Syllabus
Learn to Program: The Fundamentals
(3 ECTS)

Author:

Jennifer Campbell, Associate Professor, Teaching Stream, Department of Computer Science

Paul Gries, Associate Professor, Teaching Stream, Department of Computer Science

Psychology Programme
Meeting Minute #2.6-06.7/4 dated 28 June 2019

1. **Course Description**
a. **Title of a Course**: Learn to Program: The Fundamentals
b. **Pre-requisites:**
There are no prerequisites.
c. **Course Type** (compulsory, elective, optional): elective
d. **Abstract**:
Behind every mouse click and touch-screen tap, there is a computer program that makes things happen. This course introduces the fundamental building blocks of programming and teaches you how to write fun and useful programs using the Python language.

2. Learning Objectives:
This course introduces the fundamental building blocks of programming and teaches you how to write fun and useful programs using the Python language.

3. Learning Outcomes:
After completing the course, the student acquires the following skills:
- Python Syntax And Semantics
- Computer Programming
- Python Programming
- Idle (Python)

4. Course Plan
Week 1
Python, Variables, and Functions.
This module gives an overview of the course, the editor we will use to write programs, and an introduction to fundamental concepts in Python including variables, mathematical expressions, and functions.
Week 2
Strings and Designing Functions.
This module introduces strings (a Python data type used to represent text), and a process to follow when creating a function.
Week 3
Booleans, Import, Namespaces, and if Statements.
This module introduces Booleans (logical values True and False), how to convert between types, how to use Boolean expressions in if statements to selectively run code, and the concept of a Python module.
Week 4
For Loops and Fancy String Manipulation.
This module introduces one way to repeat code (using a for loop), how to manipulate strings, and how to use a debugger to watch a program execute step by step.

Week 5
While Loops, Lists, and Mutability.
This module introduces another way to repeat code (using a while loop), how to properly document your code to help other programmers understand it, Python's list data type, and the concept of mutation.
Week 6
For Loops Over Indices, Parallel and Nested Lists and Strings, and Files.
This module introduces how to use a for loop over the indexes of a list, how to nest lists, and how to read a write files.
Week 7
Tuples and Dictionaries.
This module introduces tuples (an immutable version of lists), and Python's dictionary type.

5. Reading List:

a. Required:

Campbell J., Gries P., MontojoJ., Wilson G., Practical Programming (2nd edition): An Introduction to Computer Science Using Python. – URL: https://doc.lagout.org/programmation/Pragmatic%20Programmer /ISBN: 978-1-93778-545-1

b. Optional:

Лутц, М. Изучаем Python / М. Лутц; Пер. с англ. А. Киселева. – 4-е изд. – М.-СПб: Символ-Плюс, 2014. – 1272 с. - ISBN 978-5-932861-59-2.

Прохоренок, Н. Python. Самое необходимое: видеокурс / Н. Прохоренок. – СПб.: БХВ - Санкт-Петербург, 2011. – 408 с. - ISBN 978-5-9775061-4-4.

6. Grading System:

Final grade will be given on a scale of 1 to 10 throughout the class. All grades, having a fractional part greater than 0.5, are rounded up. smaller than 0.5, are rounded down.
Final result = 0,5 * Python, Variables, and Functions + 0,5 * Strings and Designing Functions + 0,1 * Assignment 1: Time Zones + 0,5 * Booleans, Import, Namespaces, and if Statements + 0,5 * For Loops and Fancy String Manipulation + 0,15 * Assignment 2: DNA Processing + 0,5 * While Loops, Lists, and Mutability + 0,5 * For Loops Over Indices, Parallel Lists and Strings, and Files + 0,15 * Assignment 3 + 0,5 * Tuples and Dictionaries + 0,25 * Final Exam.
All test scores are set as percentages (from 1 to 100), for successful completion you need to score 80% or higher. The final grade for the course is exhibited on the platform in percentages, that should be transformed in a ten-point scale in accordance with the scale of compliance
96 - 100% - 10 points
85 - 95% - 9 points
80% -84% - 8 points
0 - 79% - 0 points

7. Examination Type:
Sample Questions for assessing the quality of knowledge:
1. Select the expression(s) that evaluate to an int value.
2. Write an expression that evaluates to the string ’L8R’ using only the variables start, middle, end, one call on function str, and string concatenation.
3. Select the code fragment(s) that correctly complete this function.
4. Select the code fragment(s) below that run without error.

8. Methods of Instruction:
Blended course: On-line lectures (https://www.coursera.org/learn/learn-to-program/home/welcome) and out-of-class work.

9. Special Equipment and Software Support (if required): PC, internet access
You'll need a computer that can run Python 3. Download and install Python 3.4.3: https://www.python.org/downloads/release/python-343/.

- Linux users: use your package manager to install Python 3.4.3.
- Windows users: choose the "Windows x86 MSI Installer" from the downloads page.
- Mac users: choose the "Mac OS X 64-bit/32-bit installer" from the downloads page. Also install ActiveTCL 8.6.4.