Australian National University

COMP1730/COMP6730
Programming for Scientists

Introduction and administrative matters

Acknowledgment of the country

Australian National University

Conveners

Dr. Alberto F. Martin

Computational Science, High Performance Computing

Dr. Minh Bui

Bioinformatics

https://comp.anu.edu.au/courses/comp1730/people/

Australian National University

Senior tutors

Dr. Alexei Khorev

Hancheng Shao

Malcolm Macdonald

https://comp.anu.edu.au/courses/comp1730/people/

Australian National University

Other Tutors

* Carina Cai
* Chathura Nagoda Gamage
* Dilmi Jayasena
* Gaurang Garg
* Han Zhang
* Jamie Whittington
* Jon Connor
* Kanav Thareja
* Madhawa Perera
* Dr. Matthew Macaulay
* Muhammad Salman
* Richa Awasthy
* Robert McArthur
* Ruiqi Li
* Sam O Brien
* Sandy Zhao
* Shashank Gummuluru
* Vimukthini Pinto
* Xiaodi Zhang
* Yingnan Shi
* Zongyu Fan

Announcements

* Read announcements on Wattle news forum, **importantly the weekly information**
* Complete “week 1 checklist” on course website
* Lectures: Mondays 10-11am and Tuesday 11am-noon. Ask questions live on MS Teams, joining with code **sd01s2e**.
* Labs: from week 2. Installation troubleshooting in week 1:
  - Tue 2-4pm, CSIT N114
  - Wed 2-6pm, CSIT N112
  - Thu noon-2pm, CSIT N113
  - Fri 2-4pm, CSIT N114

Australian National University

Lecture outline

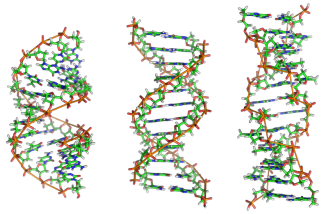
* Why programming for scientists?
* Course overview.
* Info, contacts & schedule.
* Assessment scheme.
* Academic integrity.
* Student representatives.

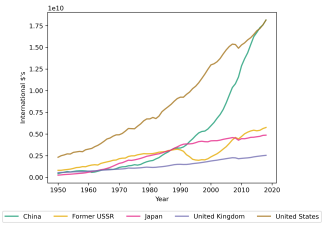
Australian National University

Why programming for scientists?

“Science rests on data, processing data needs software.”

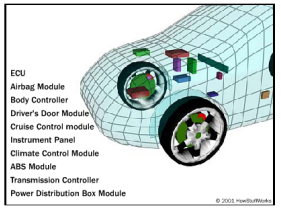
* **Biology:** use DNA data to understand evolution of Life on Earth and track COVID-19 virus variants. http://www.iqtree.org
* **Economics:** Modelling GDP growth over time and across countries. https://quantecon.org (Prof. John Stachurski, ANU)



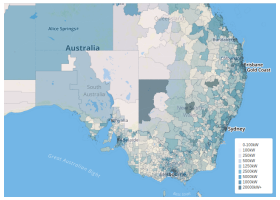


Australian National University

* Technical systems increasingly run on software.
  - **Engineering:** Software on a modern car has $>$ 100M lines of code.



* Simulation and optimisation are needed to solve large-scale design challenges.
  - Intermittent renewables produced ∼35.9% of Australia’s electricity in 2022. How do we design the grid to work with 100%?



- *As scientist or engineer, you need to understand how software works, and perhaps extend it*:
  - understand algorithms and implementation to interpret and explain their results;
  - debug programs (find and correct errors);
  - modify existing programs to solve your (unique) problem.
- By the end of the course, we hope you'll tackle a novel problem by saying, "Hey, I can just write a program to solve that..."

Australian National University

Programming example

- you want to calculate the monthly repayment of a $500,000 home loan...
  - use one of the on-line mortgage calculators?
- ...for all loan terms in 20-30 years, and an interest rate of 2% to 6%.
- The formula is

$$A = P\frac{r(1+r)^n}{(1+r)^n - 1}$$

(derive it, or look it up on wikipedia).
Let's write a program!

Australian National University

```
# first way to calculate monthly repayment
P = 500000
r = 0.02
n = 30
A = P * r * (1+r)**n / ((1+r)**n - 1)
print(A/12)

# second way to calculate monthly repayment
P = 500000
r = 0.02/12
n = 30*12
A = P * r * (1+r)**n / ((1+r)**n - 1)
print(A)
```

Live quiz in lecture: which code is correct?

1. First code
2. Second code
3. Both
4. I don't know
5. I don't care

Australian National University

Quiz results: total 245 votes

1. First code: 82 votes
2. Second code: 58 votes
3. Both: 91 votes
4. I don't know: 7 votes
5. I don't care: 7 votes

And the right answer is: ***2. Second code!***

Why python?

* This is *not* a course on programming in python; it's a course on programming, that uses python.
* Python is nowadays the *most popular* programming language,
* particularly for science and engineering uses.
* Open source, available on most platforms.
* Many modules:
  - over 200 in the python standard library;
  - over 100,000 on `pypi.org`.
* We will use **python 3**.

Australian National University

Course description & aims

* Introduction to programming (using python).
  - No prior programming or computer science knowledge is required.
  - This does not mean it is easy!
* Two aims:
  - Programming as a practical skill.
  - Understand some basic CS concepts; build foundation for later courses.

Australian National University

Learning outcomes

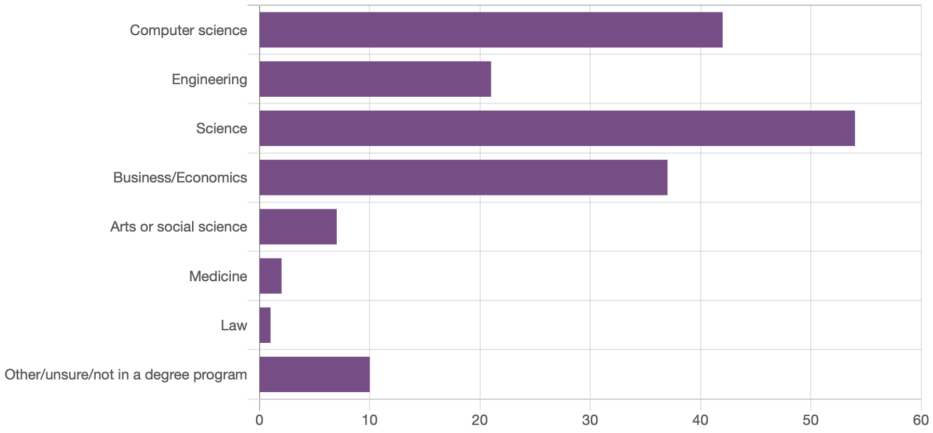
(revised from ANU Programs & Courses)

Students who succeed in all aspects of this course will:

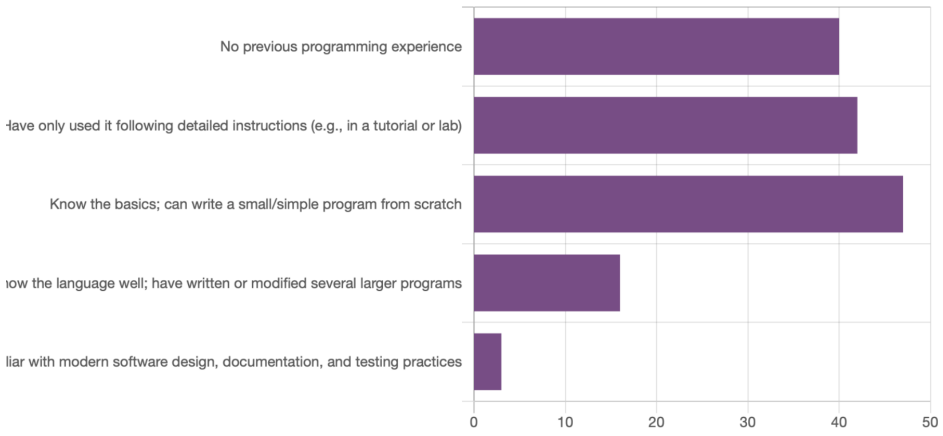
* be able to design and write readable and correct small programs to solve practical data processing problems;
* be able to read, understand and debug small computer programs;
* understand some practical limitations on computer programs, including scaling (wrt time and memory) and numeric precision (rounding errors) issues.

Australian National University

About you: students in the course

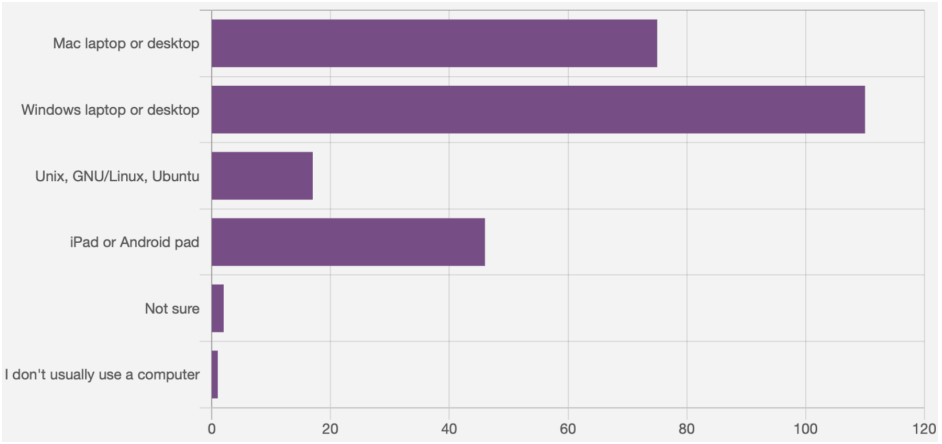


Prior programming knowledge



Australian National University

Platform



Australian National University

Course info & contacts

* https://comp.anu.edu.au/courses/comp1730/
* Wattle forums, quizzes, surveys, assignment submission. *Read Wattle news & announcements*!
* To ask a question:
  - MS Teams channel for live lecture questions.
  - Use the discussion forum on wattle.
  - For *personal* questions, email `comp1730.convener@anu.edu.au` (Always *use your ANU email*), or
  - come to my Office Hours: Mon 12-1pm and Tue 1-2pm.

Australian National University

Discussion forum – 3 simple rules

1. **Read before you post**.
   Before posting a question, check if your question has already been answered.
2. Give your post a **good, descriptive topic**.
   Don't write "A question". Write something like "Variable assignment: why does the value not change?".
3. You **may not post** solutions to assignment problems.

- This applies to any on-line forum.

Schedule overview

* https://timetabling.anu.edu.au/sws2023/
* Two lectures / week.
  - Recording on echo360, if unable to attend or for later use.
  - Follow content & schedule on the course web site, and read the news & announcements.
* One 2-hour lab / week (starting from week 2).
  - Select your lab on MyTimeTable until end of week 1!
* You are expected to spend another 6 hours to study the course (e.g., solving all lab exercises).

Australian National University

Assessment scheme

https://comp.anu.edu.au/courses/comp1730/assessments/

* 5 small homework assignments (15%)
* 1 larger project assignment (25%)
  - **Mandatory in-lab assessment!**
* Final exam (60%), no hurdle

S. Week	
2	Homework 1 due (Sunday)
3	Homework 2 due (Sunday)
5	Homework 3 due (Sunday)
	Break
Break	Homework 4 due (Sunday)
8	Homework 5 due (Sunday)
10	Project due (Sunday)
11	**In-lab project assessment!**
Exam period	Final exam(s)

Australian National University

* Final mark is the sum of assessment marks, with 50% required to pass.
* If you are close to the boundary of the higher grade, e.g., 48-49% or 78-79%, we will consider pushing it up based on your **lab participation**, such as interaction with tutor, submitting code to CodeBench, answering Wattle quizzes.
  - Records are kept in Lab attendance on Wattle.
* Note: "*any submitted work may be subject to an additional oral examination*", which can change the assessment mark in any way.
* All assignment **deadlines are hard** – late submissions without an approved extension will not be accepted.

Australian National University

Academic integrity

* Academic integrity is taken seriously at ANU! – Academic Integrity Rule 2021 is a legal document at the University.
  - Being uninformed of or misunderstanding the Rule is never an excuse for a breach of academic integrity.
  - Any student is expected to undertake the online **Academic Integrity modules on Wattle**.
* Discussing programming problems (e.g. from labs) and ways to solve them with other students is a great way to learn – just don't discuss assessment problems.
* All assignments are *individual*. You must write your own code, and be able to show that you understand every aspect of what you have written.

- The final exam will be in-person and *individual*. You may not discuss the exam questions or your answers with anyone.
- Any academic misconduct will leave a record on internal student file or even appear on your transcript in severe cases.
- If you are unsure, please ask your tutor or conveners.
- Examples of NOT OK:
  - *"The code I used in my assignment was pulled from StackOverflow but I didn't realise I had to reference an online post."* **(Plagiarism)**
  - *"I used several sources to solve this assignment. There's a mix of my ideas and parts of others. I thought it was considered mine."* **(Plagiarism)**
  - *"I discussed the **individual assignment** with a friend and acknowledged their contribution to my assignment."* **(Collusion)**

Australian National University

Responsibilities of Class Reps

- Act as the official liaison between your peers and convener
- Be available and proactive in gathering feedback from your classmates
- Attend regular meetings, and provide reports on course feedback to your course convener
- Close the feedback loop by reporting back to the class the outcomes of your meetings
- Interested? contact us comp1730.convener@anu.edu.au by end of week 2.

Australian National University

Student representatives

- Class Student Representation is an important component of the teaching and learning, quality assurance and quality improvement processes.
- Students can nominate themselves for one or more of the courses they are enrolled in.
- The role is to provide ongoing constructive feedback on behalf of students to course conveners and to Associate Director (Education) for improvements to the course.