ISTANBUL MEDIPOL UNIVERSITY
SYLLABUS

COE1210754, EEE1210754, IND1210754 & CEE1210754
Advanced Programming

2020 Spring Semester

Course Code	Course Name	Course Type	Weekly T	Weekly A	Weekly L	Credits	ECTS	Weekly Class Schedule
COE1210754 EEE1210754 IND1210754 CEE1210754	Advanced Programming	Required	2	0	2	3	5	Mon 13:30-16:30

Prerequisite	Introduction to Programming	**Prerequisite to**	-

Lecturer	Selim AKYOKUŞ	**Office Hours Schedule**	Monday 16:30
E-mail	sakyokus@medipol.edu.tr		
Phone	216 651 0000 x 5350	**Office / Room No**	C-320 North Campus
Assistants	- Asst A,B,C		
E-mail	- A@medipol.edu.tr		

Course Objectives

The objective of this course is to improve programming and problem solving capabilities and skills of students using Python with an emphasis on programming practice, efficiency and data science. Pyhton is widely used language in education, scientific computing and data science with a large number of libraries. Students will learn, design, develop and test efficient programs that take advantage of built-in libraries developed for AI and data science without having to know about complex logic and mathematics behind them. Topics include programming efficiency and analysis, study and analysis of some basic algorithms, graphical user interfaces, advanced featues of Python, Python Data Structures, Loading Datasets from Different Data Stores, Array-Oriented Programming with NumPy, High-Performance NumPy Arrays, Pandas Series and DataFrames, Regular Expressions and Data Wrangling, Time Series and Simple Linear Regression, Natural Language Processing (NLP), Web Scraping, Data Mining Twitter: Sentiment Analysis, Machine Learning: Classification, Regression and Clustering, Deep Learning Convolutional and Recurrent Neural Networks, Recommendations with Collaborative Filtering, Optimization

Textbook

- Paul J. Deitel et al., Intro to Python for Computer Science and Data Science: Learning to Program with AI, Big Data and The Cloud, Pearson, 2020. [DE]
- Toby Segaran, Programming Collective Intelligence, O'Reilly Press, 2007. [TO]
- Brad Miller and David Ranum, Luther College, Problem Solving with Algorithms and Data Structures using Python, Franklin, Beedle & Associates, 2011. [BR] https://runestone.academy/runestone/books/published/pythonds/index.html
- Y. Daniel Liang, Introduction to Programming Using Python, Pearson. [LI]
- How to think like a computer scientist https://runestone.academy/runestone/books/published/thinkcspy/index.html
- Richard L. Halterman, FundamentalsofProgramming http://python.cs.southern.edu/pythonbook/pythonbook.pdf
- Python Practice Book https://anandology.com/python-practice-book/index.html

Learning Outcomes

After successful completion of the course, the student will be able to:

1	Design, implement and test efficient programs
2	Improve programming skills by learning, analyzing, solving and developing program code for different problems
3	Learn how to design, develop and implement modular programs by using structured programming, abstract data types, classes and objects.
4	Take advantage of capabilities of built-in and third party libraries available in many areas.
5	Learn how to store, load, manipulate and explore data.
6	Summarize, visualize and analyze data.
7	Write programs for a wide variety problems in math, science, engineering, financials, and games.
8	Learn how to use and apply some of machine learning, data mining and optimization libraries on several examples.

Teaching Methods	Class discussions with examples. Labs for the demonstration	
WEEK	**TENTATIVE TOPICS**	**REFERENCE**
Week 1	Developing Efficient Algorithms	Ch 16 [LI]
Week 2	Analysis of Searching and Sorting Algorithms	Ch 17 [LI]
Week 3	Array-Oriented and Scientific Programming with NumPy and SciPyPython	Ch 7 [DE]
Week 4	Python Data Structures & Pandas	Class Notes and Web
Week 5	Regular Expressions and Data Wrangling	Ch 8 [DE] and Web Resources
Week 6	Data Analysis and Visualization	Class Notes and Web
Week 7	Time Series and Simple Linear Regression	Ch 10 [DE] and Web Resources
Week 8	Exam Week	
Week 9	Natural Language Processing (NLP), Web Scraping	Ch 12 [DE]
Week 10	Data Mining Twitter: Sentiment Analysis, JSON and Web Services	Ch 13 [DE]
Week 11	Machine Learning: Classification, Regression and Clustering	Ch 14 & 15 [DE]
Week 12	Deep Learning Convolutional and Recurrent Neural Networks	Ch 16 [DE]
Week 13	Collaborative Filtering, Making Recommendations	Ch 2 [TO]
Week 14	Optimization	Class Notes and Web
Week 15	Review	

Assessment Methods and Criteria

Evaluation Tool	Quantity	Weight
Final Exam	**1**	**30%**
Midterm exam	**1**	**20%**
Quizes	**2**	**16%**
Lab Exercises	**1**	**10%**
Programming Assignments	**8**	**20%**
Attendence	**1**	**4%**
Total		**100%**

*** ECTS Credit Calculation ***

Language of Instruction: English

Activity	Hours	Weeks	Student Workload Hours	Activity	Hours	Weeks	Student Workload Hours
Lecture hours	2	14	28.0	In-term exam study	8	3	24.0
Labs	2	10	20.0	Final exam study	13	1	13.0
Programming Assignments	6	8	48.0				
				Total Workload Hours =			133.0
				Recommended ECTS Credit =			5