# CSC111 Fall 2025 Midterm Sample Exam Questions

1. For the following multiple-choice questions, each question has exactly one correct answer key. If multiple choices are correct, choose the option "All of the above". **Fill in the answer keys in the table above. (Answer keys written in the question area will not be counted.)**

MCPs will be based on the quizzes.

**1.** For the 4-bit binary values in the table below, show the equivalent decimal values when the data is interpreted as unsigned binary or signed binary.

| Binary Value | Signed Decimal Value | Unsigned Decimal Value |
|---|---|---|
| **0000** | **0** | **0** |
| **0001** | **1** | **1** |
| **0111** | **7** | **7** |
| **1000** | **-8** | **8** |

**2.** (a) Assume a 4-bit system. Copy bits 3..2 in R4 to be the right most bits of R5, and set other bits in R5 to 0. Use only LSL, LSR, AND instructions.
(b) Assume a 4-bit system. Copy bits 3..2 in R4 to be the left most bits of R5, and keep other bits in R5 unchanged. Use only LSL, LSR, AND, OR instructions.

A*NS*:
(a)

```
AND R5, R4, #0b1100
LSR R5, R5, #2
```

(b)

```
AND R3, R4, #0b1100
AND R5, R5, #0b0011
ORR R5, R5, R3
```

**3.** Fill in the table for the N, V flags and condition for signed comparison.

ANS:

| **Signed Compare Example** | **N** | **V** | **Condition (GE or LT)** |
|---|---|---|---|
| %R1 = 0101, R2 = 0011<br>CMP R1, R2 | | | |
| %R1 = 0111, R2 = 1101<br>CMP R1, R2 | | | |
| %R1 = 0010, R2 = 0101<br>CMP R1, R2 | | | |
| %R1 = 1001, R2 = 0011<br>CMP R1, R2 | | | |

ANS:

| Signed Compare Example | N | V | Condition (GE or LT) |
|---|---|---|---|
| %R1 = 0101, R2 = 0011<br>CMP R1, R2 | 0 | 0 | GE |
| %R1 = 0111, R2 = 1101<br>CMP R1, R2 | 1 | 1 | GE |
| %R1 = 0010, R2 = 0101<br>CMP R1, R2 | 1 | 0 | LT |
| %R1 = 1001, R2 = 0011<br>CMP R1, R2 | 0 | 1 | LT |

**4.** Assume a 5-bit system. Fill in the table for the N, V flags and condition for signed comparison.

ANS:

| Signed Compare Example | N | V | Condition (GE or LT) |
|---|---|---|---|
| %R1 = 00111, R2 = 00011<br>CMP R1, R2 | | | |
| %R1 = 00011, R2 = 00111<br>CMP R1, R2 | | | |
| %R1 = 10110, R2 = 00111<br>CMP R1, R2 | | | |
| %R1 = 01010, R2 = 11001<br>CMP R1, R2 | | | |

ANS:

| Signed Compare Example | N | V | Condition (GE or LT) |
|---|---|---|---|
| %R1 = 00111, R2 = 00011<br>CMP R1, R2 | 0 | 0 | GE |
| %R1 = 00011, R2 = 00111<br>CMP R1, R2 | 1 | 0 | LT |
| %R1 = 10110, R2 = 00111<br>CMP R1, R2 | 0 | 1 | LT |
| %R1 = 01010, R2 = 11001<br>CMP R1, R2 | 1 | 1 | E |

**5.** (20 points) Program understanding. Translate the following assembly program code into C and explain what it does.

| Assembly program | C program |
|---|---|
| MOV R2, #1<br>MOV R1, #1<br>loop CMP R1, R0<br>BGT done<br>MUL R2, R1, R2<br>ADD R1, R1, #1<br>B loop<br><br>done MOV R0, R2 | |

**ANS: Factorial = R0!**

| Assembly program | C program |
|---|---|

```
        MOV R2, #1
        MOV R1, #1
loop    CMP R1, R0
        BGT done
        MUL R2, R1, R2
        ADD R1, R1, #1
        B       loop

done    MOV R0, R2
```

```
int factorial(int n) {
   int result = 1;   // R2
    int i = 1;       // R1
    while (i <= n) {  // loop until i > ns
       result = result * i;
       i = i + 1;
    }
    return result;    // R0 holds return value
}
```