

ECE 131 Programming Fundamentals

Overview

Course Catalog Description: Fundamental programming concepts, including consideration of abstract machine models with emphasis on the memory hierarchy, basic programming constructs, functions, parameter passing, pointers and arrays, file I/O, bit-level operations and interfacing to external devices.

Prerequisites: none

Textbooks:

- S. G. Kochan, *Programming in C: A Complete Introduction to the C Programming Language*, 3rd edition, 2005.
- C. Moler, *Numerical Computing with MATLAB, The MathWorks*, 2nd printing, 2008 (free book downloadable from: www.mathworks.com/moler).

Class Goals: To provide a basic introduction to programming that gives the student an appreciation for the scope of modern programming activities, and provides a solid foundation for further study in this area. The focus of the course is on teaching the fundamental building blocks of programming, and as such we should not worry too much about the language that is used -- we should think of the language as the vehicle for presenting the programming topics. With that said, the things we look for in a programming language are simplicity and relevance. That is, a language that has a bare bones syntax that will not overwhelm the students is needed. In addition, if the language is also currently a major programming language, then we will be providing students skills with tools that will be relevant to their profession. For our purposes, since we are teaching ECE students, a language that exposes the student to the architecture is desired. Since we will be teaching topics that deal with interfacing, we want to make the memory model explicit. The C programming language provides a good vehicle for teaching the basics, and the Matlab programming environment provides a good vehicle for teaching the students about interpreted and visual programming languages.

Course Coordinator: Prof. Gregory L. Heileman

Table I: Objectives and Implementation

Objectives		Implementation	A	B	C	D	E	F	G	H	I	J	K
O ₁	History of programming, basic programming models and programming paradigms.	Lectures					X					X	
O ₂	How to make programs: edit-compile-link, prototyping & testing, makefiles, project organization.	Lectures and programming assignments	X		X								
O ₃	Basic programming elements: numbers & their representation, program flow, control structures.	Lectures and programming assignments	X		X								
O ₄	Pointers & arrays: indirect addressing, pointer arithmetic, array indexing, memory management.	Lectures and programming assignments	X		X								
O ₅	Functions: separate compilation and subroutines, formal vs. actual parameters, call-by-reference and call-by-value.	Lectures and programming assignments	X		X								
O ₆	File I/O, bit-level operations, and computer interfacing: how computers interact with the world through programs.	Lectures and programming assignments and robotics project		X	X	X	X		X				X

<see attached>

Sample Course Schedule