Welcome to CSE2003: System Programming (Spring 2010)

Jin-Soo Kim (jinsookim@skku.edu)
Computer Systems Laboratory
Sungkyunkwan University
http://csl.skku.edu
Introduction

- Basic information
  - Class 41 (KOR): 15:00 – 16:15 (M), 12:00 – 13:15 (Th)
  - Class 42 (ENG): 16:30 – 17:45 (M), 13:30 – 14:45 (Th)
  - Lecture room #400102 (Semiconductor Bldg.)

- Instructor
  - Jin-Soo Kim ([jinsookim@skku.edu](mailto:jinsookim@skku.edu))
  - Computer Systems Laboratory ([http://csl.skku.edu](http://csl.skku.edu))
  - Office: Semiconductor Bldg. #400630 (6th floor)
  - Tel: 031-299-4593
  - The best way to contact me is by email.
Course Outline (1)

- **What is “System Programming”***?
  - Low-level programming?
  - Learning about system programs?
  - Assembly programming?
  - Programming with operating system services?

- **We try to answer the following questions:**
  - How does the computer system work?
  - How does your program run?
  - How to make your program run faster?
  - How to make your program more robust?
  - ...
Course Outline (2)

- Computer systems
- Computer systems internals

Software

Architecture

Hardware

Operating Systems

CPU

Mem

I/O Devices

Application
Course Outline (4)

- Levels of abstraction

- Application programs
- Data structures & algorithms
- Programming languages & compilers
- Operating System
- Architecture
- Microarchitecture
- Hardware Description Languages
- Digital logic
- VLSI layout
- Processing, Fabrication
- Chemistry, Physics
Course Plan

- "System" programming
  - Representing and manipulating information
  - Assembly languages
  - Processor architecture
  - Memory hierarchy
  - Compilers, linkers, and loaders
  - Operating systems
  - Performance optimization

- System “programming”
  - Assembly programming
Prerequisites

- C programming skills (GEDD07)
- Basic knowledge of UNIX/Linux systems
  - ICE2001: Logic circuits
  - ICE2002: Data structures
Course Components

- **Lectures**
  - Concepts
  - Backgrounds

- **Quizzes**
  - On topics covered in previous classes

- **Projects**
  - Mostly on assembly programming
  - Design, implementation, measurement, optimization
Textbook

- **Computer Systems:** A Programmer’s Perspective
  - Authors’ homepage: http://csapp.cs.cmu.edu
C Programming

- C Programming Language, (Second Edition)
  B. Kernighan and D. Ritchie,
References (2)

- **Assembly Programming**
  - *The Art of Assembly Language Programming*, Randall Hyde,
    http://webster.cs.ucr.edu/
  - Intel Architectures Software Developer’s Manual,
    Volume 1: Basic Architecture
    Volume 2: Instruction Set Reference
    Volume 3: System Programming Guide
Class Policies (1)

- **Grading Policy (subject to change)**
  - Class attendance (10%)
  - Projects (35%)
  - Exams, Quizzes (55%)

  - If you miss one or both of exams, you will fail this course.
  - Do not be late! You should be present in the lecture room when I take class attendance.
  - You have **four** “tokens”; these tokens can be used for unexcused absences and **for excused absences as well** (i.e., up to four absences will be OK).
Class Policies (2)

- **Project Policy**
  - You will work on each project alone
  - The submission status will be posted on the course homepage.
  - Only the assignments submitted before the deadline will receive full credit.
  - 25% of the credit will be deducted for every single day delay.
Class Policies (3)

- Cheating Policy
  - “Copying all or part of another person’s work, or using reference material not specifically allowed, are forms of cheating and will not be tolerated,”
  - For a student involved in an incident of cheating, the following policy will apply:
    - You will get zero points in the particular project and the final grade will be lowered by one grade (e.g., B+ \(\rightarrow\) B)
    - For serious offenses, you will get F and this will be notified to the department chair
  - Share useful information: helping others use systems or tools, helping them with high-level designs or debug their code is NOT cheating!
Any Questions?