MARYLAND APPLIED
GRADUATE ENGINEERING

PYTHON PROGRAMMING FOR ENGINEERING AI
currently ENPM818W, will transition to ENAI607

TENTATIVE SYLLABUS – SUBJECT TO CHANGE

Term: *Fall 2025* **Email:** zsafar@umd.edu
Professor: Zoltan Safar
Course Dates: From September 2nd 2025 – December 12th 2025
Course Times: Tuesday 4:00PM – 6:40PM
Classroom: TBD

Course Description

This course covers topics including:

- Basic Python program structure, variables and assignments, built-in data types, strings, lists, tuples and dictionaries, control flow, functions and modules; basic I/O and file operations.
- Classes, object-oriented programming and exceptions.
- Introduction to algorithms and data structures: recursion, searching, graph algorithms.
- Priority queues, search trees and hash tables.
- Algorithms for AI and machine learning: regression, classification, and clustering.
- Introduction to the Numpy, Scipy and Matplotlib libraries.

Prerequisites

Some background in programming and discrete mathematics is useful, but not required.

Learning Outcomes

After successfully completing this course you will be able to:

- understand and utilize basic Python syntax,
- work with simple and complex data types like integers, strings, lists, and dictionaries,
- implement control flow structures such as loops and conditionals, and define functions and modules,
- understand object-oriented programming concepts, including class structure, objects, encapsulation, inheritance and polymorphism,
- implement and apply various algorithms and data structures to solve problems efficiently
- utilize frequently used Python libraries to build solutions to complex problems

Course Assignments

Homework	10%
Quizzes	10%
Midterm exam	20%
Project	30%
Final exam	30%

Course Topics

Week	Topic
1	Python program structure, variables and assignments, simple data types: integers, booleans, floating-point numbers, complex numbers, strings, string formatting, console I/O.
2	Collection data types: tuples, lists, dictionaries; iterating and copying collections.
3	Control flow: conditional branching and looping; functions and arguments, lambda functions, exception handling; modules, imports and packages.
4	Classes, object-oriented programming, attributes and methods, encapsulation, inheritance and polymorphism.
5	File I/O operations, file system operations and interacting with the operating system.
6	Introduction to the Numpy, Scipy and Matplotlib libraries: numpy arrays, vectorized operations, solving linear systems, matrix decompositions, statistical functions and data visualization.
7	Midterm exam
8	Recursion, linear and binary search, quadratic sorting: selection sort and insertion sort.
9	Queues, stacks and linked lists.
10	Heaps, heapsort, and priority queues.
11	Graphs, graph representation, breadth-first search, depth-first search, topological sort.
12	Hash tables, binary search trees, kd-trees.
13	Algorithms for regression: linear regression, nearest neighbors; algorithms for clustering: k-means, mixture models, the EM algorithm
14	Algorithms for classification: linear/quadratic discriminant analysis, logistic regression, nearest neighbors.

Note: This is a tentative schedule, and subject to change as necessary – monitor ELMS-Canvas for current deadlines. In the unlikely event of a prolonged university closing, or an extended absence from the university, adjustments to the course schedule, deadlines, and assignments will be made based on the duration of the closing and the specific dates missed.