program test;
Begin
  Writeln('Hello world');
End.
```

Example of the code in Python (Hello world):

```
print "Hello world"
```
Example of the code in Pascal (branching):
```
Program test;
Var
  x,y,z:integer;
Begin
  X:=5;
  Y:=2;
  Z:=3;
  If x<y and y<z then
    Writeln('Good');
  Else
  Begin
    Z:=x+y;
    Writeln(z);
  End;
End.
```
Example of the code in Python (branching):
```
x,y,z = 5,2,3
# Another way
#x=5
#y=2
#z=3
if x < y < z:
  print "Good";
else:
  z = x+y
  print z
```
Example of the code in Pascal (cycles):
```
Program test;
Var
  x:integer;
Begin
  x:=5;
  While x<10 do
  begin
    Writeln(x);
    x:=x+1
  end
End.
```
Example of the code in Python (cycles):
```
x=5
while x < 10:
  print x
  x = x + 1
```

Example of the code in Pascal (functions):

```
Program QuickSort;
Const
  n = 5;
Var
  A : array[1..n] of integer;
Procedure Swap(i, j : Integer);
Var
  b : Data;
Begin
  b := a[i];
  a[i] := a[j];
  a[j] := b
End;

Procedure Hoare(L, R : Integer);
Var
  left, right : Integer;
  x : Data;
Begin
 If L < R
   then begin
     x := A[(L + R) div 2];
     left := L;
     right := R ;
     Repeat
       While A[left] < x do
         left := left + 1;
       While A[right] > x do
         right:=right - 1;
       If left <= right
         then begin
           Swap(left, right);
           left := left + 1;
           right := right - 1;
          end
     until left > right;
     Hoare(L, right);
     Hoare(left, R)
   end
End;

Begin
  A[1]:=2;
  A[2]:=7;
  A[3]:=1;
  A[4]:=5;
  A[5]:=2;
```

```
    Hoare(1, n);
End.
```

Example of the code in Python (functions):

```python
def hoare(l, r, arr):
  if l<r:
    x = arr[int((l+r)/2)]
    left = l
    right = r
    while not(left>right):
      while arr[left] < x:
        left = left + 1
      while arr[right] > x:
        right = right - 1
      if left <= right:
        arr[left],arr[right] = arr[right],arr[left]
        left = left + 1
        right = right - 1
    hoare(l, right, arr)
    hoare(left, r, arr)
  return arr

vector = [2, 7, 1, 5, 2]

vector = hoare(0,len(vector)-1,vector)

print vector
```

Among the advantages of Python there is a need to note that the interpreter is a cross-platform one; there is a big amount of third-party and standard libraries, which enable solving of almost any task; the language is supported by the development environment Eclipse.

Python is ported and works properly on all famous platforms. It is possible to program for the .NET platform on Python.

Among drawbacks of the Python language for educational purposes, it is possible to name dynamic typing and impossibility to use references. If the possibility to use references refers to the part of system programming and was excluded intentionally, dynamic typing is undesirable for educational programming language. This is the main shortcoming of Python in the context of using it for teaching.

7 Conclusion

The use of Pascal language to teach programming to beginners at schools and in universities has become old-fashioned. The discussion around adequate replacement of this language has lasted for quite a long time. Now Python language, which is powerful and simple, is suggested to be used for these purposes.

Python supports lots of programming paradigms: structural, object-oriented, functional, imperative and aspect-oriented, and learning can be started without any

preparation. There is one more advantage of the language: all algorithms are written easily and structurally in Python. Therefore, due to all mentioned above, it is possible to affirm that Python pretends to become a decent replacement for educational programming language PASCAL both at schools and on the first courses of higher education establishments. Programmers with strong knowledge of this language are always in demand, and they're well paid. Hence, the students, who mastered Python, can aspire to start positions in IT companies.

References

1. Zelle, J. M.: Python as a First Language, http://mcsp.wartburg.edu/zelle/python/python-first.html
2. Dr. Donaldson, T.: Python as a First Programming Language for Everyone, http://www.cs.ubc.ca/wccce/Program03/papers/Toby.html
3. http://en.wikipedia.org/wiki/Python_(programming_language)
4. http://ershov.iis.nsk.su/archive/eaimage.asp?did=41919&fileid=224284
5. http://python.org/
6. Spivakovsky, O.V., Lvov, M.S.: Ways of Improvement of the Course "Basic Algorithmization and Programming" at the teachers' training higher education establishments. In: Computer at School and in Family. vol.4, pp.22-24 (2001) (in Ukrainian)
7. Spivakovsky O.V.: The Concept of Information Science Teaching at School and in Teachers' Training Higher Education Establishment . In: Computer at School and in Family. vol3., pp. 21-25 (2003) (in Ukrainian)
8. Stolyarov A.V.: Essay"C Language and Teaching Beginning Programming", http://www.stolyarov.info/pvt/anti_c . (in Russian)
9. http://www.tiobe.com
10. http://www.ohloh.net
11. http://www.python.org/dev/peps/pep-0020/
12. http://en.wikipedia.org/wiki/JavaScript
13. Phillips D.: Python 3 Object Oriented Programming. ISBN-13: 978-1-849511-26-1 Packt Publishing, 404 pages (July 2010)
14. Alchin M.: Pro Python. ISBN-13: 978-1-4302-2757-1 Apress, 368 pages (June 2010)