

# 計算機程式設計 (II)

## Computer Programming II

- **Instructor**

- Ming-Feng Tsai (蔡銘峰)
- Email: [mftsai@cs.nccu.edu.tw](mailto:mftsai@cs.nccu.edu.tw)
- Office: 大仁樓 413 室
- Office Hours: Wed. 1-2pm or by arrangement

- **Time**

- Lecture: Wed. 2-4 sessions (9:10 – 12:00)
- Lab: Wed. C session (12:10 – 13:00)

- **Place**

- Lecture & Lab: 大仁 200301

- **Teaching Assistants**

- 吳武峰
- 簡傑
- 洪駿宸
- 陳奕玄

- **Goal**

This course is designed to provide a substantial exposure to the C programming language in a Unix environment for students with basic programming experience. Advanced features of the C language that are emphasized include structs and unions, File I/O, dynamic memory allocation (malloc and free), pointers, pointer arithmetic, and casting. Data structures that are emphasized include linked lists, queue, and stack. Students will develop a sense of proper programming style in C, and will be exposed to cross-platform portability issues. Students will also learn to use several Unix tools such as gdb, emacs/vi, svn/git, make, grep, diff etc. to assist them in the design, testing and debugging of their programs. In addition, students will learn some basic shell scripting, and Perl/Python to solve simple problems.

- **Description**

The course will involve lectures by instructor, lab exercises, and programming assignments on various topics. Students are expected to participate the exercises, and write the assignments on their own. There will be a midterm conducted as a written test, and a final exam including written and programming tests. The content of the course includes the following topics:

- Part I – Advanced Features of C
  - Pointer and Arrays
  - C Preprocessor
  - Bitwise Operations
  - File Input/Output
  - Debugging and Optimization
  - Advanced Pointers
  - Modular Programming
  - Portability Problem
- Part II – Advanced Unix Tools
  - Unix Basics: essential commands, pipes, redirections
  - Effective Programming: emacs/vi, svn/git, make, gdb
  - Useful Tools: grep, sed, diff, xargs, find, ...
  - Scripting: Bash, Perl/Python
- Part III: Simple Data Structure and Algorithms
  - Queue, Stack, Linked List
  - Searching, Sorting

- **Enrollment:** Freshmen of CS department

- **Prerequisite:** Basic C programming knowledge

- **Textbooks (Optional)**

- *The C programming Language*, K&R.
- *Advanced C*, Peter D. Hipson.
- *C Unleashed*, Richard Heathfield et al.
- *Unix Unbounded a beginning approach*, AMIR Afzal.

- **Grading**

- Midterm: 30%
- Final Exam: 35%
- Labs & Assignments: 35%
- Bonus (participation): < 5%

- **Assignments:** There would be 6 to 8 assignments.
- **Assignment Submission**
  - **Printouts:** a hard copy of your code
  - **Electronic submission:** Make sure your codes can be compiled and executed in 'ghost', because it is the platform that we grade your submissions.
- **Late Policy**
  - Score for late submission is calculated as follows:  
**original points - n \* 20 points** (note: n = # of delay days)
  - For example, if an assignment were delayed by 3 days and its original score were 80, then the score of the assignment would be:  $80 - 3 * 20 = 20$  points.
- **Grade Appeal**
  - Student has two weeks (from the date handed back) to request a re-grade or appeal the grade recorded in the GradeBook. A re-grade will be performed on the entire Lab/Assignment/Exam and can lower the score!!
  - In any appeal procedure, it's the student's responsibility to keep possession of his/her Lab/Assignment/Exam. In the process of a re-grade, a student has to arrange for a TA to modify the grade in the presence of the student. A student should not hand over any material to the TA for keeping.
  - A lost or missing Lab/Assignment/Exam is no reason for a modification of a grade.
- **Cheat Policy**
  - SEVER PENALTIES ARE APPLIED FOR THE FOLLOWINGS
    - actively sharing (or copying) all or parts of someone else's code/answers on Assignment/Exam
    - passively allowing the sharing (or copying) of your own codes/answers on Assignment/Exam
  - What is Cheating?
    - On Assignments: allowing others to view your code or reading some else's source code constitutes cheating. Students should protect their own work so that another student cannot copy any part of their code.

Therefore, if copying has been detected, it will be assumed that cheating has occurred by all parties involved.

- On Exams: allowing others to view your answers or reading someone else's answer constitutes cheating.

- Penalties

- A zero score for the assignment and a deduction of 20 points from the student's course point total.
- A second occurrence of cheating will mean an automatic grade of ZERO in the course and the notification of a Dean in student's college.

- **Schedule (Tentative)**

Week	Topics
1	Course Introduction
2	Unix Basics (I)
3	Unix Basics (II)
4	C Revisited; Pointer, Array, and Strings
5	Advanced Pointers (1): malloc, calloc, realloc, and free
6	Advanced Pointers (2): Dynamic Allocation
7	Module Programming
8	Basic Data Structure (1): Linked List
9	<b>Midterm</b>
10	Basic Data Structure (2): Stack and Queue
11	Basic Data Structure (3): Tree and Graph
12	Basic Algorithm (1): Big O notation, Algorithm Analysis
13	Basic Algorithm (2): Searching and Sorting
14	Problem Solving (1)
15	Problem Solving (2)
16	Introduction to Shell Scripting (1)
17	Introduction to Shell Scripting (2)
18	<b>Final Exams</b>