

The Hebrew University of Jerusalem

Syllabus

INTRODUCTION TO PROGRAMMING IN PYTHON - 76608

Last update 01-09-2016

HU Credits: 3

Degree/Cycle: 1st degree (Bachelor)

Responsible Department: programming instruction unit

Academic year: 0

Semester: 1st and/or 2nd Semester

Teaching Languages: Hebrew

Campus: E. Safra

Course/Module Coordinator: Hilla Moshieff

Coordinator Email: hila.mo@mail.huji.ac.il

Coordinator Office Hours: Mondays at 12:00 in the computer farm

Teaching Staff:

Ms. Hila Moshieff

Course/Module description:

This is designed to be a first course in computer programming in which the student will acquire knowledge, thinking methods and skills for the solution of programming problems, using the Python programming language.

Course/Module aims:

The goal of the course is to teach fundamental concepts, knowledge and understanding of Programming in general and programming skills using the Python programming language.

Learning outcomes - On successful completion of this module, students should be able to:

On successful completion of this course, the student should be able to (a) analyze a problem through the identification of its components, and (b) express its programming solution by using the tools that Python provides us.

Attendance requirements(%):

0

Teaching arrangement and method of instruction: Lectures will be conducted in the computer farms, examples will be displayed and students will be able to experiment with writing small programs.

Every week, a summary of the lesson's material, will be available in the Module site of the course.

During the practice session, the teacher will work individually with the students as needed.

An exercise will be given every week. At least 8 of the given exercises must be submitted in order to be eligible for a course grade.

Course/Module Content:

Basic Concepts

- what is a program?: a program as a virtual machine.*
- basic data types: integers, reals, and strings.*
- the concept of variable: memory place that is given a name and into which a value may be stored.*
- basic input: the functions input() and raw_input()*
- basic output: the print statement.*

Procedural Abstraction – the Function

- a function as a "black box": function application.
- system and programmer functions, modules.
- a function as a "white box": function implementation.
- parameter passing.
- returning the computed value with the return statement.
- local variables, global variables, and constants.

Flow Control Tools

- conditional branch – the if statement.
- loop/iteration – the while and for statements, and the continue and break statements.

Data Containers

- Sequences: strings, lists, tuples, sets, dictionaries.
- Sequences as iterators.

Text files processing

- creating, opening and closing a file.
- direct and sequential access: seek, write, and read.
- files as iterators.

Iterative Patterns

- the map, filter and reduce patterns.

High-order functions

- functions as values.
- functions as parameters.
- the built-in high-order functions: map, filter, and reduce.
- writing our own high-order functions.

The random module

- writing random-dependent programs.
- the functions seed(), getstate(), setstate(), random(), randint(), randrange().

Advanced topics (depending on students' interests and available time)

Required Reading:

There isn't required reading.

Additional Reading Material:

Grading Scheme:

Additional information:

The final grade will be determined by the student's test score at the end of the

course. To take the test, the students must submit, on time, at least 8 of the exercises that will be given during the course