

SWE3019: Software Experiment 3

Hyo-bong Son(proshb@csl.skku.edu)
Computer Systems Laboratory
Sungkyunkwan University
<http://csl.skku.edu>



Introduction



▪ Schedule

- 15:00 – 18:50 (Wed)
- Lecture room #400202 (Semiconductor Bldg.)

▪ Course homepage

- <http://csl.skku.edu/SWE3019S13/>

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
- The best way to contact me is by email.

About Me



■ Hyo-bong Son

- MS student
- Computer Systems Laboratory
- Office: 산학협력센터 #85561 (5th floor)
- Email: proshb@csl.skku.edu
- E-mail, SMS, Kakao-talk, etc..

Course Plan (1)



- **Introduction to computer systems**
 - We will learn various system calls provided by Linux systems for advanced programming.
- **Main topics**
 - Basic Unix system calls
 - File I/O, Process management
 - Inter-Process Communication (IPC)
 - Network programming
 - Sockets
 - Concurrent programming
 - Processes, Threads

Course Plan (2)

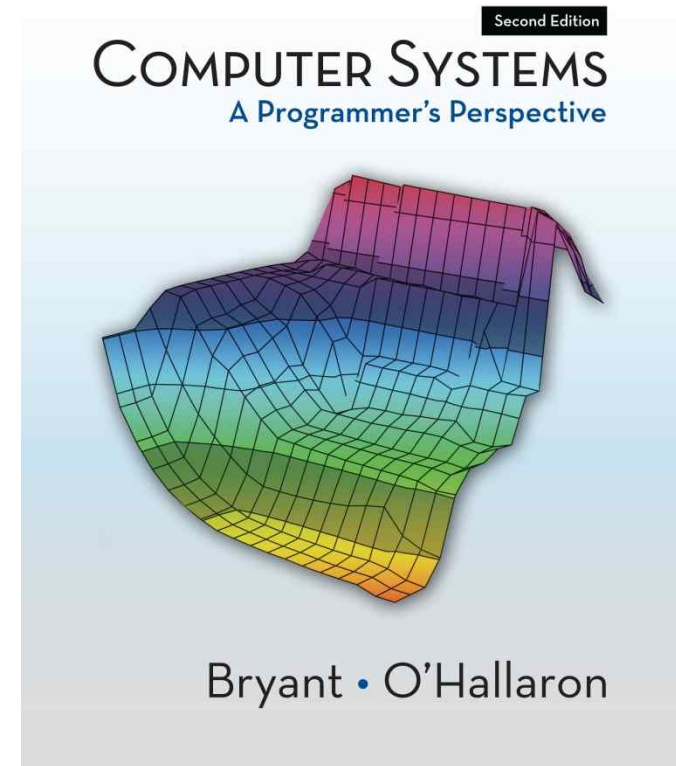
■ Projects: Google

- Midterm
 - How well you manage C language
 - Some portion of data structure
 - File I/O & socket programming
- Final
 - Concurrent programming
 - How well you optimize your own masterpiece

Textbook

- **Computer Systems:
A Programmer's Perspective**

- Randal E. Bryant and David R. O'Hallaron, **Second Edition**, Prentice-Hall, Inc. 2010.
- <http://csapp.cs.cmu.edu>



Class Policies (1)

- **Grading Policy (subject to change)**
 - Class attendance (10%)
 - Lab Exercises (10%) > 30%
 - Programming Assignments (80%) > 60%
- **Project Policy**
 - Projects must be maintain on VCS
 - Individual repository will be supported.

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?





CODING IN LINUX

+ HOW I GRADE

Coding standard (1)

- **The best code is coded properly.**
 - It means that code not only does its job well, but is also easy to add to, maintain and debug!
- **Pros**
 - Allows developers to easily share code.
 - Looks better
- **Cons**
 - It's bothering

Coding standard (2)

■ Comments

```
#ifndef DELIVERCLASS_H_
#define DELIVERCLASS_H_

#include "QueueClass.h"

#include "../Header/InterpretClass.h"

#include "../Header/Types.h"

#include <pthread.h>

// *****
// Author : JL
// This class manage headers by Queue. Receiving and Sending data is automatically
// done by additional thread. We only push at 'clInput'(name of queue structure)
// and then another thread send it to host stub. We only pop at 'clOutput' if it
// isn't empty then this header form is sent by host stub. But these all features are
// private value of this class, so I provide some API to use this class well.
//
// API for DeliverClass
// InitThread   : Initialize and start thread. It will call SetDesc, StartThread
// SetDesc      : Set Input, Output descriptor. This will be set by Nonblock mode
// StartThread  : Check if thread started before. If not, start thread
// StopThread   : Check if thread started before. If yes, stop it.
// RecvData     : Receive data from Receive Queue. This will return header class
// SendData     : Send data to Send Queue.
//
// You only use InitThread at initialize this class
// At running, you have to use RecvData, SendData to communicate with host stub
// At closing, call StopThread API to synchronized thread exit
// *****
class DeliverClass {
public:
    DeliverClass();
    virtual ~DeliverClass();
```

Coding standard (3)

■ Naming convention

- Class, enum, typedef : Suffix
 - Calendar**Class**, Month**Enum**, Day**Type**
- Variables : Capitalization, type prefix
 - **n**Day**Value**, **e**nMonth**Value**, **p**stCalender
 - » Int : n
 - » Char : ch
 - » Struct : st
 - » Pointer : p
- Function : Capitalization
 - **Get**Day**Value**, **Set**Month**Value**

```
const int x = 5;

int main(void)
{
    const int x = 3;
    ...
    printf("%d\n", x);

    return 0;
}
```

Coding standard (4)

- **Physical formatting**

- Code alignment
 - Tab for 4 spaces
- Brace Rule
 - Locate {} at conditional statement
- etc.

Sample code

- **By Jinhyuk Lee @ CSL**

- @ csl.skku.edu/SWE3019S13/Schedule
- http://csl.skku.edu/uploads/SWE3019S13/Sample_code.zip