|  |  |
| --- | --- |
| **Course** | ECE 30300 – Engineering Software Design |
| **Type of Course** | Elective for CmpE and EE programs |
| **Catalog Description** | The purpose of this course is to introduce a variety of advanced programming and software design tools to Electrical and Computer engineering students, with an emphasis on problem solving. Topics include object-oriented programming, Unix shell script programming, advanced programming techniques in both compiled and interpreted languages, as well as programmable logic controller (PLC) programming. |
| **Credits** | 3 |
| **Contact Hours** | Lecture 2: Lab: 2 |
| **Prerequisite Courses** | ECE 22900 |
| **Recommended Textbooks** | 1. S. Prata, C*++ Primer Plus,* Pearson, current edition. 2. M. Sobell, *A Practical Guide to Linux Commands, Editors, and Shell Programming*, Prentice Hall, current edition. 3. J. Guttag, *Introduction to Computation and Programming Using Python: with Application to Understanding Data*, MIT Press, current edition. 4. C. Green, *Learn How to Program and Troubleshoot Ladder Logic*, CreateSpace, current edition. |
| **Course Objectives** | This course introduces a variety of advanced programming and software tools, including object-oriented programming in C++, Unix shell scripting, Python programming language, and PLC programming. Students are expected to design and test software programs in integrated software/hardware systems to solve advanced engineering problems. |
| **Course Outcomes** | On successful completion of this course, students should be able to:   1. Make use of concepts in object-oriented programming to formulate and solve engineering problems. **(1)** 2. Test and evaluate software program for correctness, reliability, and efficiency. **(2)** 3. Use scripting languages to interact with the operating system. **(1)** 4. Use scripting languages to read, analyze and visualize data from a text file. **(6)** 5. Execute software program within integrated software/hardware devices. **(1)** 6. Design, program, and test a basic PLC system to meet a set of specifications. **(2)** |
| **Lecture Topics** | * 1. C++ classes and encapsulation   2. Function overloading and operator overloading   3. Inheritance and polymorphism   4. Templates and C++ standard template library   5. Common C++ debug tools: gdb, make, grof, valgrind, etc.   6. Bash commands and shell scripting   7. Introduction to Python   8. Python: file processing   9. Python: data visualization   10. Electric ladder diagrams   11. PLC programming |
| **Computer Usage** | High |
| **Laboratory Experience** | High |
| **Design Experience** | High |
| **Coordinator** | Chao Chen |
| **Date** | October, 2018 |