

COURSE OUTLINE

ENES-159

Programming Concepts for Engineers

4 Credits

HOWARD COMMUNITY COLLEGE

Description

This course will introduce students to intermediate principles of software development and will include high-level languages, object-oriented design, documentation, data structures, graphs and dynamic memory allocation. Students will become experienced in software applications in electrical and computer engineering, and software development in teams. Programs will utilize the C and Java languages under the Windows/cygwin environment. Software development projects will involve relevant engineering topics, such as analysis of digital and analog circuits, cryptography, bioinformatics, embedded software, game programming, image processing, and wireless sensor networks. Prerequisite: ENES-100. (4 hours lecture)

Overall Course Objectives

Upon completion of this course, the student will be able to:

1. Develop robust and extensible software through effective software engineering practices.
2. Apply object-oriented design and complex data structures.
3. Demonstrate the ability to independently learn new languages and libraries.
4. Create test suites and periodic builds, as a member of a team, using version control.
5. Demonstrate an ability to use GNU and Common Public license tools such as cygwin and graphviz.
6. Demonstrate an ability to model objects using the standardized specification language UML.

Major Topics

- I. Unix Concepts
- II. Coding Conventions and Style
- III. Unit Testing
- IV. Separate Compilation and Makefiles
- V. Pointers
- VI. Dynamic Memory Allocation
- VII. Linked Lists
- VIII. Brief Introduction to Digital Circuits
- IX. Graphs
 - A. Mathematical Definition
 - B. Adjacency List and Adjacency Matrix Representations
 - C. Displaying Graphs Using Graphviz
 - D. Representing and Analyzing Digital Logic Circuits
- X. Team-Based Software Projects
 - A. Version Control Systems/Subversion
 - B. Test Suites and Periodic Builds
 - C. Team Projects on Analysis of Digital Logic Circuits

- XI. Other Dynamic Structures
- XII. Abstract Data Types
- XIII. Object-Oriented Design
 - A. Implementation in C
 - B. Implementation in Java
- XIV. The Unified Modeling Language
- XV. Presentations

Course Requirements

Grading/exams: Grading procedures will be determined by the individual faculty member. Grades will be based on homework, tests, and class participation.

Other Course Information

This course is required in the Engineering AA Degree Program.