

ECET 230 Object-Oriented Programming in Hardware

Hours: 2/2.5/0

Prerequisite: ECET 130 Engineering Applications in C

Short description:

Students will be introduced to design techniques using object-oriented programming. Students will find that emphasis is placed on the development of graphical user interfaces (GUIs) and on the hardware and software interface to electronics applications. They will also cover communication with embedded devices such as the Arduino, or Raspberry PI, or other ARM microcontroller platforms.

Learning outcomes:

Upon successful completion of this course a student will be able to:

- employ object-oriented design techniques to produce solutions to detailed problem specifications;
- design and implement programs that perform low level input and output to non-standard devices using appropriate methods with an object-oriented programming language (OOPL);
- design and implement hardware and software interfaces in an OOPL;
- use an OOPL to program embedded devices;
- design and implement graphical user interfaces (GUIs) to drive low-level communications between PCs and between the PC and embedded devices using the serial and/or USB ports;
- create, test and demonstrate programs that interact with electromechanical machinery via USB;
- make effective use of established protocols for serial input and output (I/O);
- use external file storage methods;
- use industry-standard graphical development software to develop GUI and non-GUI programs for hardware interfaces.

Course outline:

1. Introduction to object-oriented programming 2 hours
 - 1.1 Procedural programming vs object-oriented programming (OOP)
 - 1.2 The Win32 application programming interface (API)
 - 1.3 Graphical user interface (GUI) programming vs command line programming

2. Design of GUIs 2 hours
 - 2.1 Integrated development environment (IDE)
 - 2.2 Designing a GUI program
 - 2.3 A simple GUI example

3.	Object-oriented programming for GUIs	6 hours
3.1	Namespaces	
3.2	References	
3.3	Functions and operators	
3.4	Overloading functions	
3.5	Function templates	
3.6	A linked list implementation	
4.	Object-oriented programming for embedded devices	6 hours
4.1	Input ports	
4.2	Output ports	
4.3	Analog-to-digital (A/D) converter	
4.4	Interaction between embedded devices and a GUI environment	
4.5	Communicating with an embedded device using RS-232 serial	
4.6	Communicating with an embedded device over USB	
5.	Streams and files in C#	2 hours
5.1	Input and output (I/O) streams	
5.2	File I/O	
6.	Classes and objects in C#	6 hours
6.1	Encapsulation, inheritance, and polymorphism	
6.2	Abstract data types	
6.3	Member data and functions	
6.3.1	Access functions	
6.3.2	Helper functions	
6.4	Objects as instances of classes	
6.5	Constructors	
6.5.1	Default	
6.5.2	Parameterized	
6.6	The destructor	
6.7	Class templates	
6.8	Overloading member functions and operators	
6.9	Relationships between classes	
6.10	Friend classes	
	Tests and review	4 hours
	Total	28 hours

EVALUATION (Grading according to College policy):

Marks toward the final grade are distributed as follows:

Final exam:	35%
Quizzes, assignments and Mid-term:	25%
Laboratory assignments:	40%
<u>TOTAL</u>	<u>100%</u>

LATE LAB POLICY: 50% penalty

REFERENCES:

Online Resources:

<https://msdn.microsoft.com>

<http://www.microsoftvirtualacademy.com/>

<http://stackoverflow.com/>