ECE2035 Course Syllabus

ECE2035

Programming for Hardware/Software Systems (3-0-3-4)

CMPE Degree
This course is Required for the CMPE degree.

EE Degree
This course is Elective for the EE degree.

Lab Hours
0 supervised lab hours and 3 unsupervised lab hours

Course Coordinator
Wills,Linda M

Prerequisites
ECE 2020 [min C]

Corequisites
None

Catalog Description
Creation of complex execution and storage mechanisms, based on instruction set architecture, for software design including high-level programming languages and operating systems. Programming design projects.

Textbook(s)
Patt & Patel, *Introduction to Computing Systems* (2nd edition), McGraw-Hill, 2004. ISBN 0072467509, ISBN 978-0072467505 (required)

Course Outcomes
Upon successful completion of this course, students should be able to:

1. Develop a software design using appropriate data and procedural abstractions given an engineering problem specification
2. Implement high-level programming language storage, control, and procedural constructs in the assembly language of a hardware platform.
3. Use commonly available tools for software development, system management, debugging, performance monitoring, and optimization.
4. Develop a software system that uses multiple files and libraries.
5. Develop, test, and deploy reliable software using appropriate operating system capabilities (e.g., file and directory access, I/O interface).

Student Outcomes
In the parentheses for each Student Outcome:
"P" for primary indicates the outcome is a major focus of the entire course.
“M” for moderate indicates the outcome is the focus of at least one component of the course, but not majority of course material.
“LN” for “little to none” indicates that the course does not contribute significantly to this

outcome.

1. (P) An ability to identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics
2. (LN) An ability to apply engineering design to produce solutions that meet specified needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors
3. (LN) An ability to communicate effectively with a range of audiences
4. (LN) An ability to recognize ethical and professional responsibilities in engineering situations and make informed judgments, which must consider the impact of engineering solutions in global, economic, environmental, and societal contexts
5. (LN) An ability to function effectively on a team whose members together provide leadership, create a collaborative and inclusive environment, establish goals, plan tasks, and meet objectives
6. (LN) An ability to develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions
7. (M) An ability to acquire and apply new knowledge as needed, using appropriate learning strategies.

Topical Outline

1. Introduction to High Level Language and Assembly Programmin

2. High Level Programming Language Syntax

3. Processing: ISA datapath and controller

4. Control: conditionals, iteration, recursion

5. Storage
   a. static
   b. stack
   c. heap

6. Supporting Procedural Abstraction
   a. procedure calling
   b. activation frames

7. Supporting Data Abstraction
   a. structs
   b. arrays
   c. linked lists
   d. hash tables

8. Compilation, assembly, linking, loading, libraries

9. File Systems and I/O

10. Exception and Interrupt Handlers

11. Software Performance & Reliability
    a. testing
    b. debugging
    c. performance monitoring

12. Embedded Software
    a. interaction with the execution platform
    b. energy efficiency

13. Basic Concurrency in Multicore Systems
    a. data-level parallelism
    b. support for atomicity

Optional Advanced Topics:

14. Dynamic typing

15. Automatic storage reclamation (garbage collection)

16. Security issues (e.g., software protection against buffer o