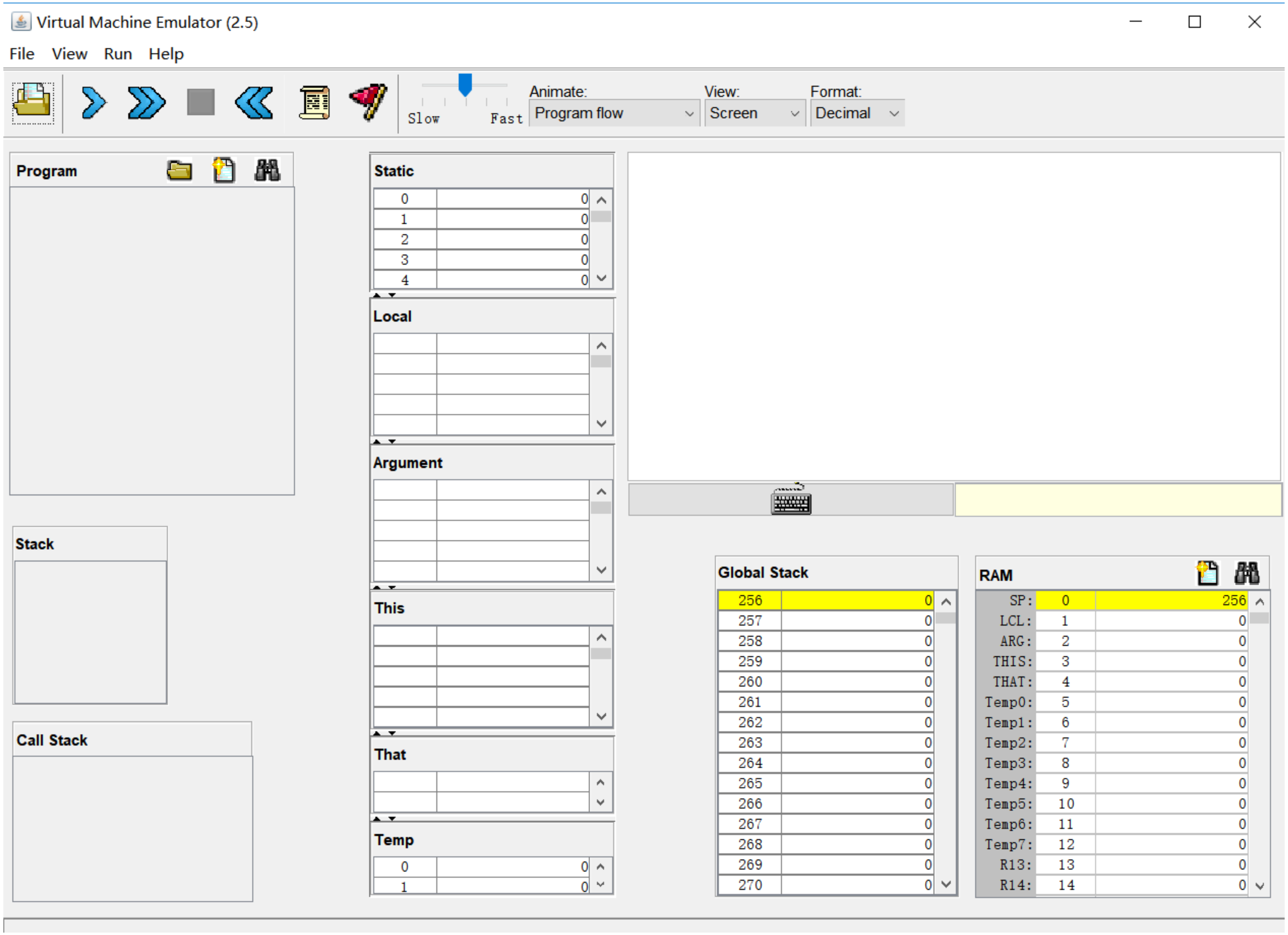
University of Nottingham
UK | CHINA | MALAYSIA

Virtual Machine

Dr. Wooi Ping Cheah

Introduction to VMEmulator



Animate: Program flow

View: Screen

Format: Decimal

Static

0	0
1	0
2	0
3	0
4	0

Temp

0	0
1	0

Global Stack

256	0
257	0
258	0
259	0
260	0
261	0
262	0
263	0
264	0
265	0
266	0
267	0
268	0
269	0
270	0

RAM

SP:	0	256
LCL:	1	0
ARG:	2	0
THIS:	3	0
THAT:	4	0
Temp0:	5	0
Temp1:	6	0
Temp2:	7	0
Temp3:	8	0
Temp4:	9	0
Temp5:	10	0
Temp6:	11	0
Temp7:	12	0
R13:	13	0
R14:	14	0

VMEmulator Memory Layout

- Pointers: RAM[0 … 4]

  Base address for various memory segments: Global Stack (SP), Local (LCL), Argument (ARG), THIS, THAT
- Temp0 –Temp7: RAM[5 … 12]
- General: RAM[13 … 15]
- Static: RAM[16 ... 255]
- Stack: RAM[256 ... 2047]
  Global Stack, Local, Argument
- Heap: RAM[2048 onward]

  Dynamic memory locations pointed with pointers THIS and THAT

How to test VM codes?

- Two ways to test VM codes:

  - ➢Manually set the base memory address first before you test the VM codes, e.g.
    - □set sp 256,   // stack pointer
    - □set local 300,   // base address of the local segment
    - □set argument 400,   // base address of the argument segment
    - □set this 3000,   // base address of the this segment
    - □set that 3010,   // base address of the that segment

  - ➢Use test scripts to test the VM codes.

    e.g. **BasicTest.tst**

Task 1: Tracking Stack Status

- Given the following VM codes, manually track the stack status after each stack operation. What is the final status of the stack?
    - □SimpleAdd.vm,
    - □StackTest.vm,
    - □BasicTest.vm,
    - □StaticTest.vm,
    - □PointerTest.vm.
- Verify your answers by running the VM codes on the VMEmulator.

Examples of Bitwise Operations

- Examples of bitwise operations found in **StackTest.vm**

Notation:

1’s – 1’s Complement

2’s – 2’s Complement

Boolean Value Representation:

True – -1 (Binary: 1111111111111111)

False – 0 (Binary: 0000000000000000)

Examples of Bitwise Operations

- Examples of bitwise operations found in **StackTest.vm**

57 and 28 = 24

57	0000000000111001
28	0000000000011100
and	0000000000011000 = 24

24 or 82 = 90

24	0000000000011000
82	0000000001010010
or	0000000001011010 = 90

Examples of Bitwise Operations

- Examples of bitwise operations found in **StackTest.vm**

57 and 0 = 24

57	0000000000111001
0	0000000000000000
and	0000000000000000 = 0

57 or -1 = -1

57	0000000000111001
0	1111111111111111
or	1111111111111111 = -1

Examples of Bitwise Operations

- Examples of bitwise operations found in **StackTest.vm**

not 90 = -91

90 0000000001011010

not 1111111110100101 = -91

-(-91)

-91 1111111110100101

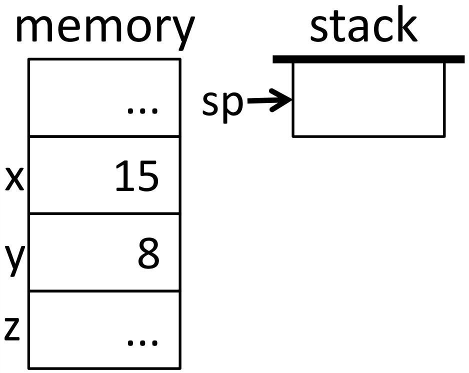
1’s 0000000001011010

2’s 0000000001011011 = 91

Task 2: Show Stack Operations

VM code

```
…..
…..
…..
…..
…..
…..
…..
…..
```



Show the stack operations for the following expression:

z = x > 8 and y < 8

What is the final memory status?

Task 3: Write VM Loop Function

- Write a basic loop function in VM code.
  - Computes the sum 1 + 2 + ... + Argument[0], and pushes the result onto the stack.
- Test the VM code on VMEmulator.
- You may refer to the sample code called ***mult*** for multiplying two numbers.

Task 4: Translate VM to Assembly

- What is the functionality of the following VM code?

```
push local 0
push local 1
add
pop local 2
```

- Translate it to Hack Assembly Code.

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.