National Central University, Department of Civil Engineering

EG1001B; PROGRAMMING FOR ENGINEERING (Fall 2015)

Unique Number: **93002**

Instructor: **Dr. Chung, Chih-Chung (鐘志忠)**
Room: E.263 **Phone: (03) 422-7151 ext.34120**
E-mail: ccchung@ncu.edu.tw

Class Meetings: **3:00pm – 5:00pm, Mondays at Room I.002**
9:00am – 10:00am, Thursdays at Room E.315

Office Hours: Wednesdays 8-10am, or by appointment

Teaching Assistant:

Name	E-mail

Reference Materials:

Required: 子由. (2011) *深度學習C++*, 3rd Edition, 藍海文化, ISBN: 9789866432477.
http://www.ziyou.math.ncu.edu.tw/~ziyou/c++/

Recommended: Bruce Eckel. (2000) *Thinking in C++, Volume One*, 2nd Edition, Prentice Hall, ISBN: 0139798099. (可自網路免費下載)

Bruce Eckel and Chuck Allison. (2003) *Thinking in C++, Volume 2*, 1st Edition, Prentice Hall, ISBN: 0130353132. (可自網路免費下載)

Bjarne Stroustrup. (2013) *The C++ Programming Language*, 4th Edition, Addison- Wesley, ISBN: 0321563840.

Prerequisites:

None.

Course Description:

This course teaches fundamental computer concepts and programming skills for engineering applications. Emphasis is focused on object-oriented programming. Assignments cover programming concepts, user interfaces, numerical methods, data structures, algorithms, and selected advanced topics. The C++ programming language is used.

Programming languages are one of the key tools extensively utilized by engineers to design or orchestrate computers. This course will use a particular programming language, C++, to introduce you to how to use a computer to solve problems. In this course, most of your learning will take place when you are putting in to practice the concepts from the book and lectures to solve more and more complex problems using more and more sophisticated programming practices. You will learn through doing. Most of your grade will therefore be based on developing your programming skills.

Course Objectives:

Upon successful completion of the course, students will be able to comprehend the following topics:

- To understand the nature of programming as human activity
- To learn and experience main components of programming process
- To understand main control structures of programming languages
- To learn and being able to use major programming patterns
- To learn how to make a reusable program library

Upon satisfactory completion of this course, students will be able to write small but meaningful C++ programs that includes major control structures (such as loop, selection), programming patterns (such as sequential array processing), and data types. On the way to that goal we will have to learn a reasonable subset of C++ language, explore many working examples and write multiple C++ programs.

Important note: this course is targeted to the students who have no or very little programming knowledge and experience. The goal of this course is to get you familiar with programming concepts and constructs at a relatively slow pace. Students with some good programming background may find this pace boring and are strongly advised to help other students. ***Extra credits will be given for the students who can provide assistance during in-class exercise sessions.*** If you have any doubt, check the course schedule below to see what exactly will be taught.

Grading:

Grade components will be weighted as follows in the computation of the final course grade:

	Points
Homework/Programming Competition	30
In-Class Exercise	30
Mid-Term Exam	20
Final Exam	20
TOTAL	100

Final Grades:

Final grades will be determined by ranking the students' final scores, which are calculated with the weighting scale above. Grade distribution will be determined based upon the instructor's experience of acceptable performance in the course.

Homework/Programming Competition:

Homework assignments provide insight and practice in sharpening computer programming skills. Each assignment must be submitted electronically using LMS at any time before or on the due date. Basically, your submissions are time stamped, and those turned in late will count off 10% per day. Remember that extensions delay the posting of solutions so be considerate of your classmates when asking for extensions. My goal is to return all homework assignments to students within three course-weeks from the date submitted.

In-Class Exercise:

Programming exercises will be administered during each class session. It is important that you are familiar with the course materials as the course evolves. Your ability to answer questions and discuss the materials will be part of the overall evaluation. Therefore, you should review class materials ahead of time. Regular attendance is expected and encouraged. Your attendance will be used to evaluate your in-class exercise grade. I consider a student missing more than one week of class lectures without excuse to be a serious participation problem. Each student is responsible for all materials and administrative instructions given during the lecture period. If you have illness or personal problems that will affect your performance during the course of the semester, please let me know as soon as possible. "After the fact" provides little protection unless there are extreme circumstances. I have an e-mail address if you need to get in touch with me after hours. Do not hesitate to use them.

Cooperation and Cheating:

You may discuss homework assignments or in-class exercises with others, but you must write up by yourself with the full understanding of what you write. Students handing in identical solutions will be violating university regulations and will not receive credit. Exchanging homework or in-class exercise solutions is cheating and will be reported to the university. At minimum, you will fail this course.

Mid-Term and Final Exams:

These are open-book exams. A student is responsible for all reading assignments and class handouts whether or not covered in class or listed on the syllabus. The exams will include materials covered in reading assignments and class discussions. Exam make-ups will be given only in the event of a verified emergency or doctor- verified sickness.

C++ Compiler Source Software (Free)

1. **Dev-C++**
   (http://sourceforge.net/projects/orwelldevcpp/files/Setup%20Releases/)
   (http://www.bloodshed.net/devcpp.html)

2. **Microsoft Visual Studio**

Course Schedule

Date	***Week***	***Topic (Lab / Exercise Topic)***	***Assignments/Projects***
9/14, 17	1	Introduction (Survey of Students' Programming Background) (IDE, Integrated Development Environment)	Assignment 1
9/21, 24	2	Introduction to Programming	
9/28, 10/1	3	Data Types, Variables, and Operators	
10/5, 8	4	Data Types, Variables, and Operators Iterated Loops I	
10/12, 15	5	Iterated Loops II	Assignment 2
10/19, 22	6	Iterated Loops III	Assignment 3
10/26, 29	7	Pointer	
11/2, 4	8	Array	Assignment 4
11/9, 12	9	11/9 3-6pm: Mid-term Exam 11/13: Array	Mid-term Exam
11/16,19	10	String	Assignment 5
11/23, 26	11	Function I	
11/30, 12/3	12	Function II	Assignment 6
12/7, 10	13	Class, Object, Instance, and Static I	
12/14, 17	14	Class, Object, Instance, and Static II	
12/21, 24	15	Input/output	Assignment 7
12/28, 31	16	Encapsulation, Inheritance and Polymorphism	
1/4, 7	17	Numerical method	Assignment 8
1/11	18	3-5pm Final Exam	Final Exam / Programming Competition

Note: The course schedule is subject to changes. Any changes in the course schedule will be communicated in advance and posted in the course LMS web page.