COMP100
Introduction to Computer Science and Programming

Course Description

COMP100 Introduction to Computer Science and Programming is intended for Computer Engineering students in their first year with little or no programming experience. It aims to provide students with an understanding of the role computation can play in solving problems and also to help students gain proficiency with shell tools.
The class will use the Python 3 programming language.

Course Overview

The lectures will be allocated to introducing concepts and programming and one lecture every other week will be allocated to introduce tools. The labs will alternate between solving Python questions related to the Problem Sets and the Practicals on tools.

A significant portion of the material for this course will be presented only in the lecture, so students are expected to regularly attend lectures. Labs give students a chance to ask questions about the lecture material or the problem set for the given week. Sometimes, new material may be covered in the labs. Regular lab attendance is also expected.

Problem Sets and Quizzes

Each problem set will involve programming in Python. There will be 5 problem sets in the course, each lasting for two weeks. *You must complete the problem sets to pass the course.* In the second week, there will be a practical week where there will be exercises to complete about the tools learned in the last lecture of the week.

Participation is not mandatory but there will be random in-class quizzes.

Catalogue Description

Overview of Computers, Programming, Algorithms, and Programming Languages. Programming with Python: Data types, variables, operators. Control statements: conditionals, loops, iteration. String manipulation. Functions, recursion, decomposition and abstraction. Tuples, lists, dictionaries. Aliasing, mutability, cloning. Files. Object oriented programming, classes, inheritance. Testing, debugging, exception handling, program efficiency.

The shell. Shell Tools and Scripting. Editors (Vim). Data Wrangling. Command-line Environment. Version Control (Git).

Textbook

The textbook is Guttag, John. *Introduction to Computation and Programming Using Python: With Application to Understanding Data Second Edition*. MIT Press, 2016. ISBN: 9780262529624. The book and the course lectures parallel each other, though there is more detail in the book about some topics. It is available both in hard copy and as an e-book.

Course Learning Outcomes (CLOs)

1) Implement a given algorithm as a computer program in Python.
2) Explain what a given program in Python does and learn and practice with testing and debugging techniques to identify errors in a program.
3) Understand and follow good programming practices in coding and testing.
4) Help students master the command-line, use a powerful text editor, understand basics of SSH and version control systems.
5) Learn the concepts of Abstraction and Decomposition via functions and variable scopes in Python.
6) Learn control flow via branching and different types of loops.
7) Learn and use different data types in Python such as list, tuples, dictionaries.
8) Learn recursive thinking and adapt it to different problems.
9) Learn basic principles of Object-Oriented Programming and Inheritance in Python.
10) Learn basics of program efficiency via examples of search and sort algorithms.

Assessment Methods

Method	Weight
Problem Sets + Practicals	30 + 10
Quiz (in-class)	10
Midterm	20
Final	30

All percentages are subject to change within a margin of 5% during the semester at the instructor's discretion. Total number of quizzes, hence points per quiz will depend on the progress of the class as a whole, although you should expect about 10 quizzes. A passing grade based on the other factors may be converted to a failing grade under the following conditions:

- If you do not turn in all the problem sets, where turning in means including a serious attempt to complete each problem.
- If you copy another person's work and represent it as your own in any single problem set or exam.
- If your midterm and final average is below 30.

Course Schedule

Weeks	Topics	Assignments
1	What is computation? GitHub Classroom	Problem Set: 0
2	Branching and Iteration The Shell	Practical on The Shell
3	String Manipulation, Guess and Check, Approximations, Bisection	Problem Set: 1
4	Decomposition, Abstractions, Functions Editors	Practical on Editors
5	Variable Scopes	Problem Set: 2
6	Tuples, Lists, Aliasing, Mutability, Cloning Shell Tools and Scripting	Practical on Shell Tools and Scripting
7	Recursion, Dictionaries	Problem Set: 3
8	Testing, Debugging, Exceptions, Assertions	Midterm Study Questions
Midterm		
9	Object Oriented Programming Data Wrangling	Practical on Data Wrangling
10	Python Classes and Inheritance	Problem Set: 4
11	Understanding Program Efficiency - I Command-line Environment	Practical on Command-line Environment
12	Understanding Program Efficiency - II	Problem Set: 5
13	Searching and Sorting Version Control (Git)	Practical on Version Control (Git)
14	Files and Modules	Final Study Questions
Final		

Academic Dishonesty

Honesty and trust are important to all of us as individuals. Students and faculty adhere to the following principles of academic honesty at Koç University:

- Individual accountability for all individual work, written or oral. Copying from others or providing answers or information, written or oral, to others is cheating.
- Providing proper acknowledgement of original author. Copying from another student's paper or from another text without written acknowledgement is plagiarism.
- Study or project group activity is effective and authorized teamwork. Unauthorized help from another person or having someone else write one's paper or assignment is collusion. Cheating, plagiarism, and collusion are serious offenses resulting in an F grade and disciplinary action.
- Academic Regulations-Academic Integrity

Academic dishonesty in the form of cheating, plagiarism, or collusion are serious offenses and are not tolerated at Koç University. University Academic Regulations and the Regulations for Student Disciplinary Matters clearly define the policy and the disciplinary action to be taken in case of academic dishonesty. Failure in academic integrity may lead to suspension and expulsion from the University. Cheating includes, but is not limited to, copying from a classmate or providing answers or information, either written or oral, to others. Plagiarism is borrowing or using someone else's writing or ideas without giving written acknowledgment to the author. This includes copying from a fellow student's paper or from a text (whether printed or electronic) without properly citing the source. Collusion is getting unauthorized help from another person or having someone else write a paper or assignment.