

# IM 1003 – Programming Design, Spring 2014

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 running 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 five units and put efforts accordingly. 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, so please do not hesitate to let me know if you need any help.*

## Basic information

- |                            |   |
|----------------------------|---|
| <b>Instructor</b>          | <ul style="list-style-type: none"><li>• Ling-Chieh Kung (孔令傑).</li><li>• E-mail: lckung(AT)ntu.edu.tw.</li><li>• Office: Room 413, Management Building II. Tel: 02-3366-1176.</li><li>• Office hour: 9:10am-11:10am, Tuesday.</li><li>• <a href="http://www.im.ntu.edu.tw/~lckung/">http://www.im.ntu.edu.tw/~lckung/</a></li></ul>   |
| <b>Teaching Assistants</b> | <ul style="list-style-type: none"><li>• George Lee (李孟修). E-mail: r01725007(AT)ntu.edu.tw.</li><li>• Michael Hsu (徐承志). E-mail: r02725013(AT)ntu.edu.tw.</li><li>• Christine Hsieh (謝佳吟). E-mail: r02725040(AT)ntu.edu.tw.</li></ul>  |
| <b>Meetings</b>            | <ul style="list-style-type: none"><li>• Lectures (<b><i>in English</i></b>):<ul style="list-style-type: none"><li>○ 2:20-5:20pm, Monday.</li><li>○ Room 305, Management Building II.</li></ul></li><li>• Labs (<b><i>in Chinese</i></b>):<ul style="list-style-type: none"><li>○ 6:30-8:15pm, Wednesday.</li><li>○ The large computer classroom, Management Building I.</li></ul></li></ul>             |
| <b>Textbook</b>            | <ul style="list-style-type: none"><li>• C++ <i>How to Program: Late Objects Version</i> by Paul Deitel and Harvey Deitel, Seventh edition, Pearson Education. 臺灣代理: 歐亞圖書, (02) 8912-1188.</li></ul>   |
| <b>References</b>          | <ul style="list-style-type: none"><li>• <i>A First Book of C++</i> by G. Bronson.</li><li>• <i>C++ Primer</i> by S. B. Lippman, J. Lajoie, and B. E. Moo.</li><li>• <i>The C++ Programming Language</i> by B. Stroustrup.</li></ul>   |
| <b>On-line Resources</b>   | <ul style="list-style-type: none"><li>• To check grades: CEIBA.</li><li>• To download materials: <a href="http://www.im.ntu.edu.tw/~lckung/courses/PDSp14/">http://www.im.ntu.edu.tw/~lckung/courses/PDSp14/</a></li><li>• To submit homework: <a href="http://lckung.im.ntu.edu.tw/PD/">http://lckung.im.ntu.edu.tw/PD/</a>.</li><li>• To discuss: the bulletin board "NTUIM-lckung" on PTT.</li></ul> |

## Modules of this course

This course is divided into three modules: procedural programming (six weeks), object-oriented programming (six weeks), and advanced topics (four weeks). In the first module, we will tell students how to write programs with selections, repetitions, arrays, and functions to complete tasks or solve problems. In the second module, we focus on how to utilize classes, objects, and some other advanced techniques to enhance the quality of our programs. The last module discusses some advanced topics.

## Grading

- Breakdown**
- Homework: 30%.
  - Project: 15%.
  - Three lab exams: 40% (15% each for the best two and 10% for the worst one).
  - One written exam: 15%.
  - (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]

## Policies

- 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 will refuse to debug for you. If you do not want to come in the designated time, feel free to send me an e-mail to schedule a meeting.

- Labs**
- To enhance the learning efficiency, 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, demonstrate programming skills, and give students on-site practices. These practices do not count for any grade. However, attending labs is as important as attending lectures.
  - As most students taking this course speak Chinese as their native language, the TAs will hold labs *in Chinese*. This is why we make labs optional rather than required. All the lab materials will be prepared in English so that one may practice by herself/himself without understanding Chinese. However, if you do not understand Chinese and you have no background in computer programming, you may need to spend really a lot of time to get a good grade in this course. For those who do not understand Chinese, they may contact the instructor or TAs to have private discussions in English.

- Study Groups and Tutors**
- Beginners always need to put a lot of efforts in writing programs. At the first stage, advices from other people may help a lot. As it is impossible to have five instructors and ten TAs for this course, we would like to invite some students to become "tutors" and lead study groups.
  - A study group can include up to eight students, with one of them being the tutor. When a group member encounters some questions, she/he is encouraged to ask the tutor for help. Group members are also encouraged to study together or discuss homework problems. One student can belong to at most one group.
  - The tutor gets bonus points in the written exam. In the exam, each student will be asked to "send a reward" to her/his tutor (if there is one) by secretly sending up to five reward points to the tutor. The grades of the tutor will be the original grades plus the sum of all reward points with 120 as the cap.

By setting up this mechanism, tutors get rewards by helping group members and the amount of rewards is linked to how helpful she/he is.

**Class  
Participation**

- We do not require one to attend all the lectures and labs. If you have something more important to do, feel free to drop a class. Nevertheless, as communication skills are essential for almost everyone, we encourage class participation and include it in evaluating each student. In other words, class participation is not just sitting in the classroom. 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 the course bulletin board on PTT. These will not only give you a good grade but also significantly help your learning.

**Homework**

- Weekly homework will be assigned every Monday or Tuesday and due in one week (unless there is an exam). Please upload your C++ source codes (and other files, if required) to the course website on the due date. 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).
- 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.
- The TAs will grade and regrade your homework upon request. If you have a regrading request, please contact the TAs directly (e.g., in the labs).

**Project**

- Please form a team with at most three students to solve a task specified by the instructor. Each team needs to submit its C++ course codes and the document for their program. Each team will also need to do a demonstration of their program to the instructor or TAs. All team members must show up for the demonstration.

**Lab exams**

- For three Wednesdays we will have lab exams during the lab time. Students will be asked to write several C++ programs in 100 minutes. Students will be assigned seats randomly. One may decide to use the computer at the seat or her/his own laptop. 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.

**Written  
exam**

- We will have a written exam during the lecture time. It will be in-class and open-book. Electronic devices and discussions are disallowed. Cheating will result in a severe penalty.

## Tentative schedule

Week	Date	Lecture	Textbook	Note
Module 1: Procedural Programming				
1	2/17	Overview and the first example	Chs. 1 and 2	
2	2/24	Control statements	Chs. 3 and 4	
3	3/3	Functions	Ch. 5	
4	3/10	Arrays	Ch. 6	IM Night on 3/10: See you there!
5	3/17	Pointers	Ch. 7	
6	3/24	File input/output	Ch. 8	<u><i>Lab exam 1</i></u> on 3/26
Module 2: Object-oriented Programming				
7	3/31	Self-defined data types	Ch. 22	<u><i>No lab</i></u> on 4/2: spring recess
8	4/7	Classes (1)	Ch. 9	
9	4/14	Classes (2)	Ch. 10	
10	4/21	Operator overloading	Ch. 11	
11	4/28	Inheritance	Ch. 12	
12	5/5	Polymorphism	Ch. 13	
13	5/12	<u><i>Midterm exam</i></u>		<u><i>Lab exam 2</i></u> on 5/14
Module 3: Advanced Topics				
14	5/19	Templates and exception handling	Chs. 14 and 16	
15	5/26	Strings	Ch. 18	
16	6/2	Searching and sorting	Ch. 19	
17	6/9	Data structures	Ch. 20	
18	6/16	<u><i>Final project demonstrations</i></u>		<u><i>Lab exam 3</i></u> on 6/18