THE UNIVERSITY OF WESTERN AUSTRALIA

Lecture 1
Computers and Recipes

Programming

- Why do we need it ?
  - *Projected to grow by 37% by 2026 in Australia [ACS Australia's Digital Pulse 2021]*
  - *Everything is getting digitized and we need to interact with computers*
  - *It is **problem solving** for the most part but using computers*
- What is computer like ?
  - *Happy to do whatever asked*
  - *Happy to do repetitive and boring tasks*
  - *Deaf-mute who understands 0's and 1's only*
  - *Having IQ of zero ☺*

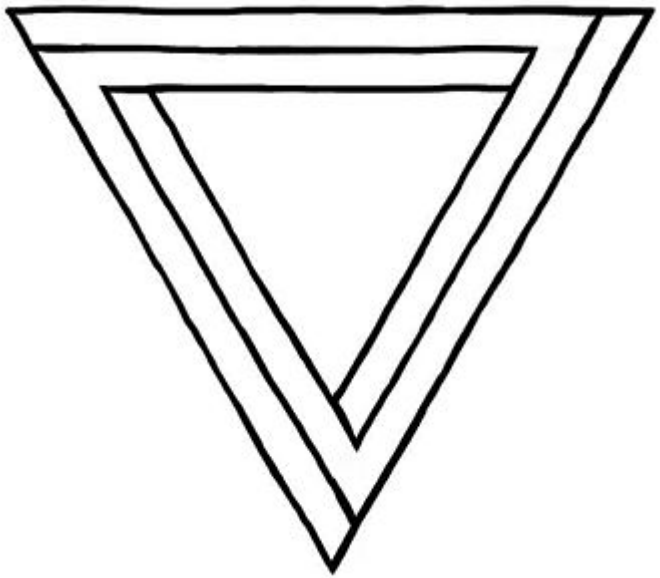
This makes programming a challenging and fun task

Problem Solving Step by Step

- Computational thinking involves problem solving by breaking down the problem to a series of steps that can be achieved
  - *Problem decomposition*
  - *If these steps still too big, decompose further*
  - *"A journey of a thousand miles begins with a single step" (Chinese saying, based on a quote from Lao Tzu)*

Exercise

- Write step by step instructions for a person (may be sitting next to you) to draw the below mentioned image.



Is it easy or difficult ? Test it yourself

Recipe to boil an Egg

- Step 1:

  Bring your eggs to room temperature before boiling. If the eggs are too cold, the shells may crack during cooking.

- Step 2:

  Place the eggs in a saucepan of cold water. Place the pan over medium heat. Bring to a gentle simmer, gently stirring the eggs constantly in a clockwise direction. The movement of the water helps to centre the egg yolks.

- Step 3:

  Simmer the eggs for 4 minutes for soft-boiled eggs. For semi-firm yolks and hard whites, simmer for 5 minutes. For hard-boiled eggs, simmer for 8 minutes. Use a slotted spoon to remove the egg from the water. Transfer to an egg cup and serve immediately.

Source: http://www.taste.com.au/how+to/articles/2508/how+to+boil+eggs

Recipe to boil an Egg (2)

- **Step 1**:

  - *Wait until your eggs reach room temperature*.

- **Step 2**:

  - *Place the eggs in a saucepan of cold water.*

  - *Place the pan over medium heat.*

  - *Until temperature between 90°C to 95°C stir the eggs gently in a clockwise direction. That is a simmer.*

- **Step 3**:

  - *If soft-boiled eggs desired, simmer the eggs for 4 minutes*

    else If semi-firm yolks and hard whites desired, simmer for 5 minutes.

    else if hard-boiled eggs desired, simmer for 8 minutes.

  - Use a slotted spoon to remove the egg from the water and transfer to an egg cup

  - Serve immediately.

Actions words – RED , Control words – BLUE

What is a Computer Program?

- A detailed, step–by–step set of instructions *executed* by a computer
  - *Programming is the creation of the lists of instructions*
- If we change the program, the computer performs a different set of actions or a different task.
- That is, the machine stays the same, but the program changes!
  - *Compare with mechanical systems, e.g. locks.*

Masterlock.com

What is Computer Science?

- It is NOT the study of computers!

  "*Computers are to computer science what telescopes are to astronomy.*"

  Edsger Dijkstra

- Since a computer can carry out any computation, the question really is,
  - *"What computations we can describe?"*
- The fundamental question is,
  - *"What can be computed"?*

Wikipedia.org

What is Computer Science?

- Computer scientists find the answers to questions through
  - *Design*
  - *Analysis*
  - *Experimentation*

Design

- One way to show a particular problem can be solved is to actually design a solution.
- This is done by developing an algorithm
- *Algorithm*: A step–by–step process for achieving the desired result
  - *An algorithm is simply an abstract recipe*
  - *A program implements that recipe in a particular computer language*
- This Unit will teach you how to
  - *design an algorithm*
  - *write a program for it*

Analysis

- "**Design**" can only answer the question "What is computable?" in the positive.
  - *Not being able to design an algorithm does not mean it is unsolvable.*
- **Analysis** is the process of examining algorithms and problems mathematically.
- Some seemingly simple problems are unsolvable by any algorithm.
  - *Integer partition: Can you partition n integers into two subsets such that the sums of the subsets are equal.*
- Ways of comparing algorithms, e.g. time required to solve problem, or memory required, as a function of the size of the input

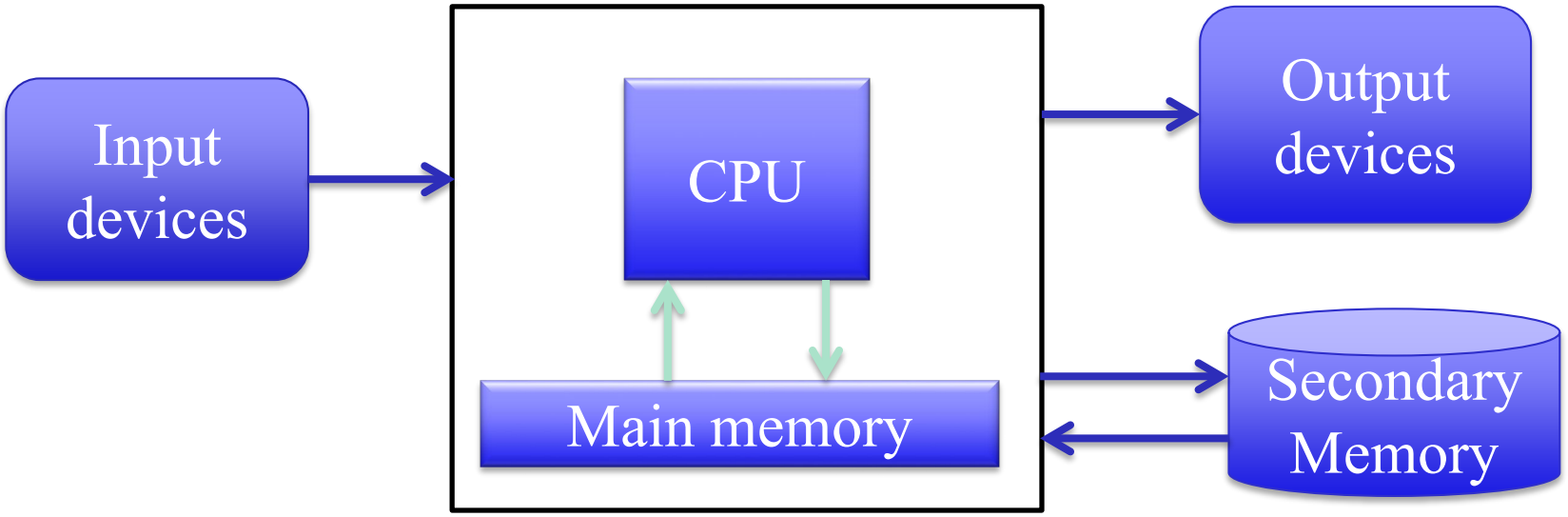
Experimentation

- Some problems are too complex for analysis.
  - *World climate*

- Implement a system and then study its behaviour under different conditions *(Generalization)*

- Experimentation is sometimes still needed after theoretical analysis
  - *To verify the analysis*
  - *To refine the analysis*

Computer Hardware Basics



- The Central Processing Unit (CPU) carries out the computations
  - *Just simple instructions like adding two numbers.*

Computer Hardware

- Memory stores programs and data.
- CPU can only directly access information from the main memory: Random Access Memory (RAM)
- RAM is fast but volatile i.e. all information is lost when power is lost.
- Secondary memory provides more permanent storage (non-volatile).
  - *Magnetic (hard drive)*
  - *Optical (CD, DVD, Blue Ray Disc)*
  - *Solid state drives (USB, SSD memory)*

Input Devices

- Input devices – pass information to the computer
  - *Keyboards and Mice*
  - *Touch pads*
  - *Camera*
  - *Microphone*
  - *Sensors, e.g. accelerometer, gryoscope, data glove*

Output Devices

- Output devices – pass information back to the user or device
  - *Screen*
  - *Printer*
  - *Speaker*
  - *Motor actuator, e.g. robot arm*

The Fetch Execute Cycle

1. Load program into the main memory (RAM)
2. Fetch the next instruction from memory
3. Decode the instruction to see what it represents
   - *Fetch data as required*
4. Carry out the appropriate instruction.

Instructions, data, memory locations – ***everything*** – is represented as binary numbers

Programming Languages

- Natural languages cannot precisely describe an algorithm.
  - *Try giving directions without waving your arms about your arrival to lecture theatre*
- Programming languages used to express algorithms in a precise way.
- Every structure in a programming language has a precise *form* called its ***syntax***.
- Every structure in a programming language has a precise *meaning* called its ***semantics***.

Programming Language Levels

- High-level programming languages
  - *Designed to be understood and written by humans*

- Low-level language
  - *Computer hardware can only understand a very low level language known as machine language*

High-level Programming Language

- In a high-level language, a typical statement may be

  $b = a + 2 \times b$

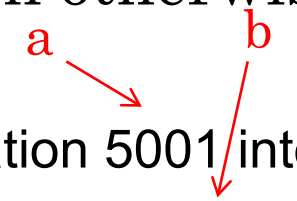
  Note the sequence of operations that is implied

- This needs to be translated to machine language so the computer can execute it

- Compilers convert programs written in high-level languages into machine language in one go
- Interpreters do the same instruction by instruction

Low-level Language

- The corresponding low-level language may look something like this: if translated in English otherwise it is 0's or 1's



  - Load the number from memory location 5001 into CPU Register 0
  - Load the number from memory location 5002 into the CPU Register 1
  - Multiply value in CPU Register 1 by 2 and restore in CPU Register 1
  - Add Register 0 to CPU Register 1 and restore in CPU Register 0
  - Store CPU Register 0 into memory location 5002

Note: A Register is a space for temporary results in the CPU (very fast access)

Compiling vs Interpreting

Compiling

- Once program is compiled, the machine language program can be executed over and over without the source code or compiler
- Compiled programs generally run faster since the translation of the program happens only once
- Program needs to be compiled after every minor change in it
- A program compiled for Windows will not run on OS (Mac) or Linux
- C, C++ language programs

Interpreting

- The source code and interpreter are needed each time the program is executed
- Interpreted programs run slower due to each line being interpreted each time it is executed
- More flexible programming environment since programs can be developed and run interactively
- Interpreted programs are more portable across different platforms e.g. Macs, Windows, Linux
- Python, Java language programs

Python 3

- We will be using Python 3 which is embedded in Thonny
- When you start Python, you may see something like:

```
Python 3.6.4
>>>
```

- >>> is a Python prompt indicating that Python is ready for us to give it a command. These commands are called statements.

```
>>> print("Hello, world")
Hello, world
>>> print(2+3)
5
>>> print("2+3=", 2+3)
2+3= 5
>>>
```

Summary

- Understanding the roles of hardware and software in a computing system.
- Learning what computer scientists study and the techniques they use.
- Understanding the basic design of a modern computer.
- Understanding the form and function of programming languages, and how programs in those languages are executed.