comp20005 Engineering Computation

Semester One, 2021

Welcome!

Lecture slides prepared by Alistair Moffat

Staff in 2021

Coordinator and lecturer: Alistair Moffat.

Contact information is on the handout and on the LMS page.

Plus tutors who will run the workshops, listed on the LMS.

Students in 2021

Mostly future engineers, plus some mathematics, physics, biology, commerce students, etc. Plus some (current, and if I do my job, future) CS students.

I'll be working to convince you that "programming is fun". *But no-one should think that they will be a "programmer" after taking just this one subject; comp20005 is just the beginning!*

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 please make a real effort in your workshops to build a network. Turn cameras on, and be willing to chat to others and swap contact details. They need you in exactly the same way as you need them.

Subject overview

Engineering Computation provides an introduction to programming using the language C, and builds on your computing experiences with MATLAB in ESD2.

You may make use of any C programming environment. The Engineering labs support two different mechanisms: use of a web browser to access the `grok` on-line service; and one based on `jEdit` and `gcc` command-line compilation.

`Grok` can also be accessed from outside the University; and `jEdit` and `gcc` are both free downloads. Instructions are provided on the LMS.

Subject overview

The emphasis is on you doing programming, and learning the necessary skills in a hands on manner. We will be making use of numeric problems, and, where possible, engineering examples.

You need to work steadily through the semester, and write (and execute) programs throughout.

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

Teaching mode

Lecture material will be delivered via pre-recorded videos, delivered via the Echo360 portal in the LMS. There will be approximately three hours of video to watch per week.

Workshops are being offered in two alternative modes:

- On-line mode, using zoom
- In-person mode, in a computer laboratory on campus.

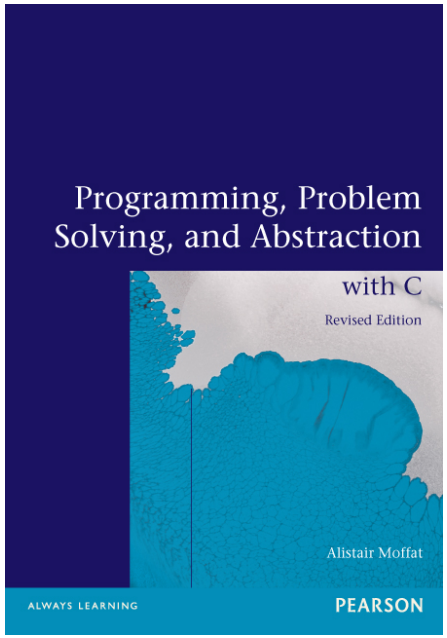
You are free to choose which you prefer. If restrictions become necessary again, in-person workshops will also be held on-line.

All assessment will be carried out fully on-line. There will not be any in-person assessment requirements.

Textbook

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 http://booktopia.com.au for approximately $75. An e-edition is available from the publisher's website for $50, with more info at http://people.eng.unimelb.edu.au/ammoffat/ppsaa.



(Yes, I receive royalties on the modest number of new sales each year; no, I don't get rich out of it; first I pay the tax, and then the rest gets donated to charity.)

Total workload

Three hours of lecture videos, and a two-hour workshop.

Plus:

- Further review of lecture content, including reading the text and re-listening to (parts of) the lecture.
- Two preparation hours for the workshop.
- Two hours of general review/reading, perhaps in a study group.

In total, around 10–12 hours per week per subject is required, starting immediately; 150–180 hours over the semester.

Make a study timetable for all activities. Then follow 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.

When you need assistance...

Lecturer consultation will be available via zoom at most of the "lecture times" right through the semester:

- Monday, 5:15pm–6:15pm
- Wednesday, 4:15pm–5:15pm
- Friday, 1:00pm–2:00pm.

I'll be sitting there in zoom, waiting and hoping that students arrive that I can chat to. (I like talking to students!)

When you need assistance...

- Check the LMS for general announcements.
- Check the LMS discussion forum for existing answers.
- Post a question to the LMS discussion forum.
- Attend a lecturer consultation.
- Ask your tutor during your workshop.
- Connect with a tutor via the on-line discussion in `grok`.
- For personal matters, send an email requesting a zoom appointment.

Assessment

Your final mark is the combination of six components.

Week	Task			
5	Quiz 1	March 31	4:15pm	10%
8	Assignment 1	April 30	6:00pm	20%
9	Quiz 2	May 7	1:00pm	10%
11	Assignment 2	May 21	6:00pm	20%
12	Quiz 3	May 28	1:00pm	10%
–	Exam			30%

To pass the subject as a whole you must attain at least 24/60 (combined) in the quizzes and exam; plus 16/40 (combined) in the two projects; plus 50/100 overall.

Academic integrity

All assessment in this subject is individual, and at the time you submit each item you will be required to certify that it is your own unassisted work.

We run sophisticated similarity checking software over all submissions. It looks at the innate program structure, not just syntactic presentation. Mark deductions will be applied if duplicate or inexplicably similar work is detected.

Where warranted, the University's Academic Integrity policy will be applied. In the past this has led to students receiving zero for the assignment or subject, regardless of other assessment components.

Week one checklist

Things to be done:

- Check that you can access the LMS page.
- Get hold of a copy of the textbook, Programming, Problem Solving, and Abstraction with C. Read Chapters 1 and 2.
- (By Thursday) Confirm your workshop time, and check the LMS for any late messages about workshop locations. If your workshop is on Friday, it will take place this week.
- Most importantly, mentally prepare to make some new friends, and get set for a fun semester of programming.

When something goes wrong. . .

There are bound to be things that go wrong; and we can't fix things we don't know about.

So please send email if you can see an issue brewing.

Hardware faults are notified using the IT-Help page.

Next step: Let's do some programming

The next video will get you started on programming in C. You can use these links to access the two programs shown in the video:

- `helloworld.c`
- `addnumbers.c`

Don't hesitate, go straight on!