## IN1006 Systems Architecture 2024\_2025 Tutorial 4: Instruction Set Architecture and Assembly Programming

1. What are the components of a very simple computer?

CPU, Memory, IO

2. Name some special purpose registers?

Accumulator, Program Counter

3. What is the purpose of a clock in a computer processor?

Every computer contains at least one clock that synchronises the activities of its components. A fixed number of clock cycles are required to carry out each data movement or computational operation

4. What is meant by an Instruction Set Architecture?

A specification of the interface between software and hardware describing the operations and addressing format

5. What classes of instructions generally make up an ISA?

Arithmetic, memory access, I/O, control

- 6. What are the advantages of assembly programming over machine code? *Human friendly, extra structuring features*
- 7. Write a MARIE assembly language program to:

Read 3 values from memory addresses FA, FB, FC, sum them, display the value and write the value back to FD (assuming no overflow)

LOAD FA

ADD FB

ADD FC

OUTPUT

STORE FD

8. Write a simple Marie program that reads 3 numbers from memory, add the first 2 together, subtract the third, output the result and store the result back to memory. i.e. result = X+Y-Z; print(result);

LOAD X ADD Y SUBT Z

```
OUTPUT
STORE RESULT
HALT
X, DEC 6
Y, DEC 3
Z, DEC 4
RESULT, DEC 0
```

9. What does these Marie code do?

INPUT
STORE W
ADD W
STORE X
ADD X
STORE Y
ADD Y
STORE Z
ADD Z
OUTPUT
HALT
W, DEC 0
X, DEC 0
Y, DEC 0
Z, DEC 0

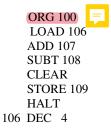
This program reads a value from input to the accumulator, store it in memory doubling it up 4 times, outputting the final value, e.g. if 1 is entered it does 1, 2, 4, 8, 16 and prints out 16

10. What does this Marie code print to the output

ORG 100 LOAD 104 ADDI 105 OUTPUT HALT 104 HEX 4 105 HEX 106 106 HEX 3

-This code add indirectly (through 105) the values in address 104 and 106. If there was not ADDI but just ADD, the result would be  $010A_h$  which is 266 in decimal.

11. After executing the following Marie code, what value will be stored at address location 109?



```
107 DEC 9
108 DEC 5
```

- Zero will be stored at address 109, as the CLEAR instruction will set the Accumulator to 0 before the store instruction.
- 12. Write a program in Marie that print/output the numbers from 1 to 10. In high level language it would look like:

```
for(int i = 0; i <= 10; i++){
  print(i);
  LOOP,
          LOAD COUNTER
          ADD ONE
          OUTPUT
          STORE COUNTER
          SUBT TEN
          SKIPCOND 400
          JUMP LOOP
          HALT
COUNTER, DEC 0
ONE,
          DEC 1
          DEC 10
TEN,
```

13. Using a subroutine, write a program that reads two numbers from memory, multiply them and show the result in the output.

```
i.e C = AxB;
```

Using a routine would look like this in high level language:

```
result = multiply(2,3);
print(result);
```

```
ORG 100
       LOAD A
       STORE X
       LOAD B
       STORE Y
       JnS multiply
                      /Jump to multiplication subroutine
       LOAD RESULT /Get result
       OUTPUT
       HALT
       Dec 2
В,
       Dec 3
X,
       Dec 0
                      /First parameter
Y,
       Dec 0
                      /Second parameter
```

## //\*\*\*\*Subroutine to multiply\*\*\*

Loop,

multiply, Hex 0 /Store return address here

Load X /Load first parameter to be used as counter

Store Counter /Counter = X Load result /Load the sum

Add Y /Add second parameter Store result /Store result in result

Load Counter /load to AC the value in Counter

Subt One /Decrement counter Store Counter /Store counter

SkipCond 400 /If AC == 0 skip next line Jump Loop /jump to label loop

JumpI multiply /return or jump from subroutine to value stored in multiply which will indirectly jump back to instruction after Jns in main program.

Counter, Dec 0 /Counter for looping
One, Dec 1 /Constant with value 1

Result, Dec 0 /return value for multiplication subroutine

## 14. Write a MARIE assembly language program to:

Calculate  $A \times B + C \times D$ 

Hint: treat multiplication as a series of additions. Use the commands Skipcond together with Jump to loop. Use the multiply subroutine from Question 13 or look at the examples in the MARIE simulator directory that you downloaded when you installed the simulator.

15. Consider the following MARIE assembly language program. What does the program do?

INPUT
LOOP, STORE COUNTER
OUTPUT
SUBT ONE
SKIPCOND 000
JUMP LOOP
HALT
COUNTER, DEC 0
ONE, DEC 1

- -This takes a number n from the input and prints all the numbers in reverse order from n down to 0.
  - 16. Describe the F-D-E cycle for a simple machine.

The **fetch-decode-execute cycle** is the series of steps that a computer carries out when it runs a program.

We first have to fetch an instruction from memory, and place it into the IR. Once in the IR, it is decoded to determine what needs to be done next.

If a memory value (operand) is involved in the operation, it is retrieved and placed into the MBR.

With everything in place, the instruction is executed.