

PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
COLLEGE OF ENGINEERING
DEPARTMENT OF COMPUTER SCIENCE
ABET COURSE SYLLABI

IIC1103 INTRODUCTION TO COMPUTER PROGRAMMING

Credits and contact hours:	10 credits / 10 hours (4 h. Lectures; 3 h. Labs; 3 h. Independent learning experiences)
Instructor's name:	Mauricio Arriagada Benítez
Course coordinator's name	Mauricio Arriagada Benítez
Textbook:	<ul style="list-style-type: none">- The quick python book, V. Ceder, 2010.- Think Python: How to think like a computer scientist, A. B. Downey, 2013.- Python programming: An introduction to computer science, J. M. Zelle, 2010.- Python software foundation, Python v3 Documentation, http://docs.python.org/3/.
Course Catalog Description:	Develop students needed to be able to solve various problems using computer programming skills. It is expected that students learn to develop models and algorithms for solving these problems using the object-oriented approach and write programs that implement these algorithms using the Python programming language
Prerequisite Courses:	None
Co-requisite Courses:	None
Status in the Curriculum:	Required
Course Learning Outcomes:	<ol style="list-style-type: none">1. Explain basic concepts of a computer program such as algorithms, variables, expressions, flow control, functions, lists, strings, classes and objects.2. Apply basic techniques for solving various problems using the computer, how to identify relevant data problem and the relationships between them, modeling this data in a computer and representation to decompose large problems into several smaller problems.3. Apply reasoning to generate algorithmic solution to a problem as a sequence of well-defined steps, including conditional steps, repeat steps, function calls, and recursion.

4. Perform software development process, writing and debugging object oriented programs that meet certain specifications using Python programming language.
5. Use a software environment to write, compile and get better programs.

**Relation of Course to ABET
Criteria:**

- a. Knowledge of mathematics, science and engineering
- b. Design and conduct experiments: analyze and interpret data
- k. Techniques, skills, and modern tools for engineering practice.

Topics covered:

1. How to resolve problems using a programming language.
2. Data type: integers, floats, strings, booleans. Arithmetics and logic operators.
3. Input and Output.
4. Control flow: if statements and loops.
5. Operations over strings, list, tuples and dictionaries.
6. Programming object oriented: classes (constructor, methods), use of objects
7. Ordering and sorting over one and two dimensional list and tuples.
8. Files, handling exceptions.
9. Recursion
10. Simulation by iteration and event.