Creative Software Design

Course Intro

Yunho Kim
yunhokim@hanyang.ac.kr
Dept. of Computer Science

Course Information

- Instructor: Yunho Kim (김윤호)
  - yunhokim@hanyang.ac.kr, ITBT 614

- TA
  - Jina Jung (정지나), snowgina00@hanyang.ac.kr
  - Taebin Bang (방태빈), kevinbang@hanyang.ac.kr

- Course Homepage
  - Hanyang LMS (https://learning.hanyang.ac.kr/courses/160619)

SMART-F Lecture

- SMART-F lecture is conducted through a combination of **online** and in-person sessions

- You will watch one online lecture video and take two in-person sessions for each week

Lectures & Labs

- Lecture (Mon) + Labs (Wed, Fri)

- Lecture (by instructor)
  - Traditional-style learning through recorded video
  - I will open a Zoom session every Monday 3:00pm~5:00pm for live Q&A
    - There will be no in-person classes on Mondays unless otherwise notified.

- Labs (by TA)
  - Solving programming assignments by yourselves.
  - TAs will help you.

Course Overview

- In this course you will
  - Learn the fundamentals of C++ language
    - Key concepts of object-oriented programming such as classes, inheritance, data hiding, and polymorphism
    - Generic programming using template
    - Memory management such as references, pointers, dynamic memory allocation
  - Practice C++ programming skills through exercises
  - Practice basics of testing and debugging skills

Schedule (subject to change)

Week	Topic
1	Introduction to Course, Review of the C programming language
2	Introduction to C++
3	Class
4	Inheritance
5	Overloading
6	Pointers and References in C++
7	Polymorphism part1
8	Mid-term exam
9	Polymorphism part2
10	Template
11	Standard Template Library(STL)
12	Exception
13	Modern C++ part1
14	Modern C++ part2
15	C++ development tools
16	Final exam

References

- Beginner's book:
  - Accelerated C++, Koenig et al.
  - C++ Primer Plus (6th edition), Stephen Prata

- For deeper understanding:
  - Effective C++ (3rd edition), Scott Meyers
  - More Effective C++, Scott Meyers
  - Effective STL, Scott Meyers
  - Effective Modern C++, Scott Meyers

Prerequisites

- Introduction to Software Design (소프트웨어 입문 설계)
  - Or the basic knowledge of C programming language such as control structure, functions, types, etc

- If you do not know C, please reconsider taking this course seriously.

Grading

Midterm exam	30%
Final exam	30%
Assignments	30%
Attendance	10%

- You will get **F** unless you take at least **9 lectures && 18 labs**
- If you do not take mid-term or final exam, you will get **F** regardless of your total scores

Assignments

- 3~5 programming assignments per each lab session

- TA will help you to solve the problems.
  - You can ask questions!

- Lab1(Wed) assignment due: 23:59 **on the day**.
- Lab2(Fri) assignment due: 23:59 **on next Tue**.

Policy for Assignments

- **NO SCORE** for late submissions
  - Submit before the deadline!

- **NEGATIVE SCORE** for cheating
  - If A copies B's code, A and B will get -100 points.
    - If B's code is just stollen by A, B will get credits as normal
  - If A, B, C copies the same code from the internet, all of them will get -100 points.
  - Collaboration encouraged, **but assignments must be your own work.**

Policy for AI Assistants

- Any form of AI assistants are NOT allowed to use to solve your homework
  - You should not use AI assistants such as chatGPT, Gemini, Codellama, copilot, etc

- Why not AI assistants?
  - This course is too easy to use AI assistants
    - My personal experiment shows that ChatGPT can get almost A+
  - At some point, you will use AI assistants to solve more difficult problems, but at this time, you need to think and do programming by yourself

Development Environment

- Development environment for this course:
  - You will use Visual Studio Code as an IDE for C++
    - IDE standing for Integrated Development Environment is a program to help programmers easily write, run and debug code
  - Visual Studio Code supports Windows, MacOS, and Linux

- Which versions of C++ we use?
  - C++17 (but most features came from C++03)