Welcome to SWE2007: Software Experiment 2

Woo-Yeong Jeong (wooyeong@csl.skku.edu)
Computer Systems Laboratory
Sungkyunkwan University
http://csl.skku.edu
Introduction

- **Schedule**
  - 18:30 – 21:45 (Wed)
  - Lecture room: #400202, Semiconductor Bldg.

- **Course homepage**
  - All materials will be posted on this site.
  - We do not use i-campus system.
About Professor

- **Jin-Soo Kim**
  - Professor @ CE & SSE & SW Dept.
  - Computer Systems Laboratory

- Office: 산학협력센터 #85566 (5th floor)
- Email: jinsookim@skku.edu
- URL: http://csl.skku.edu/jinsoo
- Tel: 031-299-4593
- Office hours: Monday & Wednesday
- The best way to contact him is by email.
About Me

- **Woo-Yeong Jeong**
  - TA of this class
  - MS student
  - Computer Systems Laboratory

- Office: 산학협력센터 #85533 (5th floor)
- Email: wooyeong@csl.skku.edu
- The preferred way to contact me is (also) by email.
Looking up host 'csl.skku.edu'...
Host 'csl.skku.edu' resolved to 115.145.179.100.
Connecting to 115.145.179.100
Connection established.
To escape to local shell, press 'Ctrl+Alt+]'.

Linux csl       #1 SMP

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

You have mail.
Last login: Mon Sep 1 00:22:40 2014 from 59.15
wooyeong@csl:~$
wooyeong@csl:~$
wooyeong@csl:~$ whoami
wooyeong
wooyeong@csl:~$ what should i do?
-bash: what: command not found
wooyeong@csl:~$
wooyeong@csl:~$ help me T_T
-bash: help: no help topics match `T_T'. Try `help help' or `man -k T_T' or `info T_T'.
wooyeong@csl:~$
Course Outline (2)

User Space
- cd
- ls
- vi
- wget

Kernel Space
- System Call Interface
  - Operating System (Kernel)

Hardware
- CPU
- MEM
- Disk
- NIC
Course Outline (3)

- Why we use Linux?
  - Used in many scientific and industrial settings
  - Internet servers and services run on Linux
  - It’s free!

- How to use Linux?
- How to make programs on Linux?
- How to make [advanced] programs on Linux?
  - We will learn various system calls provided by Linux systems
Topics

- **Very basic Linux commands**
  - Shell, text editor, compiler

- **Basic Linux system calls**
  - File I/O, Process management
  - Inter-Process Communication (IPC)

- **Network programming**
  - Sockets

- **Concurrent programming**
  - Processes, Threads
Projects (1)

- Basic Linux system calls
  - File I/O
  - Process management
  - Inter-Process Communication (IPC)

- Concurrent programming
  - Processes
  - Threads

- Network programming
  - Sockets
Projects (2)

- Policies
  - There will be 4~5+ lab exercises
  - We will solve one or two term projects
  - Each project must be done individually
Projects (3)

- Evaluation

• Your code will be evaluated in the following ways:
  – Demonstration
  – Documentation
  – Your progress

• You should be able to answer any questions on basic system architecture, design decisions, and implementation details

• Always pay attention to
  – Performance issues
  – Documentation
Reference

- **Computer Systems: A Programmer’s Perspective**
  - [http://csapp.cs.cmu.edu](http://csapp.cs.cmu.edu)
Class Policies (1)

- Grading Policy (subject to change)
  - Class attendance (10%)
  - Lab exercises (50%)
  - Term projects (40%)

- There will be no exams (subject to change)
Class Policies (2)

- Cheating Policy
  - What is cheating?
    - Copying another student’s solution (or one from the Internet) and submitting it as your own
    - Allowing another student to copy your solution
  - What is NOT cheating?
    - Helping others use systems or tools
    - Helping others with high-level design issues
    - Helping others debug their code
  - Penalty for cheating:
    - Severe penalty on the grade and report to dept. chair
  - Ask helps to your TA if you experience any difficulty!
Any Questions?