

CSE340: Computer Architecture

Assignment 2

Chapter 2 - Instructions: Language of the Computer (MIPS Instructions)
and Chapter 3 - Arithmetic for Computers (Multiplication part)

Total Marks: 30 (Marks are indicated in third brackets after each question)

1. **[CO2] Explain** the difference between Program Counter and \$zero. In the case of 16-bit and 128-bit architecture, what would be the increment in memory address for sequential instruction execution? **[2]**
2. **[CO2]** Let us consider the instruction `lw $4, X($5)`. Now, suppose we have an array *A* and the base address of that array is 256 in decimal. If we are looking to load the contents of *A*[5], **identify** the value of *X* in the `lw` instruction in the case of 256-bit architecture. **[1]**
3. **[CO2]** Assume that the base address of the array *A* is in `$s0`, and the values of *i* and *f* are stored in `$s1` and `$s2`. Then **translate** the following statement into MIPS assembly code. Assume that *A* is a byte array and *f* and *i* are 32-bit integers. **[1]**

$$f = A[i]$$

4. **[CO2]** Let us consider the set of instructions given below. Here, *X* and *Y* are in registers `$s0` and `$s1` respectively. The base address of the array *Arr* is in register `$s4`. Now, write the equivalent MIPS code for the given set of instructions, **identify** the instruction type and write the machine code for each instruction. **[5]**

$$X = 15Y - 5;$$
$$Arr[5] = 2X + Arr[10];$$

5. **[CO3] Calculate** the branch destination address of the instruction `beq $9, $8, 124` if the PC holds `0x1278A4B1`? Show all the steps and write the calculated branch address in hex. **[2]**

6. **[CO2]** Identify the jump address of the instruction j 1590 if the PC holds 0x00AB1203? **Show** the steps in your calculations and write the final address in hex. **[1]**

7. **[CO2]** Consider the instruction: lw \$8, 52(\$17). If the base address is 0x15632017. **Identify** the memory address of the data that will be loaded to \$8? **[1]**

8. **[CO2]** Given the following code sequence:

```
for (i = 0; i < 10; i++) {  
    if (A[i] != 5)  
        A[B[i]] += 1  
    else  
        A[i] = B[i+1] }
```

If the base address of arrays A and B are in \$s1 and \$s2 respectively and i, 5, 1 are in \$s3, \$s4, \$s5, **convert** the above code to its equivalent MIPS code. **[4]**

9. **[CO2]** Consider the following code sequence:

```
x = 20;  
y = x - 10;  
a = 7;  
z = y + a;  
total = sum(x, y, z);
```

```
int sum (x, y, z){  
    a = x + y + z;  
    return a;  
}
```

Suppose the values a, x, y and z are in \$s0, \$s1, \$s2 and \$s3 respectively.

Translate the code to its equivalent MIPS code. **[3]**

10. **[CO3]** Based on what we have learned about 32-bit MIPS architecture, **calculate** the size of the data memory for a 64-bit MIPS architecture? **[2]**