

Programming Practice (CS 109)

Instructor: Otfried Cheong

Class time: Monday 16:00–17:15

Lab time: Friday 13:00–16:00

Course webpage: <http://otfried.org/courses/cs109>

Teaching Assistants:

안가빈, 하대근, 구재현, 김단우, 이민지, 이주희, 정재원.

Students will be distributed into three classrooms for the labs:

- room 3445 (class room #5): all foreign students, and all students whose student number ends with 8 or 9;
- room 2443 (class room #3): all Korean students whose student number ends with 0, 1, 2, or 3;
- room 2445 (class room #4): all Korean students whose student number ends with 4, 5, 6, or 7.

If you don't speak Korean, you must go to class room #5.

We have a lab once a week:

- Friday, 13:00–16:00

You implement a programming task during the lab.

If you cannot finish the task during the lab, you need to program outside the lab (you can use the TA office hours).

(Programmer productivity varies by a large factor.)

Lab attendance is **required**. You need to formally excuse yourself if you have to miss a lab.

If you are done early, work on a bonus problem. **You cannot just leave early.**

There are **no other** programming homeworks.

Sometimes you are expected to prepare for each lab in self-study using the tutorials and example code on the course website.

We will announce the material to be studied for each week. For example, until next week (March 9), you need to study the sections **The command line** and **Introduction to Kotlin**.

It is important that you **do not just read through** the tutorials briefly.

Turn on your computer, and follow the steps of the tutorial on your own computer. That way the material will become active knowledge that you can apply yourself when you program.

Classum

You must regularly check the announcements on Classum (www.classum.org, see webpage).

We will use Classum for answering all your questions about the course contents. You can ask questions anonymously. You can ask questions in English or Korean.

Lecture attendance

We will take attendance in every lecture. You have two missed lectures free—use this for doctor appointments, interviews, etc. You do not need to send me email about missing a lecture.

Grading Policy

Project (20%), Midterm exam (30%), Final exam (40%), Lecture attendance (10%).

- dynamic and static typing,
- dynamic memory allocation,
- objects and methods,
- binary representation of numbers,
- using an editor and compiler from the command line,
- running programs with arguments from the command line,
- graphical user interfaces and event-based programming,
- using basic data structures such as arrays, lists, stacks, sets, and maps,
- writing a tiny Android app.

Class purpose: Give confidence in programming to students with little programming experience.

I imagine students who have taken CS101 or learnt a little programming in high school, and who have not yet taken CS major courses (but perhaps take them at the same time).

If you already program well, this course may not be so useful (and you would still have to spend the same amount of time on it).

Many of you know some Python, Java, or C.

In my experience, somebody's first programming language shapes their programming style.

It is difficult to write beautiful, elegant code in C.

Java is a boring, verbose, ugly language.

C++ is a powerful language, but has a steep learning curve, and it is very easy to make mistakes that are hard to find.

We will use **Kotlin**, a modern and clean language that encourages good style and concise, elegant code.

You will learn concepts and techniques that are not possible/easy in C, C++, or Java.

Even experienced programmers will have to work in this course.

Of course, many (most?) of you will still have to learn C, C++, and perhaps Java. Some of you may even have to use C in this semester.

The only problem with Kotlin is that it is very new and not commonly taught to beginning programmers, so there are no great teaching materials for beginners.

The CS109 course page has a Kotlin tutorial that should contain everything you need in this course.

The official Kotlin on-line documentation is not written for beginners. Don't despair if you do not understand it—it is not because you are not smart enough!

If you don't understand—ask! (the TA or on Classum)