

# IM 1003: Programming Design, Spring 2017

Instructor: Ling-Chieh Kung  
Department of Information Management  
National Taiwan University

In this course, we will introduce how to write computer programs for general purposes. The programming language we will study is C++, one of the most popular and powerful high-level programming language nowadays. We will start from the procedural programming part of C++, which is quite similar to the programming language C, and then discuss those object-oriented features of C++. While we will spend a lot of time on how to write “correct” programs, we will also try to write “good” programs, i.e., those run faster, using less memory, having better formats, generating friendly user interfaces, being more extendable, etc. The language C++ is just something that facilitates the delivery of the principles of computer programming. What really matter are the conceptual principles, not the C++ syntax or rules. Our objective is not to teach you how to write C++ programs; we want to make you be able to teach yourself any other programming languages. C++ is chosen to be taught in this course mainly because, in my opinion, it is a “broad” language. This will be discussed in lectures.

This is a required course for first-year undergraduate students in the Department of Information Management in National Taiwan University. We do not assume any background in computer programming, and there is no prerequisite for this course. However, some experiences in courses like Introduction to Computer Science and Discrete Mathematics helps. In most cases, all students who want to enroll in or audit this course are welcome.

**Note.** *Though this course counts for three units officially, students are suggested to treat it as at least five units and put efforts accordingly. In my opinion, this course should be (and indeed is) heavier than most introductory programming courses in NTU. If you do not have any programming experience, you do need to spend enough time on programming to really learn something. Studying programming is definitely not easy. Please do not hesitate to let me know if you need any help.*

## Basic information

### Instructor.

- Ling-Chieh Kung (孔令傑). E-mail: lckung(AT)ntu.edu.tw.
- Office: Room 413, Management Building 2. Tel: 02-3366-1176;
- Office hour: by appointment.
- <http://www.im.ntu.edu.tw/~lckung/>.

### Teaching Assistants.

- Jeff Lee (李維哲). E-mail: r04725023(AT)ntu.edu.tw.
- Jack Lin (林敬傑). E-mail: r05725007(AT)ntu.edu.tw.
- Sophie Yang (楊佩蓉). E-mail: r05725028(AT)ntu.edu.tw.
- Jhih-Bang Hsieh (謝志邦). E-mail: b02705021(AT)ntu.edu.tw.

### Meetings.

- Lectures: 2:20-5:20 pm, Monday. Room 305, Management Building 2.
- Labs: 6:25-8:10 pm, Wednesday. The large computer classroom, Management Building 1.

**Textbook.** *C++ How to Program: Late Objects Version* by Deitel and Deitel, seventh edition, Pearson Education. 臺灣代理: 歐亞圖書, (02) 8912-1188.

## References.

- *A First Book of C++* by Bronson.
- *C++ Primer* by Lippman, Lajoie, and Moo.
- *The C++ Programming Language* by Stroustrup.

## On-line Resources.

- To check grades: CEIBA.
- To download materials: <http://www.im.ntu.edu.tw/~lckung/courses/PD17>.
- To submit homework: PDOGS, <https://pdogs.ntu.im>.
- To discuss: Piazza, <https://piazza.com/ntu.edu.tw/spring2017/im1003>.

## Grading

### Breakdown.

- Homework: 30%.
- Midterm project: 15%. Final project: 20%.
- Three lab exams: 35% (15% each for the best two and 5% for the worst one).
- (Bonus) class participation: 5%.

**Conversion rule.** The final letter grades will be given according to the following conversion rule:

Letter	Range	Letter	Range	Letter	Range	Letter	Range	Letter	Range
F	[0, 60)	C-	[60, 63)	C	[63, 67)	C+	[67, 70)	B-	[70, 73)
B	[73, 77)	B+	[77, 80)	A-	[80, 85)	A	[85, 90)	A+	[90, 100]

**Regrading.** The TAs will grade everything except the final project and regrade them upon request. If you have a regrading request, please contact the TAs directly.

## Tentative schedule

Week	Date	Lecture	Textbook	Note
1	2/20	Introduction	Chs. 1 & 2	
2	2/27	Control statements*	Chs. 3 & 4	No lecture on 2/27
3	3/6	Arrays	Ch. 6	
4	3/13	Functions	Ch. 5	
5	3/20	Algorithms and recursion	Chs. 5 & 19	
6	3/27	<b>Midterm exam 1</b>	N/A	
7	4/3	Pointers*	Ch. 7	No lecture & lab on 4/3 & 4/4
8	4/10	C strings	Ch. 22	
9	4/17	Self-defined data types	Ch. 22	
10	4/24	Classes	Chs. 9 & 10	
11	5/1	Operator overloading	Chs. 10 & 11	
12	5/8	<b>Midterm exam 2</b>	N/A	
13	5/15	File I/O and C++ strings	Chs. 8 & 18	
14	5/22	Inheritance and polymorphism	Chs. 12 & 13	
15	5/29	Template and exception handling*	Chs. 14 & 16	No class on 5/29
16	6/5	<b>Final exam</b>	N/A	
17	6/12	Data structures	Ch. 20	
18	6/19	<b>Final project presentations</b>	N/A	

\* Lectures will be given in online videos

## Policies

**Homework.** For most weeks, one homework will be assigned on Monday or Tuesday and due in one week. Please upload your C++ source codes (and other files, if required) to the online grading system PDOGS by the due time. No submission in class or in lab. No hard copy. No late submission. While discussions are highly encouraged, each student must turn in her/his own homework. Cheating will result in severe penalty for everyone involved. The lowest two homework grades will be dropped (i.e., you may skip two homework if you want).

**PDOGS.** For homework of this course, we rely heavily on the Programming Design Online Grading System (PDOGS, or P-Dogs). After a student uploads her/his C++ source file, the system will automatically compile and run the program with respect to some testing data, calculate grades, and display the grades to the student. One may repeatedly modify his program and upload again and again until she/he is satisfied. Only the last grades will be recorded.

**Labs.** Students are strongly encouraged to attend labs. In lectures, the instructor will convey the concepts and ideas of C++ programming; in labs, the TAs may review materials covered in lectures, discuss past homework, and give students on-site practices. These practices do not count for any grade. However, attending labs is as useful as (if not more useful than) attending lectures.

**Office hour.** You are welcome to the instructor's office hour to ask him any question. You may ask him to clarify some concepts, give hints for homework problems, or discuss the final project. In fact, discussions not related to course materials are also welcome. However, because one must have enough painful experiences in debugging and revising programs, the instructor (and TAs) may refuse to debug for any student. If you want to schedule a meeting, please feel free to send the instructor an e-mail at any time.

**Attendance and class participation.** We do not count attendance. If you have something more important to do, feel free to drop a lecture or a lab. Nevertheless, we encourage class participation and include it in evaluating each student. During lecture time or office hour, you are more than welcome to ask or answer questions and provide comments. You are also encouraged to use Piazza for after-class discussions.

**Midterm and final projects.** Students should form teams to do one midterm and one final projects. One's teammates may be different for the two projects. For the midterm project, a task will be specified by the instructor for all teams to work on. For the final project, the instructor will only specify a rough direction. Each team then decides its own topic, build a program for its own objective, and demonstrate its program to the class publicly.

**Exams.** For three Mondays we will have exams during the lecture time. Students will be asked to write several C++ programs in three hours. One needs to prepare her/his own laptop (or ask for the instructing team's help in advance). The Internet will remain active throughout the exams, and one is allowed to search whatever she/he wants online. However, no communication with any living person is allowed. Cheating will result in a severe penalty for everyone involved.