

AN ELECTRONIC PROGRAMMING COURSE NOT JUST FOR BEGINNERS

Peter Szabó - Oľga Arvaiová

Python is a very fast growing, versatile, object-oriented programming language with open source. The purpose of this work is to create an electronic course of language Python for beginner. An electronic course was implemented in electronic learning system Moodle Faculty of Aeronautics. In addition to basic knowledge, this work describes specific programming tasks which are solved by the Faculty of Aeronautics in Kosice. It can be concluded that Python can be an alternative way to solve these problems. The work also includes a database of solved programs with source code.

Key words: Python, programming language, electronic course, open source

1 INTRODUCTION

The programming language is a tool that is used to achieve a specific objective. Therefore, it is important his right choice according to the usage level of knowledge and at the same time. On learning the basics of programming in most cases using structured languages, because they are easier for beginners. The problem lies in the fact that most of today's applications are developed in object-oriented languages such as. Java. There makes it necessary to find a programming language that meets both of these attributes.

Python programming language was developed as a well structured and object-oriented language. Thus combines both features that we consider important in finding a suitable language for teaching programming technically oriented schools.

Python is a very fast growing, multipurpose, object-oriented programming language with open source. Currently, Python uses about 14% of programmers. Python has a very simple syntax rules which are easy to learn.

The programming language Python is free to use at www.python.org, without limitation it can use, modify and distribute.

On the Internet we can find high-quality literature and documentation along with sample programs that can serve also beginners. Few, however, are in Slovak language, especially not with sections of source code versions of Python 3. Despite the fact that nowadays is one of the popular programming languages the Java language, in many areas just use Python. Using it are dealt with: networked applications, web development, numerical and scientific applications, prototyping, rapid application development, and so on. These and many other advantages makes the Python programming language suitable for the University with technical direction.

2 FROM STRUCTURED PROGRAMMING TO OBJECT- ORIENTED PROGRAMMING

Structured Programming roused a strong response to the late 70th and 80 years. Programming language Pascal get the structured programming to the forefront. He was developed exactly for this kind of programming.[1]

For this style of programming is characteristic by dividing the source code into a logically ordered blocks. This blocks are considered from statements, loops and logical blocks. Structured programming processed the source code in the order, in which section of code has been written. In developing the algorithm programmer progresses from top to bottom, using basic control structures, sequences, branching and looping.[1][2]

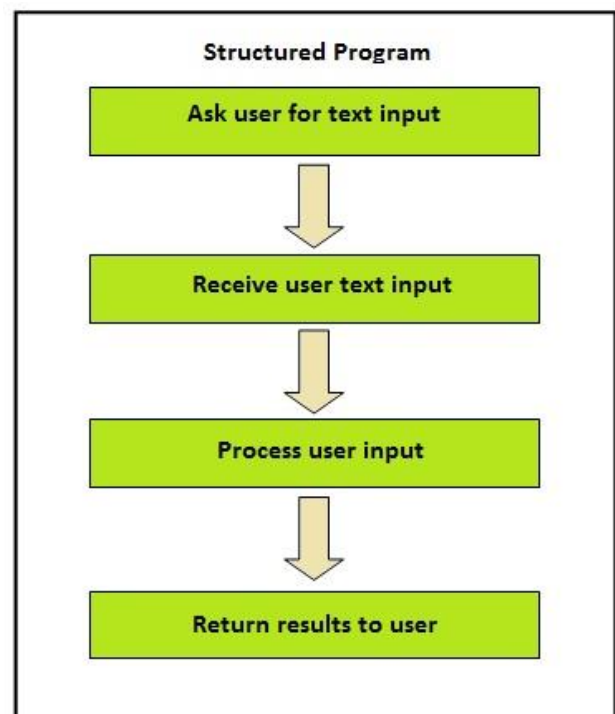


FIGURE 1: STRUCTURED PROGRAM

The rise of object-oriented programming gets the structured programming into the background. Object-oriented programming has its roots in procedural programming. The first object-oriented language was Simula I and Simula 67. Basics of the language Simula 67 is based on a procedural language Algol 60. The program is consists of objects that contain methods and data.[3][4]Chyba! Nenašiel sa žiaden zdroj odkazov.

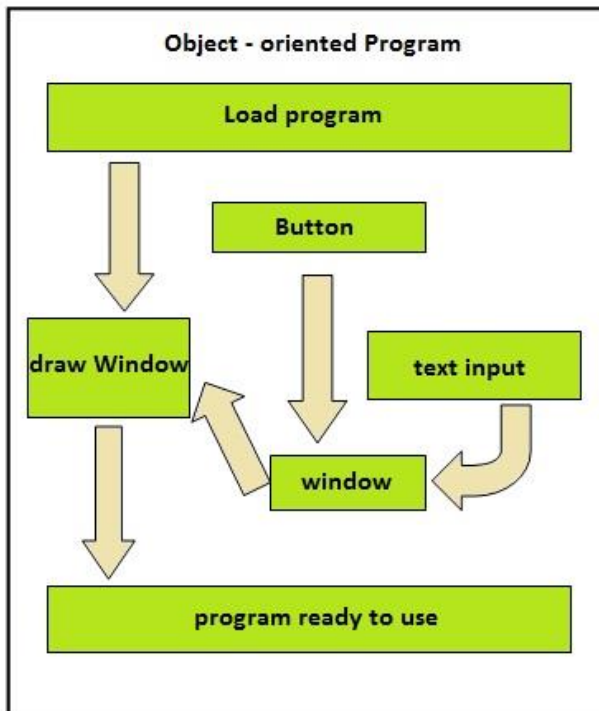


FIGURE 2: OBJECT- ORIENTED PROGRAM.

3 COURSE OF LANGUAGE PYTHON

Python was, and still is being developed as open source software. It is available free of charge. Some people still don't trust the free software. They are concerned a lack of support or loss of customers.

Python is a high- level programming language. We also know other high- level languages. For example C, C++, Perl and Java. There are also low-level languages, sometimes referred to as „machine languages“ or „assembly languages.“[6]

Two kinds of programs process high-level languages into low- level languages. Interpreter and compiler.

An interpreter reads a high-level program and executes it, meaning that it does what the program says. It processes the program a little at a time, alternately reading lines and performing computations. A compiler reads the

program and translates it completely before the program starts running. In this context, the high -level program is called the source code, and the translated program is called the object code or the executable. Once a program is compiled, you can execute it repeatedly without further translation. [6]

3.1 Preview of the first part of the course of programming – The first program

Usually, the first program you write in a new language is called “Hello, World!”. All that the program does is display the words „Hello, World!. In Python, it looks like this[6]:

```
>>>print „Hello, World!“
```

This print statement doesn't actually print anything on paper, but it display a value on the screen. The result is the words[6]:

```
Hello, World!
```

The quotation marks in the program mark the beginning and end of the text. They don't appear in the result. In Python 3, the syntax for printing is a little different[6]:

```
>>> print („Hello, World!“)
```

The parentheses indicate that print is a function. For the rest of this course, we'll use Python 3.[6]

4 PROGRAMMING IN AVIATION

Nowadays, computer technology is obviously modernized and aircraft instrumentation. Most of the variables on the plane scans using electrical sensors, pressure gauges, thermometers, tachometers. Their output signal is fed to a digital format, numbers, and is subsequently processed.

It is the point when the programmers come whose job is to programme calculations of measured data on actual flight parameters such as speed, altitude, course of flight, engine speed, fuel quantity, temperature-gas engines, brake temperature and etc.

In the cockpit of 50-60 years we find the amount of mechanical and electromechanical devices.



FIGURE 3: CABIN OF AIRBUS 380[7]

On-board imaging displays the work of programmers graphically displays information that was previously on the individual units. The most basic is about artificial horizon (Attitude director indicator (ADI) display-screen vertical situation indicator), which provides the pilot information with the longitudinal and lateral inclination angle of the aircraft.



FIGURE 4: BOEING 737 NG – FLIGHT INSTRUMENT[8]

In recent years expanded greatly unmanned aerial vehicles (UAV).

At the Faculty of Aeronautics in Kosice, unmanned vehicles have been engaged for three years with the Department of in aviation technical training (KLTP). During that time, the Katedra obtained them three kvadrapters and two tricopters. The latest addition is hexacopter. In the production are six small kvadrapters measuring up to 30 cm. There is camera with FPV (first person view) glasses with telemetric transmission image. From among different types of UAVs are interested department focuses mainly on mini UAVs with electric drive.



FIGURE 5: TRICOPTER



FIGURE 6: QUADROPTER



OBRÁZOK 7: HEXACOPTER

For data visualization is possible to use different kinds of software. KLTP using Processing or Qt.

Processing is an open source programming language and environment especially suitable for people who want to create different images, animations and interactions. It was originally developed to serve in teaching the basics of programming and visualization.[9]

Qt is a leading multi-platform application and development environment for built and mobile operating

systems. Qg use standard C++ and is used mainly for the development of software applications with a graphical user interface, and also to develop applications with database access and so on. [10]

For Qt exists extension called PySide, Python for Qt. The project PySide provides Lesser General Public License (LGPL) Python for Qt, which is actually something of a loose version license for free software. Pyside Qt enables the development of free of charge, free programs, as well as commercial programs.[11]

Depending on the needs, Pyside is available either as source code or as a binary package for different platforms. Binaries are available for various platforms such as:[11]

- Linux
- Microsoft Windows
- Mac OS X
- Maemo
- MeeGo
- Nightly Builds

The necessary visualization outputs should be able to use the Python language, which offers a variety of options for creating graphical output, processing and databases is also necessary to highlight its compatibility with other programming languages.

Using Python would therefore be resolved as programming devices such as air. altimeter, speedometer, GPS with chart showing our position and velocity vector, information about power being drawn from the battery and rotor speed, flight time, and estimated time to discharge the batteries, as well as an image from the camera.

In Python, you can also create applications that would be helpful for a variety of calculations carried out in the design of structures, analysis and recording flight parameters.

5 PRACTICAL PART- ELECTRONIC COURSE IN MOODLE

A part of the work is also the electronic course implemented in system Moodle.

The course includes 13 fields of study. Each field of study contains a lecture, exercises and tests on this topic. The end of the course provides a database of solved tasks.

Task:

Calculate the Mach number M , at the speed of $V=1000\text{kmh}^{-1}$ and the local air temperature $T=18^{\circ}\text{C}$.

Solution:

The program was written in order to calculate the Mach number at different speeds and different temperatures. The graphics has been added using the Python Tkinter module.

Source code:

```
def main():
    pass

if __name__ == '__main__':
    main()

from tkinter import *
import math
import tkinter.messagebox

class App(Frame):
    def __init__(self, master=None,):
        Frame.__init__(self, master)
        self.pack()

        self.entrythingy = Entry()
        self.entrythingy.pack()

        self.entrythingy2=Entry()
        self.entrythingy2.pack()

        self.vysledok=Label(master,text="Vysledok je:")
        self.vysledok.grid(row=5, column=5)
        self.vysledok.pack()

        self.obsah = StringVar()
        self.obsah.set("vlozte teplotu")
        self.entrythingy["textvariable"]
        = self.obsah

        self.obsah2=StringVar()
        self.obsah2.set("vlozte rychlost")
        self.entrythingy2["textvariable"] =
        self.obsah2

        self.tlacidlo=Button(master,text="vypocitaj",
            command=self.machove2)
        self.tlacidlo.grid(row=3, column=3)
        self.tlacidlo.bind("<Return>",
            self.machove2)
        self.tlacidlo.pack()

    def print_obsah(self, event):
        print("hi. contents of entry is now
        ---->",
            self.obsah.get())
```

```

def machove2 (self):
    try:
        rychlost=int(self.obsah2.get())
        teplota=int(self.obsah.get())
        rychlost= rychlost/3.6

result=rychlost/App.machove(teplota)

self.vysledok.configure(text="Vysledok
je:"+str("%.3f" % round(result,3))
        return result

except ValueError:

tkinter.messagebox.showinfo(title="upozornenie",
message="Ocakava sa realne cislo")

def machove (teplota):
    teplota=teplota+273
    result=20*(math.sqrt(teplota))
    print( "%.3f" % round(result,3))
    return result

root=Tk()
app=App(root)
root.mainloop()

```

6 CONCLUSION

Future graduates of technical schools had already often asked questions about their application after graduation. On the labor market, we find only a few companies that offer job "fresh" graduates without experience.

Most companies in the area of information-technology requires at least a minimum knowledge in this field, particularly in the area of programming. For non-technical, managerial positions, the situation is also complicated. Manager must be knowledgeable in matters pertaining to his position and aware of the problems and their solution at the level of its scope. Proper distribution of tasks requires its own technical knowledge in the field.

Our concern became the language Python, mainly by combining structured and object-oriented programming. Course of programming in Python offers an easy opportunity to expand their knowledge in the area, which currently has great potential for the labor market.

To help beginners become familiar with the terms used by programmers and programming master problem as such.

Key features of the Python language are:

- simple syntax

- intuitive object-oriented programming
- dynamic data types
- extensive libraries and modules for almost every task
- extensions for C, C++, Java, Qt and the like.

In the text, we have also demonstrated the possibility of using the programming language Python directly at the Technical University in Kosice. We believe that this course brings the potential that can be further developed.

BIBLIOGRAPHY

- [1] Dušan Polanský. *Strukturované programování* [online]. 2001 [cit. 2013-01-03]. Dostupné z: <http://www.dusanpolansky.cz/clanky/banker.html>
- [2] Find Science. *Structured vs. Object-Oriented Programming: A Comparison* [online]. 2010 [cit. 2013-01-04]. Dostupné z: <http://www.brighthub.com/internet/web-development/articles/82024.aspx>
- [3] Fakulta informatiky Masarykovy univerzity. *História objektovo orientovaných jazykov* [online]. 2003 [cit. 2013-01-04]. Dostupné z: http://www.fi.muni.cz/usr/jkucera/pv109/2003/xha_lasa.htm
- [4] Hammer Principle. *Pascal vs Python* [online]. 2013 [cit. 2013-02-21]. Dostupné z: <http://hammerprinciple.com/therighttool/items/pascal/python>
- [5] EECS User Home Pages. *A Brief History of Object-Oriented Programming* [online]. 2012 [cit. 2013-02-16]. Dostupné z: <http://web.eecs.utk.edu/~huangj/CS302S04/notes/o-o-intro.html>
- [6] *Think Python: How to Think Like a Computer Scientist* [online]. 2012 [cit. 2013-02-01]. Version 2.0.5. Dostupné z: <http://www.greenteapress.com/thinkpython/thinkpython.pdf>
- [7] Wikipedia. *Electronic flight instrument system* [online]. 2013 [cit. 2013-04-01]. Dostupné z: http://en.wikipedia.org/wiki/Electronic_flight_instrument_system
- [8] Smartcockpit. *Boeing 737 NG Flight instruments* [online]. 2013 [cit. 2013-04-01]. Dostupné z: <http://smartcockpit.com/>
- [9] Processing.org. *Processing* [online]. 2013 [cit. 2013-04-03]. Dostupné z: <http://www.processing.org/>
- [10] Qt. *About Qt* [online]. 2013 [cit. 2013-04-03]. Dostupné z: <http://qt.digia.com/About-us/>

- [11] Qt Project. *PySide* [online]. 2011 [cit. 2013-04-04]. Dostupné z: <http://qt-project.org/wiki/Category:LanguageBindings::PySide>

AUTHORS' ADDRESSES

Oľga Arvaiová, Ing., Faculty of Aeronautics - Technical University of Košice, Department of Air Traffic Management, olga.arvaiova@tuke.sk

Peter Szabó, RNDr., PhD., Faculty of Aeronautics - Technical University of Košice, Department of Aerodynamics and Simulations, peter.szabo@tuke.sk