COMPSCI 253: Intro to Systems Programming

Logistics

Instructor: Dr. Amit Jain

Office: MEC 302E      Phone: 426-3821      Email: ajain@boisestate.edu

Class Home Page: URL: http://cs.boisestate.edu/~amit/teaching/253/cs253.html

Class Time: Tu Th 12:00pm–1:15pm      Room: MEC 307

Office Hours: On the class home page.

Teaching Assistants: Nic Cornia, Farhad Rasapour, Nilab Mousa

See CS tutoring center website for schedule.

http://coen.boisestate.edu/cs/computer-science-tutoring-center-cstc/tutoring-calendar/

Catalog Description

COMPSCI 253 INTRODUCTION TO SYSTEMS PROGRAMMING (3-0-3)(F,S). Structure of C programs, function pointers, variable argument lists, other generic programming techniques. Introduction to build systems, debugging techniques and process management. Basic systems programming including topics such as streams, buffers and pipes, system calls, multi-threading, and libraries for Linux and Microsoft Windows. PREREQ: COMPSCI 221.

Objectives

By taking this the course the student will be able to:

• design and develop programs of moderate complexity in C,

• translate their knowledge of object-oriented programming in Java to C,

• use various tools like IDEs, build tools, debuggers, version control and memory checkers to improve their productivity,

• use shell commands and system utilities, and

• use basic system calls related to files, processes and threads.
Topics

- Introduction
- Topics in C
  - Tutorial introduction to C
  - Types, operators and expressions
  - Control flow
  - Functions and program structure
  - Pointers and arrays (and C-style strings)
  - Structures
  - Input and output
  - Variable argument lists: C style method overloading
  - Function pointers and generic programming in C (with case studies)
  - Creating and using shared libraries in C
  - Plugins in C (aka Method overriding using loadable code)
- Tools and Techniques
  - Make: object-oriented build process
  - Using Eclipse IDE with Make
  - Debugging Techniques using Eclipse, GDB, DDD
  - Memory checker: Valgrind
  - Subversion: introduction to version control
- The Linux/Unix Environment
- Shell scripts and system utilities
  - Advanced command-line tools
  - Object oriented use of Unix filters and pipes
  - Basic shell scripting
- Basic systems programming
  - Linux Systems Programming: files, buffers, processes, signals and pipes.
  - Multi-threaded programming with PThreads library.
  - Multi-threaded programming in Microsoft Windows API.
Prerequisites

COMPSCI 221 (Computer Science II)

Grading

- Programming projects: 50%
- In-class quizzes: 30%
- Final exam: 20%

Programming Projects:

- Programming projects can be turned in two days late for a 10% penalty without getting pre-approval from the instructor.
- Turning in a project more than two days late requires prior approval from the instructor and is unlikely to be granted other than for exceptional reasons.

In-Class Quizzes:

- In-class quizzes must be taken in class on the day they are given. Make-up quizzes will not be granted other than for exceptional reasons.
- Credit for group portion of the quizzes will only be given to group members who participate in the quiz.
- If 80% of students complete the end-of-semester course evaluations, then the lowest quiz score will be dropped.

Textbooks and References

- Lecture notes (on class website).
  URL: http://cs.boisestate.edu/~amit/teaching/handouts/cs-unix.pdf
- *Managing Projects with Make*, Oram and Talbott.
Academic Honesty

Students are expected to work on their own unless explicitly instructed otherwise. Students who copy from each other or from any other source on assignments will be considered to be cheating as will students who allow their work to be copied. Cheating is grounds for immediate failure of the course. This includes trying to find answers to problems or programs from the Internet or other sources (and uploading your completed assignments to Internet sites that are publicly accessible). For more information, see the following web page:

http://registrar.boisestate.edu/general-information-and-policies/academic-integrity/