The University of New South Wales
School of Computer Science and Engineering

COMP9021
Principles of Programming

Session 2
July 19, 2007

Contents

- Course Aims
- Lecturers and Contact Information
- Assessment
- Organisation & Schedule
- Teaching Strategies
- Course Resources
- Course Evaluation and Feedback
- Plagiarism

Course Aims

Units of credit: 6

No parallel teaching: only COMP9021 students attend the classes.

This is a group A course. It has no prerequisite. Like all group A courses, it consists of bridging material in computing taught at an accelerated pace. It is a core course for students enrolled in the Graduate Diploma in Computing and Information Technology or in the Masters of Computing and Information Technology. It is also available to students enrolled in the Graduate Certificate in Computing. Students who have already covered the material presented in this course can get exemption if they pass the exemption exam taking place on Friday of week 1.

- The aim of the course is to help you master:
  - the design, implementation and testing of programs written in a procedural language;
  - the language C, and the underlying model of memory;
  - some Unix utilities, with an introduction to shell scripts for the bash shell;
  - some fundamental data structures and algorithms.
- By the end of the course you will be able to:
  - design programs to solve small to medium scale problems;
  - write clear, reliable, well-structured, well-tested, well-documented programs in C;
  - make use of appropriate Unix utilities for problems that require data manipulation;
  - represent data with linked lists, stacks, queues, and binary trees;
  - use searching and sorting algorithms;
  - implement hash tables.

You know that at university, the focus is on your self-directed search for knowl edge. Lectures, labs, textbook and recommended reading, assignments and exams are all provided as a service to assist you in this endeavour. It is your choice as to how much work you do in this course, whether it be preparation for classes, completion of assignments, study for exams or seeking assistance or extra work to extend and clarify your understanding. You must choose the approach that best suits your learning style and goals in this course. Still note that the University expects you to do about 150--180 hours work for this course---including lectures, labs, and time spent on assignments. Of course this will vary according to your aims. The cours e is designed in such a way that passing the course will only require a good understanding of the fundamental notions as well as good practical skills, thanks to regular work. If your aim is to obtain a high distinction then you will need to invest more time in this course.

Lecturer and Contact Information

The lecturer-in-charge for this course is Rex Kwok.

Role	Name	Room	E-Mail	Phone
LIC	Rex Kwok	K17-401	rkwok@cse.unsw.edu.au	x56909

Students with academic questions and/or problems relating to this course should contact the lecturer-in-charge. Substantial problems are best handled in consultations. If you need a consultation please email the lecturer in charge for an appointment.

Students are also advised to check the course's web page frequently:

http://www.cse.unsw.edu.au/~cs9021

as many of the common problems you're experiencing may have already been resolved.

Assessment

The assessable components of the course are:

Component	Mark
Assignment 1	15%
Assignment 2	15%
Assignment 3	15%
Practical final exam (2 hours)	25%
Written final exam (2 hours)	30%

To pass the course, you will need to

- get a total mark of 50 at least;
- get a mark of 12 at least for the practical final exam;
- get a mark of 15 at least for the written final exam;

The three assignments will be programming assignments.

- The first assignment will focus on standard IO, control structures, and the use of array.
- The second assignment will focus on command line arguments and the use of pointers.
- The third assignment will focus on stream and file manipulation, and the use of structures.

The assignments give you the chance to practice what you have learnt and design solutions to common, small to medium scale problems. The learning benefits will be greater if you start working on the assignments early enough; do not leave your assignments until the last minute.

The assignment specifications will provide instructions on how and when to submit. Assignments submitted late are subject to the following penalty. The maximum mark obtainable reduces by 10% per day late, up to a maximum of 50%. Thus if students *A* and *B* hand in assignments worth 13 and 10, both two days late, then the maximum mark obtainable is 12, so *A* gets *min*(13,12)=12 and *B* gets *min*(10,12)=10.

Note that if you want help with your programs, you should make an appointment for consultation and present an up-to-data listing that is reasonably well laid out and documented. You must also bring evidence of having taken reasonable steps to solve the problem yourself. Do not send an email to the lecturer in charge of the form: *My program is attached. How does it come it does not work?*

The document "Important Advice for Students" states the supplementary assessment policy for the School of CSE. The following criteria will be used when considering a student for a supplementary exam:

- has the student submitted a fully documented request for Special Consideration within 7 days of the exam;
- was there medical or other evidence to support the claim that the student's performance was seriously affected? Minor ailments like a cold; a headache--unless chronic, or severe enough to require prescribed medication; feeling "unwell", *etc* do not excuse a poor result;
- has the student's performance during the semester, particularly in any mid-session test been of satisfactory standard? Students who fail such a test will not be granted further assessment. Assignment marks may not carry much weight in determining whether further assessment is warranted; and
- the student did not attend the exam, or attended the exam but was forced to leave through illness.

The document can be found on-line at:

http://www.cse.unsw.edu.au/~studentoffice/policies/yellowform.html

Please take the time to read it carefully.

If you are granted a Supplementary examination, then it is most likely to be held in November/December. If you think that you may be eligible for a supplementary exam, then make sure you are available around that time. It is your responsibility to check at the School Office for details of Supplementary examinations.

Organisation & Schedule

The course will include one 3-hour lecture per week during session. A tentative lecture schedule for the session is as follows:

Lecture 1: Course introduction, an introduction to C and UNIX

Lecture 2: Data types, formatted IO, operators

Lecture 3: Control flow, arrays

Lecture 4: Testing, functions

Lecture 5: Complex data types: structures and unions

Lecture 6: Pointers I, string functions

Lecture 7: Pointers II, storage classes, modules and linkage

Lecture 8: UNIX and shell scripts

Lecture 9: Stream and file manipulation

Lecture 10: Recursion

Lecture 11: Linked lists and stacks

Lecture 12: Queues and binary search trees

Lecture 13: Sorting and hash tables

Lecture 14: Course summary and exam revision

Teaching Strategies

The teaching of COMP9021 will include a mix of lectures, labs, and face-to-face consultations.

Lectures

The lecture time is:

Day	Time	Location
Wednesday	18:00-21:00	Mechanical Enginneering 304 (K-J17-304)

All lecture slides will be made available via the class home page. If you do not attend lectures, then it is your responsibility to discover any announcements and/or supplemental information that's given.

The purpose of lectures is to introduce you to the concepts covered by this course, show where they fit in the overall scheme of things, provide motivating examples to help you understand them, and demonstrate the skills and processes that you should be acquiring during this course. Note that lectures, due to time constraints, can only cover the material at a relatively superficial level. You cannot hope to completely understand the material simply by listening in lectures. You should also:

- read up on the topics covered in each lecture;
- go thorugh the exercises in your textbook; and
- explore the topic on-line by writing and running your own applications.

Laboratory Access

A lab will be run from week 2 to week 4, in order to give you a good start and resolve potential technical issues. You can enroll in one of three labs:

- Monday 16:00 - 17:30, Bugle Lab, K17 ground floor
- Wednesday 16:00 - 17:30, Bugle Lab, K17 ground floor
- Thursday 16:00 - 17:30, Bugle Lab, K17 ground floor

Practical work can be conducted either on the School's lab computers or on your own computer. If your computer is a Windows machine then you need to install `gcc` and the Unix emulator `cygwin`. For that purpose, you should look into purchasing a copy of the *Home Computing CD*. The CD contains a collection of programs and utilities and is available from the CSE Help Desk; for more information look at:

http://www.cse.unsw.edu.au/~homecomputing

When you have to make electronic submissions that will be auto-marked, you have to test your solutions on the School machines. Passing a test means passing a test on the School machines on the submission date. Never submit an assignment before checking on the School machines that it functions as expected. This generally requires that you make a booking in the CSE labs. Note that bookings are especially heavy around assignment due dates.

The labs will help you get used to the Unix environment and the setup of the computing laboratory. To learn more about

it you should browse the \emph{Unix primer} at:

http://www.cse.unsw.edu.au/help/doc/index.html

You might find the following web sites useful.

- CSE Help Desk: http://www.cse.unsw.edu.au/~helpdesk
- UNSW library: http://info.library.unsw.edu.au
- UNSW Learning center: http://www.lc.unsw.edu.au
- UNSW Communications unit: http://www.comms.unsw.edu.au
- Computer ergonomics: http://do.cse.unsw.edu.au/ergonomics/ergoadjust.html
- Occupational Health and Safety policies and expectations: http://www.riskman.unsw.edu.au/ohs/ohs.shtml
- Equity and Diversity issues: http://www.equity.unsw.edu.au/disabil.html

Consultation

As stated above, students with academic questions and/or problems relating to this course should contact the lecturer-in-charge. Substantial problems are best handled in consultations. If you need a consultation please email the lecturer in charge for an appointment.

Course Resource List

Announcements, lecture notes, example programs, lab exercises and solutions, and assignment specifications are made available at the course's web page:

http://www.cse.unsw.edu.au/~cs9021

The textbook for this course is:

> Paul Davies. *The Indispensable Guide to C* (2nd edition). Addison-Wesley.

You are expected to read and work from the textbook as a complement to the other material that is made available. The lecture notes should clarify the textbook's most difficult points. An alternative textbook is K&R:

> Brian Kernighan and Dennis Ritchie. *The C Programming Language* (2nd edition). Prentice Hall.

Though K&R is one of the best references on C, it might be too dense for newcomers to programming.

The following reference books are listed as other sources you might wish to consult. You are encouraged to read them or part of them. The lecture notes will borrow some material from these books that is not to be found in the textbook.

- Stephen Prata. *C Primer Plus*} (4th edition). Sams Publishing.
- Stephen Kochan. *Programming in C* (3rd edition). Sams Publishing.
- Stephen Kochan and Patrick Wood. *Unix Shell Programming* (3rd edition). Sams Publishing.

Course Evaluation and Development

Student feedback on this course, and on the lecturing in this course, will be obtained via electronic survey at the end of session. Student feedback is taken seriously, and continual improvements are made to the course based in part on this feedback.

Students are very strongly encouraged to let the lecturer in charge know of any problems, as soon as they emerge. Suggestions (or even complaints!) will be listened to very openly, positively, constructively and thankfully, and every action will be taken to fix any issue or improve the students' learning experience.

Plagiarism

All work submitted for assessment **must be your own work**. Assignments **must be completed individually**. We regard copying of assignments, in whole or part, as a very serious offence. **We may use plagiarism detection software to search for multiply-submitted work.** Be warned that:

- the submission of work derived from another person, or jointly written with someone else will, at the very least, result in automatic failure for COMP9021 with a mark of zero.
- allowing another student to copy from you will, at the very least, result in a mark of zero for your own assignment; and
- severe or second offences will result in automatic failure, exclusion from the University, and possibly other academic discipline.

Collaborative work in the form of "think tanking" is encouraged, but students are not allowed to derive solutions together as a group during such discussions. Students are also warned not to send solution fragments of the assignments to each other in any form (e.g. as email or listings). In addition, copying/purchasing of solutions that is available on the web is also not permitted. Students who are singled out during our regular plagiarism sweep will be dealt with according to School Policy:

http://www.cse.unsw.edu.au/~studentoffice/policies/yellowform.html.
(see also http://www.cse.unsw.edu.au/help/doc/primer/node42.html)

Where required, follow-up interviews will be conducted by a panel to decide on an appropriate course of action (if any). The decisions of the panel will be final.

Students are strongly advised to protect their work. Do not leave your terminal/computer unattended, or leave listings at the printer for others to take.

Eric Martin and Rex Kwok
19-07-2007