

School of Engineering and Natural Sciences / Computer Engineering (English)

2022 - 2023 Academic Year

INTRODUCTION to PROGRAMMING

Syllabus

Course Description					
Name	Code	Semester	T+A Hour	Credit	ECTS
INTRODUCTION to PROGRAMMING	COE1113180	Fall Semester	3+2	4	6
Prerequisites Courses					
Recommended Elective Courses					
Language of Instruction	English				
Course Level	First Cycle (Bachelor's Degree)				
Course Type	Required				
Course Coordinator	Prof.Dr. Selim AKYOKUŞ				
Name of Lecturer(s)	Prof.Dr. Selim AKYOKUŞ				
Assistant(s)					
Aim	This course introduces fundamentals of programming, problem solving and algorithm development for students with little or no prior programming experience using Python programming language. The objective of this course is to prepare students for more advanced programming courses as well as providing an understanding of computation in problem solving and engineering as a self-contained course for those students who want to write programs for their studies and professional work. The course emphasizes structured programming, algorithmic and object thinking in a problem-driven way after teaching fundamental concepts and structures. Topics include an introduction to computers, programming languages and Python; elementary programming, selections, data types, strings, iteration, functions, GUIs (graphical user interfaces), objects and classes, inheritance and polymorphism, lists (arrays) and multidimensional lists, sets and dictionaries, files, exceptions and recursion. Weekly laboratories and assignments with different problems, practice and coding exercises will improve student's capabilities and fluency in programming.				
Course Content	This course contains; Introduction to Computers, Programming, and Python,Elementary Programming,Mathematical Functions, Strings, and Objects,Selection statements,Loops,Loops,Functions,Lists, Multidimensional Lists, Tuples, Sets, and Dictionaries,Basic GUI Programming,Advanced GUI Programming ,Inheritance and Polymorphism,Files and Exception Handling,Recursion,Developing Efficient Algorithms.				
Course Learning Outcomes			Teaching Methods	Assessment Methods	
Understand programming concepts and techniques using Python Language.			1, 15, 2	A, C	
Use control statements, loops, functions, and lists.			1, 13, 14, 15	A, C, D	
Understand the differences between procedural and object-oriented paradigms.			1, 13, 14, 15	A, C, D	
Develop custom classes using encapsulation, polymorphism, inheritance, and abstraction.			1, 13, 14, 15, 16, 2	A, C, D	
Learn how to use files, exceptions and build GUIs.			1, 13, 14, 15, 16, 2	A, C, D	
Analyze and design strategies for solving basic programming problems.			1, 13, 14, 15, 2	A, C, D	
Teaching Methods	1: Lecture, 13: Experiment / Laboratory, 14: Self-Study, 15: Problem solving, 16: Project Based Learning, 2: Question - Answer				
Assessment Methods	A: Written Exam, C: Homework, D: Project / Design				
Lecture Schedule					
Sequence	Topics	Preliminary Preparation			
1	Introduction to Computers, Programming, and Python				
2	Elementary Programming				
3	Mathematical Functions, Strings, and Objects				
4	Selection statements				
5	Loops				
6	Loops				
7	Functions				
8	Lists, Multidimensional Lists, Tuples, Sets, and Dictionaries				
9	Basic GUI Programming				
10	Advanced GUI Programming				
11	Inheritance and Polymorphism				
12	Files and Exception Handling				
13	Recursion				
14	Developing Efficient Algorithms				
Evaluation Methods		Weight(%)			
Midterm Exam		30			
General Exam		70			

Resources
- Y. Daniel Liang, Introduction to Programming Using Python, 2nd Ed., Pearson, 2019. - Allen B. Downey, Think Python How to Think Like a Computer Scientist 2nd Ed., O'Reilly Media, 2015.Lecture notes that will be delivered during the classes.