Course Syllabus

PHOT 110 INTRODUCTION TO PROGRAMMING

2025 Spring

Instructor

Dr. Michaël Barbier

e-mail: michaelbarbier@iyte.edu.tr

Office: F-building, door on the right of Z5

Office hours: Friday 11:00 – 13:00 (or via appointment)

Teaching Assistants

Dr. Hazan Özkan

e-mail: hazanozkan@iyte.edu.tr

Office: Z9B

Office hours: TBA

Ozan Baran Orhan

e-mail: metintan@iyte.edu.tr

Office: Z9B

Office hours: TBA

Course Schedule

Tuesday	10:45 – 12:30	Ali Küçük lab (computer lab first floor F-building)
Tuesday	13:30 – 15:15	Ali Küçük lab (computer lab first floor F-building)

Course Fundamentals

Course Description

This course gives an introduction to programming with specific emphasis on scientific computing. Python will be used as the programming language within the course. Covered topics are: basic computer knowledge, elementary programming and algorithm implementations, vectors and arrays, control flow: loops and conditional, functions, input/output, plotting of 2D/3D graphs, object oriented programming, debugging and writing unit-tests.

Course Objectives (Learning Outcomes)

At the end of the course, you should be able to:

1. Understand the general structure of a computer program.
2. Create basic programs in Python from scratch.
3. Apply numerical techniques within Python using libraries such as Numpy and Scipy.
4. Plot graphs using Matplotlib.

5. Successfully inspect and debug your scripts.
6. Have a basic understanding of objective programming.
7. Assess the scripts' performance (finding and removing bottlenecks) and
8. Perform unit tests.

Prerequisites

There are no official prerequisites for this course, and no previous experience with programming is assumed. However, basic knowledge of calculus and algebra is assumed.

Textbooks

Course Textbook:

H.P. Langtangen, **A primer on scientific programming with Python**, Springer

Supplementary Materials:

Textbook materials: http://hplgit.github.io/scipro-primer/slides/index.html

Course website and announcements

You will find all announcements relevant to the course (homework, grades, etc.) on MS-Teams. Syllabus and course schedule may be subject to change during the semester.

Class structure

Course material will mainly be presented on the screen, and students are encouraged to type along when the functionality of short scripts is explained during the theory lectures. During the practical sessions, short scripts will be implemented. All lecture notes and class materials can be reached from Teams.

Course schedule (tentative)

Week 1	Computers and Python basics: interpreter, IDE, running a script (Ch. 1)
Week 2	Expressions, variables, variable types (Ch. 1)
Week 3	Program control flow: while-loop, for-loop, lists, and ranges (Ch. 2)
Week 4	Program control flow (ctu'd): conditional execution. Further data types: nested lists, Tuples, Arrays, (Ch. 3)

Week 5	Functions, variable scope (Ch. 3)
Week 6	Input/output and error handling (Ch. 4)
Week 7	Modules and script documentation (Ch. 4)
Week 8	Midterm exam
Week 9	Vectors, Arrays, implementation of numerical algorithms (Ch. 5), and random numbers (Ch. 8)
Week 10	Plotting graphs using Matplotlib (Ch. 5)
Week 11	Dictionaries, Strings, tables with Pandas (Ch. 6)
Week 12	Object Oriented Programming (Ch. 7)
Week 13	Object Oriented Programming ctu'd (Ch. 9)
Week 14	Additional topics: Code performance, benchmarking, profiling, unit tests and integrated testing (App. F)
Week 15	Final exam
Week 16	Final exam

Course Policies

Attendance and class behavior

Students who attend the lecture are expected to actively participate (in listening, typing along, understanding, problem solving sessions, etc.).

Homework assignments

An individual project will be assigned during the term. This project can be performed with help from others (fellow students, course instructors, etc.), but the problem solution should have a unique report. Homework solutions/reports can be uploaded to Teams on the indicated due date until 23.59. Late homework may be turned in, but 25 points will be deducted from the full score (100 points) for each day that it is late.

Examinations and Grading

There will be one midterm exam, one project, and a final exam. The exams are closed book with open questions and problems, which should be solved by creating scripts on the computer.

If you miss one of the exams without a valid excuse, a zero will be averaged into your grade. If you have a valid excuse (i.e., an official certified medical report), a make-up exam will be organized during the semester. Your final grade is the weighted average of your project, midterm and final exam grades, according to following weights (tentative):

- Midterm exam: 10%
- Project: 20%
- Final exam: 70%

Examination Dates

Midterm exam: Tuesday, April 8, at 10:45, duration: 2 hours

Project: Problems will be sent out on April 8 (after the midterm exam), the deadline of the project is on Thursday, May 8

Final exam: TBA, duration: 3 hours

Disabilities

Students with certified disabilities requiring special accommodations are urged to contact the instructor at the beginning of the semester so that suitable arrangements may be made.

Academic Integrity

Students who violate University rules on academic dishonesty are subject to disciplinary penalties, including the possibility of failure in the course and/or suspension from the University. Forms of academic dishonesty include copying the reports of homework assignments (from a fellow student or internet), cheating on exams, use of unauthorized materials for exams, and changing solutions to returned assignments and exams.