COMP10001 Foundations of Computing
The Basics of Programming

Semester 1, 2019
Tim Baldwin, Nic Geard, Farah Khan, and Marion Zalk

— version: 1463, date: March 8, 2019 —

Lecture Agenda

- Last lecture:
  - Computers speak binary, but we don't
  - High level programming languages make life easier
  - We will use Python inside Grok
- This lecture:
  - Programming basics with blockly

Lecture Outline

1. Announcements
2. Programming

Announcements

- Moomba's in the air ... Labour Day public holiday on Monday ... many people will enjoy a public holiday ...

Announcements

- Moomba's in the air ... Labour Day public holiday on Monday ... many people will enjoy a public holiday ...

  but **not** universities — see you for workshops as usual on Monday (and mind public holiday timetable on public transport!)

Ode to UniMelb Labour Day

'Twas the morning of Moomba, when all thro' Melbourne
Not a creature was stirring, except for just one:
They were a student at MelbUni
Bound for their workshop, truly;
The trains were infrequent, a day that one dreads,
With visions of lazy morns dancing in their heads.

A REMINDER: How do I Get Help?

- Post a question to the forums on Grok
- Submit a help request via Grok
- Talk to your tutor/demonstrator during your workshop
- Talk to the lecturer after the lecture
- Come along to the revision lecture
- 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**

Lecture Outline

1. Announcements
2. Programming

Turtle Programming

- As an illustration of this, without getting bogged down in the details too much, let's play around with Turtle graphics, using the "blockly" programming language
- Basic commands:
  - advance forward/backward `N` units
  - turn left/right `N` degrees
  - `goto(X,Y)`: go to position `X,Y`
  - `circle(R)`: draw a circle of radius `R` units from current position

… And Emailing the Lecturer?

- If you email me asking a question that could equally have been asked on the forums, I will respond via the forum and **not** email … not because I want to be rude, but just to clarify boundaries of what is Tim-emailable and what is not, and because I can't deal with 1400+ students' worth of one-on-one email!
- Please don't be offended by this; understand the reasons behind it
- Note that I equally can't service random dropins at my office: grab me after a lecture, or get help from your tutor/demonstrator

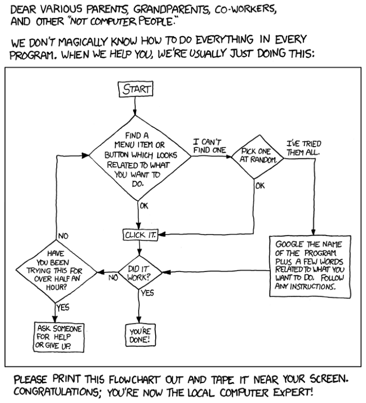
Programming

- Computer programs are simply sets of steps to complete some task
- Determining what the steps should be requires learning how computers "think"… and how a particular programming language expresses the way a computer thinks
- At its most basic level, a program is made up of a sequence of **statements** that are executed sequentially one after the other

Program Design

- Many modern computing languages express similar concepts
- They allow "conditioning" on particular values, "looping" over sub-sets of steps, and "nesting" of loops
- Common ways to abstractly represent programs are:
  - flowcharts
  - "pseudo-code" (i.e. a computer program in an abstract language, without the "bookkeeping" that individual languages require) `http://www.bestrecipes.com.au/recipe/choc-chip-cookies-L4351.html`

Example Flowchart

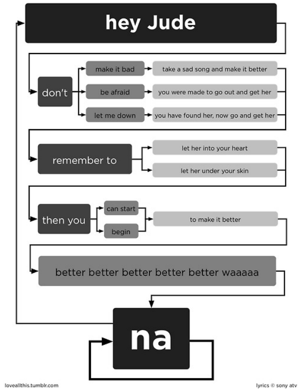

DEAR VARIOUS PARENTS, GRANDPARENTS, CO-WORKERS, AND OTHER "NOT COMPUTER PEOPLE."
WE DON'T MAGICALLY KNOW HOW TO DO EVERYTHING IN EVERY PROGRAM. WHEN WE HELP YOU, WE'RE USUALLY JUST DOING THIS:
START
FIND A MENU ITEM OR BUTTON WHICH LOOKS RELATED TO WHAT YOU WANT TO DO.
I CAN'T FIND ONE
PICK ONE AT RANDOM.
I'VE TRIED THEM ALL.
OK
OK
CLICK IT.
GOOGLE THE NAME OF THE PROGRAM PLUS A FEW WORDS RELATED TO WHAT YOU WANT TO DO. FOLLOW ANY INSTRUCTIONS.
NO
HAVE YOU BEEN TRYING THIS FOR OVER HALF AN HOUR?
NO
DID IT WORK?
YES
YES
ASK SOMEONE FOR HELP OR GIVE UP.
YOU'RE DONE!
PLEASE PRINT THIS FLOWCHART OUT AND TAPE IT NEAR YOUR SCREEN. CONGRATULATIONS; YOU'RE NOW THE LOCAL COMPUTER EXPERT!

Source(s): `https://xkcd.com/627/`

Equivalent Pseudocode

```
1:  repeat
2:      find a related menu item OR pick one at random you haven't tried
3:      if found one then
4:          click it
5:          if it worked then
6:              done!
7:          else if been going for > 30 mins then
8:              give up!
9:          end if
10:     else
11:         Google a solution
12:         go to 5
13:     end if
14: until done! OR give up!
```

More Interesting Flowchart



Source(s): `http://laughingsquid.com/hey-jude-flow-chart/`

Basic Programming Building Blocks

- The basic building blocks of programming are:
  - **statements** (= single "commands" to the computer)
  - **sequence** (= linear sequence of statements)
  - **control** (= perform sequence of statements IF condition holds)
  - **loops** (= repeat sequences of statements)
  - **functions** (= blocks of code that can be run with different inputs)
  - **recursion** (= blocks of code that call themselves with different inputs)

Class Exercise

- Using just `move` and `turn` statements, build blockly code to draw an equilateral triangle with side length 100

Looking Towards Next Week

- Commencement of Monday–Thursday workshops (work out when your workshop is)
- Make sure you can log in to Grok

Lecture Summary

- Building blocks of programming