

EECE.3170: Microprocessor Systems Design I

Key Questions

Conditional execution (Lec. 16-18)

QUESTIONS

1. Describe the operation of the compare instruction.
2. Describe the different x86 condition codes.
3. Describe the operation of the conditional move instruction.
4. Describe the operation of the SETcc instruction. How can this instruction be used?
5. Describe the two general classes of jump instruction.
6. Describe the x86 loop instructions, as well as how these instructions can be used in a typical program.
7. Rewrite the post-tested loop example from earlier to use a loop instruction.

```
L:   MOV  CX, 5
      SHL  AX, 1
      DEC  CX
      JNZ  L
```

EXAMPLES

1. Show the results of the following instructions, assuming that
(0x100) = 0x0001, (0x102) = 0x0003, (0x104) = 0x1011, (0x106) = 0x1011,
(0x108) = 0xABCD, (0x10A) = 0xDCBA

What complex condition does this sequence test?

```
MOV    AX, [0x100]
CMP    AX, [0x102]
SETLE  BL
MOV    AX, [0x104]
CMP    AX, [0x106]
SETE   BH
AND    BL, BH
MOV    AX, [0x108]
CMP    AX, [0x10A]
SETNE  BH
OR     BL, BH
```

2. Given the instructions below, what are the resulting register values if:
- AX = 0x0010, BX = 0x0010
 - AX = 0x1234, BX = 0x4321

What type of high-level program structure does this sequence demonstrate?

```
        CMP  AX, BX
        JE   L1
        ADD  AX, 1
        JMP  L2
L1:     SUB  AX, 1
L2:     MOV  [0x100], AX
```

3. Given the instructions below, what are the resulting register values if, initially, AX = 0x0001?

What type of high-level program structure does this sequence demonstrate?

```
        MOV  CX, 5
L:      SHL  AX, 1
        DEC  CX
        JNZ  L
```

4. Given the instructions below, what are the resulting register values if, initially, AX = 0x0001?

What type of high-level program structure does this sequence demonstrate?

```
        MOV  CX, 5
L:      JCXZ  END
        ADD  AX, AX
        DEC  CX
        JMP  L
END:    MOV  [0x10], AX
```

5. Describe the operation of the following program.
What is the final value of SI if the 15 bytes between 0x0A001 and 0x0A00F have the following values?

00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E

```
        MOV  DL, 05
        MOV  EAX, 0x000A000
        MOV  ESI, 0
        MOV  CX, 0x000F
AGAIN:  INC  ESI
        CMP  [EAX+ESI], DL
        LOOPNE AGAIN
```