COMP10002 Foundations of Algorithms

Semester Two, 2017

Welcome!

Introduce yourself to your neighbors while you are waiting

Summary

Staff in 2017s2

Lecturer: Alistair Moffat.

Tutors: Alicia Yang, Anh Vo, Bridget Loughhead, Daniel Porteous, Gareth Seddon, Malcolm Karutz, Robert Langtry, Samuel Yang-Zhao, Tobias Edwards, Unni Krishnan.

Contact information is on the LMS page.

Getting started

The single most important thing you have to do to get the semester off to a good start is to make sure you have friends in the class.

So, every time you enter a room over the next two weeks, sit down next to someone you don't know, and introduce yourself.

Critical information

Critical information #1: Everything is on the LMS, including all handouts.

Lecture recordings will appear on the LMS page shortly after each class.

You should look at the LMS page every couple of days right through the whole semester.

Critical information

Critical information #2: Workshops will commence Week Two. There are no workshops this week.

Critical information

Critical information #3: The textbook will be used extensively, including in Workshops.

Programming, Problem Solving, and Abstraction with C by Alistair Moffat (second edition, Pearson, 2012), on sale at the Co-op Bookshop for approximately $76. An e-edition is available from the publisher's website for $53, and more info at `http://people.eng.unimelb.edu.au/ammoffat/ppsaa`.

You are advised to have your own copy.

Having at least one other C book at your disposal will probably be helpful.

Subject overview

Foundations of Algorithms provides further programming, now using the language C, with an emphasis on fundamental techniques and algorithms, and on software development skills.

Particular topics that will be covered include dynamic data structures, and the algorithms that manipulate them (lists, trees, hash tables); searching algorithms including pattern searching; and sorting algorithms.

We go “under the hood”, and build technical understanding.

Programming environment

You can use any C programming environment that you have access to. The MSE labs support two different mechanisms, one based on `jEdit` and command-line compilation, and one based on the `Eclipse` integrated development environment.

The former provides a "bare metal" understanding of programming.

Both approaches are free and can be installed on home computers and laptops.

Programming is “hands on”...

The emphasis is on you doing programming, and learning the necessary skills in a hands on manner.

You need to work steadily through the semester, and write (and execute) programs throughout. You will also need to develop your knowledge of both programming techniques, and of the processes that lead to the development and analysis of algorithms.

Programming is like driving a car, you need lots of actual practice to become good at it.

Classes and attendance

There are three lectures each week, plus a two-hour workshop.

Worksops will consist of approximately one hour of “tutorial”-style interaction, plus one hour of supervised programming work. Two tutors will be present during the second hour to provide help and support.

You should stay up to date with all scheduled classes. If you choose not to attend lectures, be sure to watch them online prior to your next workshop.

Computer accounts

You may use any of MSE computer labs in Alice Hoy and Old Engineering when they are unbooked (but note that scheduled classes cover almost the entire week).

Your standard University account name and password will allow access. Login during your first workshop to initialize your account, *even if you plan to do your programming on your own computer.*

Your University email address (something like `jsmith@student.unimelb.edu.au`) should be directed to a location at which you will see any emails we send.

Total workload

Three lectures, and a two-hour workshop.

Plus:

- One review hour for each hour of lectures, including reading the text
- Two preparation hours for the workshop.
- Two hours of general review/reading, perhaps in a study group.

In total, around 12 hours per week per subject is required, starting immediately.

Make a study timetable for all activities.

Then start following it.

Total workload

If you have outside interests (including work) that consume more than approximately 12–15 hours per week, you are seriously jeopardizing your chances of passing.

If your outside interests cannot be restricted to fewer than 12 hours per week, you should consider taking only three subjects per semester.

Seeking assistance

There are a range of mechanisms to use when you need help.

- Check the LMS for general announcements.
- Search the LMS discussion for the same question.
- Post your query to the LMS discussion forum. Read other posts and responses while you wait for a response to your query.
- Ask a question after a lecture (or at the start of a lecture if the answer will be of wide interest).
- Ask in your workshop.

Assessment

Your final mark is the combination of three components.

Task	Due	Marks
Mid-semester test	25 Aug, tbc	10%
Assignment 1	18 Sep	15%
Assignment 2	16 Oct	15%
Examination		60%

To pass the subject as a whole you must also attain at least 28/70 (combined) in the test and exam, and 12/30 (combined) in the two projects.

Assessment

The test will take place on (probably) August 25 in the usual class time. You should use the test as early feedback of your status in this subject.

If you do well, that's great.

If you do poorly, heed the signal it sends.

A sample test, and more details of the format, will be supplied closer to the time.

Academic honesty

All assessed work in this subject is individual.

We routinely run sophisticated similarity checking software over all submissions. If you are clever enough to outsmart this software, you are also clever enough to do your own project.

The University's Academic Integrity policy will be applied if duplicate work is detected. Penalties go as far as subject failure, or even termination of enrolment.

Academic honesty

There are also rules governing misuse of the various computer systems.

Misuse includes unauthorized storage of copyright material (software as well as digital data like music); unauthorized access to other accounts; and any other activity not associated with your study.

Choose a sensible password, and keep it secure.

Week one checklist

Things to be done:

- Check that you can access the LMS page.
- Get hold of the textbook, Programming, Problem Solving, and Abstraction with C. Start reading it.
- (By Friday) Confirm your workshop time, and check the LMS for any late messages about workshop locations.
- Most importantly, make some new friends, have some fun, and get set for a great semester.

Algorithms are Fun!