

EE 160 Programming for Engineers

Credits: 4

Categorization of credits: engineering topic

Instructor(s): Matthias Fripp

Textbook and Other Required Materials:

Bailey Miller, “Programming in Python 3”, zyBooks.com (Wiley).

Designation: Required

Catalog Description:

EE 160 Programming for Engineers (4) (3 Lec, 1 3-hr Lab) Introductory course on computer programming and modern computing environments with an emphasis on algorithm and program design, implementation, and debugging. Includes a hands-on laboratory to develop and practice programming skills. A-F only. Pre: Math 241 (or concurrent) or Math 251A (or concurrent) or consent.

Pre- and Co-requisites: Pre-requisites: Math 241 (or concurrent) or Math 251A (or concurrent) or consent.

Class/Lab Schedule: 3 lecture hours per week, 1 3-hr lab per week.

Topics Covered:

- Introduction to computers and programming
- Variables and Expressions
- Data Types
- Branching
- Loops
- Functions
- Troubleshooting
- Debugging and Testing
- Strings
- Lists and Dictionaries (arrays and associative arrays)
- Classes
- Modules
- File Input/Output
- Inheritance
- Recursion
- Plotting and Arrays
- Searching and Sorting Algorithms
- Running Python Locally
- Installing and Configuring Python

Course Objectives and Relationship to Program Objectives:

A student should understand (i) the basic design process in going from a specification,

developing an algorithm, implementing the algorithm, and testing and verifying programs of small to medium size, (ii) the fundamental programming constructs including variables, scalar and compound data types, functions, and looping and branching, and (iii) the use of library functions and interaction with the operating system. Students should be able to work in teams to meet learning objectives and complete assignments. [Program Objectives this course addresses: 1, 2 and 4.]

Course Outcomes and Their Relationship to Program Outcomes

- Write an algorithm to solve a problem. [1, 2]
- Design modular programs. [1, 2]
- Create variables of an appropriate type and write expressions to perform computation. [2, 7]
- Define functions to perform subtasks and pass parameters. [1, 2, 7]
- Use scalar and compound data types as appropriate for the application. [1, 2, 7]
- Use standard library functions for I/O and computation. [2, 7]
- Define classes and use inheritance to encapsulate and extend functionality. [1, 2, 7]
- Use recursion as a computational technique when appropriate. [1, 2, 7]
- Run, troubleshoot and debug Python programs. [7]
- Work in effective teams to solve complex problems. [5, 7]

Contribution of Course to Meeting the Professional Component

Engineering topics: 100%

Computer Usage:

There is extensive computer usage as this is a programming course. Students use a web-hosted development environment to write, test and submit assignments for grading. Students also learn to install, configure and run Python on their local computer. All course related material is provided on-line.

Design Credits and Features:

This is an introductory programming course and carries no design credits. Students are exposed to the engineering design process as it applies to software engineering and are given an opportunity in homework and labs to practice the process in developing computer solutions to problems.

Person(s) Preparing Syllabus and Date: T. Dobry, 12 Feb. 2009. Revised by Matthias Fripp, Jan. 21, 2021.