University of Nottingham
UK | CHINA | MALAYSIA

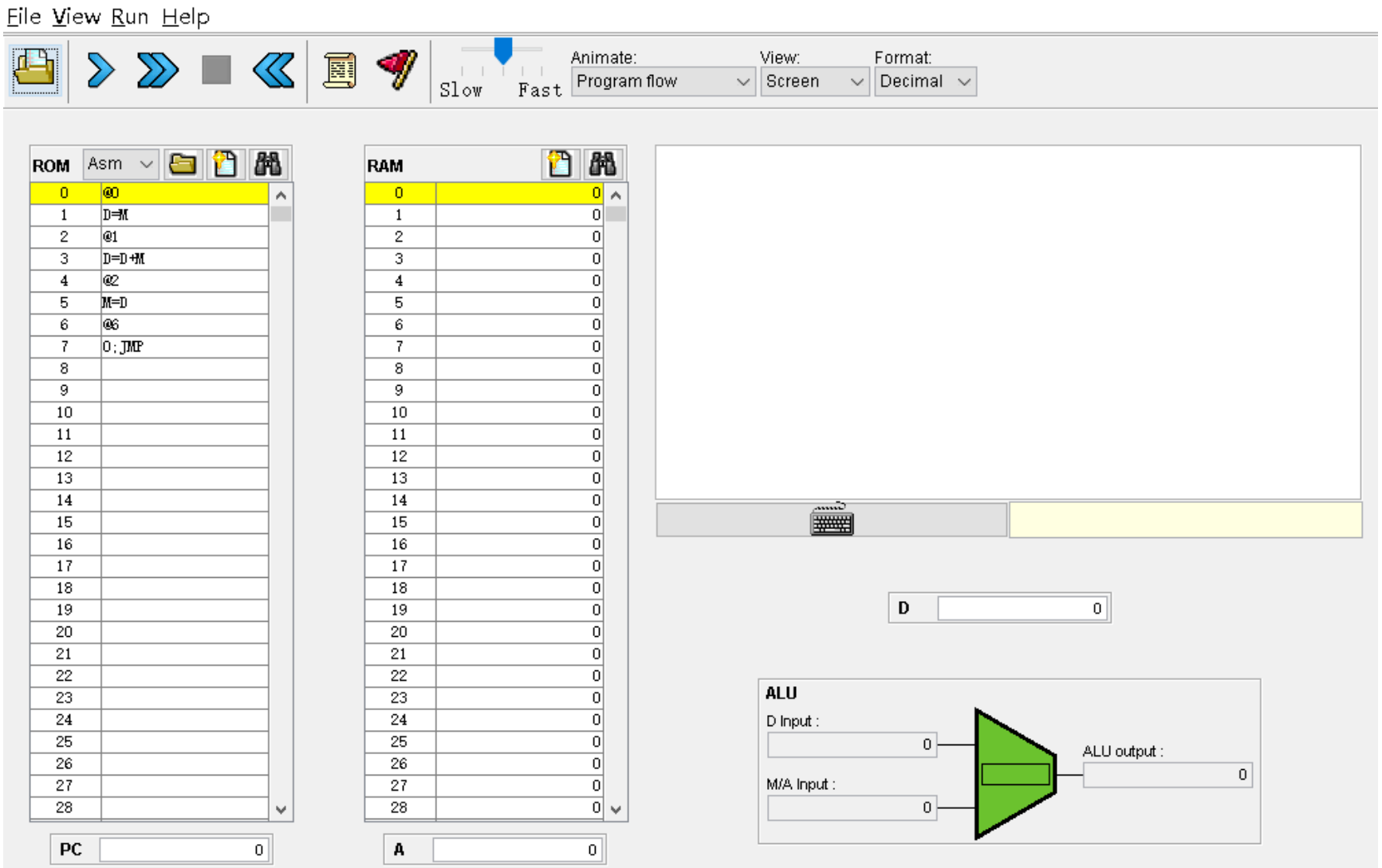
Machine Language (Part 1)

Dr. Wooi Ping Cheah

Lab Project

- Project objectives: to have a taste of
  - Low-level programming
  - Hack assembly language
  - Hack hardware

CPU Emulator



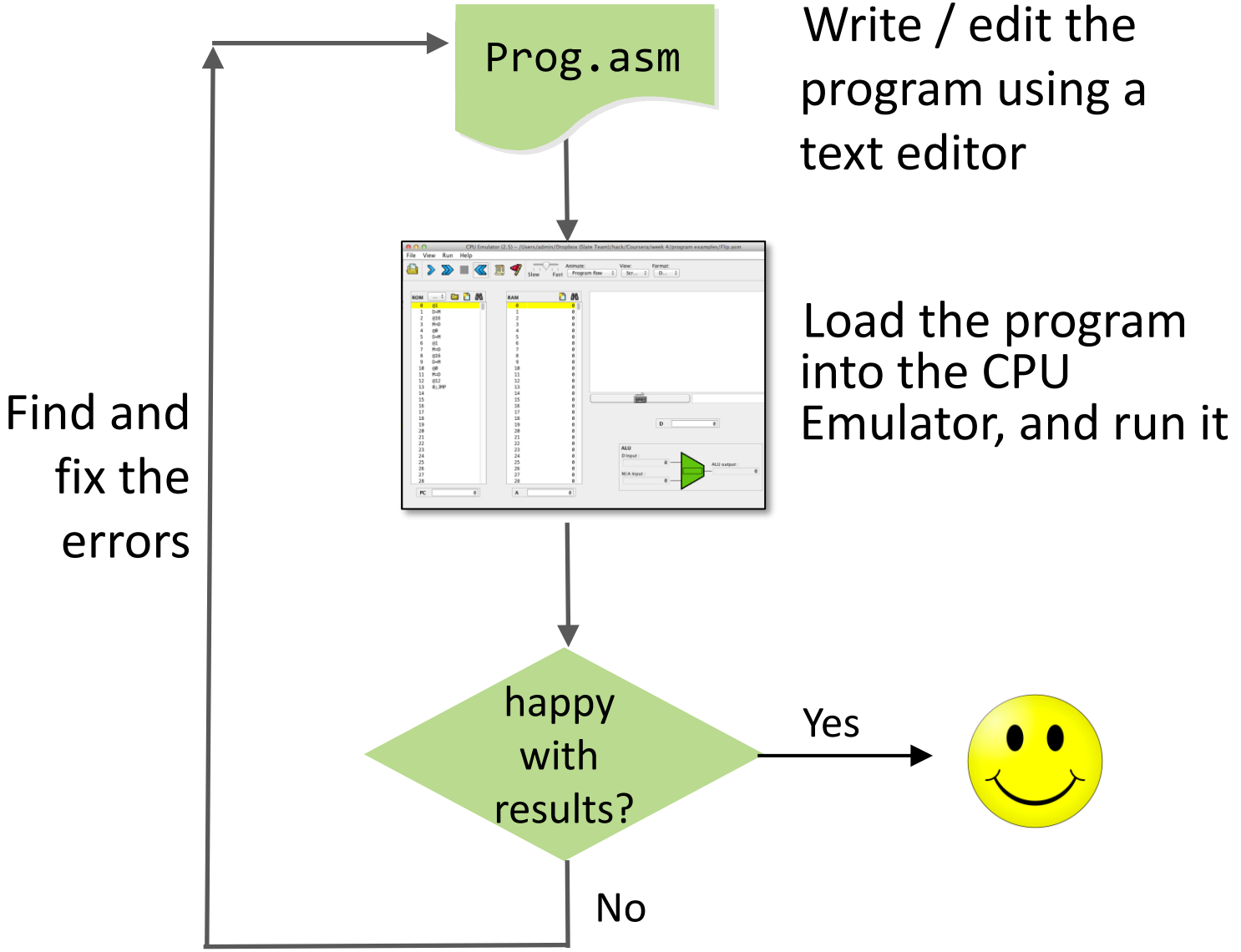
CPU Emulator

- **Slow ... Fast** (change the speed of emulation).
- **Animation**
  - **Program flow**: show how the program proceeds.
  - **Program & data flow**: show how the program proceeds and the data vary.
  - **No animation**: no animation will be shown.
- **View**
  - Script: show script.
  - Output: show running results.
  - Compare: not in used for this lab.
  - Screen: show Hack Computer screen. (256×512, B/W)
- **Format**: show numbers in decimal, hexamal or binary.

Misc

- File suffix
  - Binary code (.hack) files.
  - Assembly language (.asm) files.
- Test procedures
  - Open CPU Emulator.
  - Load xxx.asm into ROM, symbolic code without symbols (e.g. predefined symbols, labels, variables).
  - Load xxx.tst test script, run the test script.

Program development process



Best practice

Well-written low-level code is

- Short
- Efficient
- Elegant
- Self-describing

Technical tips

- Use symbolic variables and labels
- Use sensible variable and label names
- Variables: lower-case
- Labels: upper-case
- Use indentation
- Start with pseudo code.

Task 1: add two numbers

- Input: RAM[0] and RAM[1].
- Output: RAM[2] = RAM[0]+RAM[1].
- add2.asm

Task 2: swap two numbers

- Set RAM[0] = 50, RAM[1] = 100, then swap the value of RAM[0] and RAM[1]. You may use RAM[16] as the temporary variable.

Task 3: signum

- Implement signum.asm to achieve the following function.

```
// Program: signum.asm
// Computes:
//   if RAM[0]>0
//         RAM[1]=1
//   else
//         RAM[1]=0
// Usage: put a value in RAM[0],
//       run and inspect RAM[1].
```

Task 4: sgn function

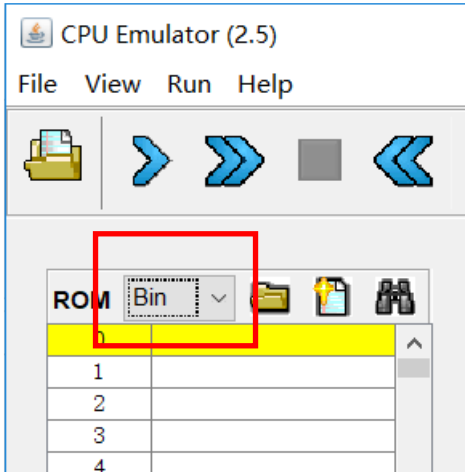
- Implement sgn function as follow.

$$\operatorname{sgn}(x) := \begin{cases} -1 & \text{if } x < 0, \\ 0 & \text{if } x = 0, \\ 1 & \text{if } x > 0. \end{cases}$$

- You may assume that x is stored in RAM[0], and the returned value of function sgn(x) is stored in RAM[1].
- You should use R0, R1.
- You should use labels such as NEGATIVE, ZERO and END.

Task 5: assembly to binary

- For task 1-4, choose one of them, translate the assembly code to binary code. Compare your translation with the translation by CPU Emulator.



Acknowlegement

- This set of lecture notes are based on the lecture notes provided by Noam Nisam / Shimon Schocken.
- You may find more information on: www.nand2tetris.org.