COMP1911/COMP1091/COMP1021 07s1 Course Introduction

Course Web Site

http://www.cse.unsw.edu.au/~cs1911

Aims of this Course

The objective of this course is for students to develop proficiency in programming in a high level imperative language and to develop a background of relevant knowledge and skills on which to base further study of computing.

Topics covered include:

- Fundamental programming concepts
- Program testing and debugging
- The underlying memory representation of data
- Unix literacy
- Version control
- Programming style
- Professional issues

Practical experience of these topics is supplied by laboratory programming exercises and assignments.

This course introduces the fundamentals for programming. It is followed by COMP1921 (Design and Algorithms), which in turn, provides the basis for COMP2911 (Design 2). In combination, the three courses form the introductory core of the Computing Programs.

After completing this course, you should be able to design, construct, document and test C programs that accurately comply with a given specification. In addition, you should also have an understanding and use of common data structures to solve problems. Your understanding should also include knowledge of how programs translate from C into assembler; and how such programs are executed.

This course is the first part of the introductory core for the computing programs. It covers fundamental programming concepts, and provides the prerequisites for COMP1921 (Design and Algorithms). COMP2911 (Design 2), in turn, builds on this material, introduces new programming concepts and discusses the design of complex systems.

Prerequisites

There are no pre-requisites for this course.

Teaching Methodology

Our teaching objectives for this course are to have students doing continuous independent research and learning in their own time throughout the entire session. This is not a course where you can do little or no work throughout session and then cram for the final exam. You will need to be active in your learning each week.

There is weekly homework consisting of tutorial preparation, practical lab exercises, and general programming practise. You are expected to spend at least 3 hours per week on this homework. Any spare time you get you should spend practising. Learning programming craftsmanship involves much the same process as learning the piano.

Topics are introduced in lectures. You will need to do further work to master the topics.

Reading Materials

The recommended text for this course is:

- Alistair Moffat, Programming, Problem Solving, and Abstraction with C, Pearson Educational, Australia, 2003, ISBN 1-74103-080-3.

The following books might also serve as additional reference material:

- Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, 2nd edition, Prentice Hall, USA, 1988, ISBN 0-13-110370-9.
- Jeri R. Hanly and Elliot B. Koffman, Problem Solving and Program Design in C, 4th edition, Addison Wesley, 2004, ISBN 0-321-21055-7.
- H.M. Deitel, P.J. Deitel, C How To Program, 4th edition, Prentice Hall, New Jersey, 2004, ISBN 0-13-122543-X

Staff

Name	Role	Extension	Consultation

Gabriele Keller	Lecturer-In-Charge	9385-7131	Wednesdays, 10am - 12am, by appointment (you're welcome to drop by without appointment during this time, but if you want to make sure I'm actually in my the office, send me an email)
Richard Buckland	Lecturer (extended)	9385-TODO	TBA
Gabriel Ditu	Lecturer	9385-5539	TBA
Angela Finlayson	Course Admin	9385-6852	Thursday, 1pm, Location: K17-G01.

Lectures

These are the lecture times and locations:

Time	COMP1911/COMP1021 Location	COMP1091 Location
Tuesday, 2-4pm	Keith Burrows	CLB 4
Wednesday, 4-5pm	Keith Burrows	CLB 2
Wednesday, 5-6pm (optional)	Keith Burrows	Keith Burrows

During the two hours of the Tuesday lecture and the first hour of the Wednesday lecture, we will cover the core content of this course, and attendance is compulsory. The slides will be available on the web page, but please note that they'll only reflect part of what is covered in each lecture. It is essential that you take your own detailed notes.

Attendance of the 5pm to 6pm Wednesday lecture is optional. We will discuss more advanced topics and look into some interesting problems. If there is demand, we may offer a remedial lecture instead. Remedial lectures will be announced in the Tuesday lecture, and also on the web page.

Tutorials

Tutorials give you a chance to clarify ideas mentioned in lectures and to practice your problem-solving skills in a small class with the assistance of a tutor. They are followed by a two hour lab session (see below). To get the most out of the tutorials, you should take time to examine the material to be covered in each week's tutorial in advance. Read through your lecture notes, and make sure you discuss your open questions with the tutor.

The tutorial questions will be posted on the Web in the week before each tute. Tutorials are one hour long and they are scheduled to start in Week 2 of session.

Generally, there will be extra questions supplied, which are not covered during the tutorial. You may use these for practice or revision.

Your tutor will also ask you to give a brief presentation of a specific topic at the beginning of one of the tutorials (usually in groups of two).

Laboratory Classes

Laboratory Classes give you a chance to practice programming skills on small, simple examples. The examples have been chosen to highlight particular aspects of programming, and will give you enough grounding in programming practice to enable you to complete your project work relatively easily. Your tutor will be present in your Lab Class to answer any questions you may have.

All lab class exercises must be shown to your tutor during your scheduled lab class. The lab exercises will be available during the session via the course's Web page. You advised not to fall behind in your laboratory work. In order to get a mark for a lab exercise you must show your tutor a completed or partially completed solution before the end of your scheduled lab class in the week that the exercise is scheduled. If you cannot complete the exercise by the end of the lab you may complete it in your own time and submit it using the `give` command by midnight on the following Sunday. Your tutor will then mark it at the very beginning of the next scheduled lab class.

For Example: You could submit files for lab 2 with the command
```
give cs1911 lab02 lab02_1.c lab02_2.c
```

Note: To gain a mark you must at the very least: show your tutor a partial solution; and, submit a completed solution using give by midnight the following Sunday. After that time, you will not receive a mark for that particular lab exercise. If you are unable to attend your scheduled lab class due to illness or misadventure, then you should apply for special consideration and an extension may be granted.

Note: You can only have your lab work marked during your scheduled lab class. It is not possible to have your work marked in another class.

To make sure that you can complete the work in your scheduled lab class, you should make an effort to prepare the work beforehand. You can make use of the Lab simply by booking a time-slot. You have sufficient bookable Lab time available each week to allow you to prepare your lab exercises; make good use of it.

Your weekly lab class is 2 hours long and it starts in Week 2 of session. Where possible, your lab class will be scheduled to follow on from your weekly tutorial and will have the same tutor (but you will need to walk to a new room).

Assessment

The assessable components of the course are:

Component	Mark
Tutorial	Presentation 1%
Lab Exercises	9%
Assignment 1	10%
Assignment 2	20%
Written Exam	60%

All submitted programs (lab exercises and assignments) have to adhere to the Course Style Guide.

Assignments

There will be two programming assignments. Assignments give you the chance to practice what you have learnt on relatively large problems (compared to the small exercises in the Lab Classes). Assignments are a very important part of this course, therefore it is essential that you attempt them yourself.

Assignment have to be submitted using the `give` command. The specification of the each assignment will contain the submission details. If you submit the assignment after the deadline, you will loose 10% of the maximum marks for each day you are late, and 0 marks if you submit more than five days late.

Important:

- Students with a mark less than 10/40 for the practical component (assignments, lab exercises and tutorial presentation) will not be allowed to attempt the final written exam.

Supplementary Exams

The Yellow Form states the supplementary assessment policy for the School of CSE. Please take the time to read it carefully. If you are granted a Supplementary Examination, then it is most likely to be held in 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.

A supplementary examination will be held soon after the results have been released. If you think that you may be eligible for the Supplementary Examination, make sure you are available around that time. Be careful not to plan any overseas travel at that time. If you can't attend the supplementary exam you will not be offered a second chance.

It is your responsibility to check your email, the CSE website, and to contact the CSE school office for details of Supplementary Examinations. If you think there is any chance you might be eligible for a Supplementary Exam then you should prepare for it.

Plagiarism

All work submitted for assessment must be your own work. Lab exercises and assignments must be completed individually. We regard copying of assignments or lab exercises, in whole or part, as a very serious offence. We use plagiarism detection software to search for suspiciously similar submissions. Collaborative work in the form of "think tanking" is encouraged, but you are not allowed to derive code together as a group during such discussions. Also, do not to send code fragments of the assignments or labs to each other in any form (e.g. as email or listings). In addition, copying/purchasing of code 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, which provides serious penalties particularly in the case of repeat offenses:

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

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

Policies and Rules

- No abuse or flaming in the forum
- No language feature may be used in assessable work unless it has been officially covered in the course. If you are unsure ask.
- All coding must comply with the course Style Guide.

The university has a number of rules and policies which affect you (see myUnsw, the university handbook, and the university home page), additionally the school of CSE has a number of rules and policies (eg see the "Yellow form"), and this course has a number of rules and policies (see this page). Where there is a conflict between these sets of rules and policies the most strict shall apply. Where there is ambiguity in the interpretation of any rule of policy the most strict

interpretation shall apply. If you are in any doubt as to the meaning, interpretation, or effect of any rule or policy please ask the course administrator or the lecturer in charge. Some of the policies which apply to you can be accessed from the course home page. Be aware that this list may not be definitive or up-to-date.

Getting Help

>From time to time, various problems may arise in your study of this course. Below is a list of typical problems, with suggestions for where you might seek help with them:

Problem	**Solution**
Help with problems unrelated to this course	CSE Web Help Page
Can't understand lecture material	Ask the lecturer during or after class, or your tutor during your tute/lab, post question on message board
Stuck with assignments, lab work	Ask your tutor during your tute/lab
Problems with lab workstations	CSE Help Desk
Problems with your account	CSE Help Desk
Problem with Dial-Up access from home	DIS-Connect Helpdesk (9385 1777)

The School Web site has useful information that may be of help:

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

Particularly useful is the Unix man pages. The searchable interface allows you to find out more about various Unix commands (this assumes that you have some basic knowledge of the Unix environment):

http://www.cse.unsw.edu.au/scripts/man

If you are new to the computing laboratory and its setup, customs and environment is read up about it in the Unix Primer. You can find the document on this page: http://www.cse.unsw.edu.au/help/doc/index.html

Besides the Unix Primer, you'll also find other useful documents there. Take the time to explore the system. The more adventurous you are, the more you will learn about the system and the more proficient you will get---a handy trait to have when it comes to doing your assignments as well as your final Practical Exam. Those interested in working from home should look into purchasing a copy of our Home Computing CD, which is available from the CSG Help Desk. It contains a collection of programs and utilities used by various courses within the School. To find out more, go to: http://www.cse.unsw.edu.au/~homecomputing

One final piece of advice, if you have a problem, do not wait until Week 14 before you seek help. Problems that are dealt with early are usually quicker and easier to resolve than one that is left to the last minute.

Course Feedback and Improvement

At the end of the session, we will ask you for feedback on this course via electronic survey. The information we collect this way will be used to make continual improvements to the course. However, if you have suggestions or like to point out problems, please don't wait until the end of session!! Contact us (lecturer or tutor) and we will make every reasonable effort to address the issues.

Gabriele Keller
Last modified: Thu Feb 22 23:49:50 EST 2007