COMP10001 Foundations of Computing
Welcome and Introduction

Semester 2, 2016
Chris Leckie

July 8, 2016

Who?

- **Lecturer:**
  - Chris Leckie (`caleckie@unimelb.edu.au`)
    - office: DMD 7.11 (level 7 of Doug McDonell Bldg)
    - office hours: TBA (from next week)

Who?

- **Lecturer:**
    - Chris Leckie (`caleckie@unimelb.edu.au`)
        - office: DMD 7.11 (level 7 of Doug McDonell Bldg)
        - office hours: TBA (from next week)

> ### ⓘ Definition
> Lecturer (*n*): person who writes/delivers the lectures, coordinates the subject, designs the projects, writes the tests/exams, informs, entertains, engages, enthuses and disentangles the undisentangleable

Who?

- **Tutors:**
  - Danny Minh Tuan Doan (head tutor)
  - Farah Zaib Khan
  - Qingyu Chen
  - Milad Chenaghlou
  - Zahra Ghafoori
  - Lucy Joel
  - Shing Sheung Daniel Ip
  - Aidan Turpin
  - Yoshua Wakeham

Who?

- **Tutors:**
    - Danny Minh Tuan Doan (head tutor)
    - Farah Zaib Khan
    - Qingyu Chen
    - Milad Chenaghlou
    - Zahra Ghafoori
    - Lucy Joel
    - Shing Sheung Daniel Ip
    - Aidan Turpin
    - Yoshua Wakeham

> ### 🛈 Definition
> Tutor (*n*): person who runs the workshops, helps with the marking, provides sagacious advice on subject-related matters, reinforms, empathises, explains, endures (the lecturers) and helps decipher the undecipherable

Who?

- **Demonstrators:**
  - Kevin Then
  - Chung Man Lam
  - Angie Hu
  - Damian Testa
  - Joanna Lee
  - Lachlan Dee
  - Ned Lechter
  - Caiti Wade

Who?

- **Demonstrators:**
    - Kevin Then
    - Chung Man Lam
    - Angie Hu
    - Damian Testa
    - Joanna Lee
    - Lachlan Dee
    - Ned Lechter
    - Caiti Wade

(i) Definition

Demonstrator (*n*): person who aids the flagging tutor in running the lab component of the workshops, possibly helps with the marking, provides sagacious advice on subject-related matters, rereinforms, empathises, explains, endures (the lecturers and tutors) and helps decipher the undecipherable

What (is the Subject all about)?

- Harnessing computation for problem solving
- Fundamental programming constructs
- Data manipulation
- Elements of maths, engineering, logic, design; dollops of creativity
- Concerned with theories, principles, limits of computation and information
- Requires logical reasoning, persistence in solving complex problems, and "technical creativity"

What (is the Subject NOT about)?

- Learning to use word processor or spreadsheet software
- Computer hardware

Let's Play ...

How do we solve everyday problems?

- Example: Word puzzle

Python Programming Language

- Easy to learn: interpreted language; interactive experimentation
- Free; open source (`python.org`)
- Highly readable
- Cross-platform compatible
- Powerful, extensive libraries
- Widely used in industry, science, education, entertainment …
- We will use Python v3.4 via Grok Learning

Where and When

- Lectures (three per week):

    Mon   2:15–3:15pm   (Kathleen Fitzpatrick Theatre, Arts W)
    Tue   3:15–4:15pm   (PLT, Old Arts)
    Wed   4:15–5:15pm   (PLT, Old Arts)

    Starting in week 2, Wednesday lectures will rotate between
    a guest lecture, a revision lecture,
    and an advanced lecture

- Workshops (one per week)

    - 2 hours
    - first part is a tutorial, second part is programming lab
    - start in Week 2 (NO WORKSHOPS THIS WEEK)

How do I Get Started?

- Check out the LMS:
  `http://www.lms.unimelb.edu.au`
- Log in to Grok Learning:
  `https://groklearning.com/login/`
- Lecture slides, lecture recordings and code snippets will be made available from the lectures/workshops page on the LMS
- Take a look over the schedule for the subject
- Check out the forums

Assessment I

- Your subject mark will be made up of:
  - Interactive Grok Learning worksheets: 10%
  - Projects ($\times$3): 30%
  - Mid-semester test: 10%
  - Final exam: 50%

- There will be rolling deadlines for Grok Learning worksheets, as listed on the LMS, with the cutoff for the set of worksheets released each week being 23:59 Monday of the next week (unless otherwise stated)

Assessment II

- There are two "hurdles" for the subject: you must achieve at least 50% for the projects/interactive worksheets AND at least 50% for the mid-semester test/final exam

  If you fail **either** component, you will fail the overall subject

  And of course you must get at least 50% overall

How do I Get Help?

- Post a question to the LMS forums
- Talk to the lecturer after the lecture
- Talk to your tutor/demonstrator
- Come along to the revision lectures
- If you are struggling with the subject, don't be shy about asking for help; similarly if you are experiencing documentable hardship and unable to meet submission deadlines, let us know **at the time**

Secret to Subject Success

- Balanced workload of 10–13 hours per week, e.g.:
  - Workshop (2 hours attendance + 2 hours follow-up)
  - Lectures (3 hours attendance + 3 hours review)
  - Study (2 hours review/reading/study group)
- Attend, listen, ask, and share, but above all do, do, do!

Tips from Uncle Chris I

- Lectures and workshops start 5 mins later and end 5 mins earlier than advertised

- All lectures will be recorded (audio and screen scrape) ... but try to come along to ask questions and get the full lecture experience

- Expect things to move faster and marks to be harder to get than in high school

- When learning programming, constant "practice" is the key to success

- If you email Chris, start the subject line with COMP10001, and I can take $\leq$ 48 hours to reply

Tips from Uncle Chris II

- Never share any **examinable** code with your fellow students (not on the forums, not via email, not via shared machines, …)

- If you need help, avail *yourself* of the various sources of assistance; don't expect us to come chasing you

- More detail on this in the next lecture, and also read carefully the **Academic Honesty** section on the **Subject Overview** page of the LMS

We Need You!

- We need 2 volunteers to act as "student representatives" for the subject, with the following responsibilities:
    - keep finger on pulse of student body
    - (possibly) act as go-between between students and teaching staff
    - attend a Staff–Student Liaison Committee meeting later in the semester to report on issues with the subject, and run a feedback session immediately beforehand to poll the student body

And another thing ...

- We will set up an "Online Learning Centre" this semester, in the form of a virtual drop-in help centre for computing-related matters
- The current plan is to have a run a virtual version in the evening (starting in week 2)

Things to do before the Next Lecture

- Make "comprade" friends
- **DON'T** go to a workshop **THIS WEEK**
- Check that you can access the subject LMS site
- Log in to Grok (you should be able to do this using via `https://groklearning.com/login/` using your student login id and password)
- Post to the subject forum (personal testimonial, computing-related material that caught your eye, ...)