**SOLUTION**

CSC332 Spring 2021 HW1 (Sec 2.4 The CPU)

**Due: Thurs March 4 midnight.**

**Submit your solution on Blackboard (BB), not my email.**

**Please write your answers in WORD, Wordpad, Notepad, or pdf files and upload it to the system.**

**Do only one question, based on the first letter of your last name:**

**A-M: Q1**

**N-Z: Q2**

Q1. Convert the following code fragment to assembly code fragment, **using instructions discussed in class.**

if X > Y then Y=X

else X=Y

Load X, R1

Load Y, R2

cmp R1, R2

jmpp L1

Store R2, X

Jmp L2

L1:Store R1, Y

L2:

**Comments on student work:**

1. **Many are using instructions not in slide. I cannot guess your assembly language. Follow the instructions in the question.**
2. **Some are saying things like CMP X, Y. In the slides, CMP takes two registers, not memory.**
3. **Many store a value in X, and then store a value in Y. There should be a JMP in-between.**

Q2.

The PC is incremented at the end of Fetch Cycle. Suppose, instead, we do it at the end of the Execute Cycle. Will there be any problem? (Hint: look at the instructions discussed in class).

Solution:

Consider the instruction:

Jmp L

Suppose the value of L is 100. So we want to jump to address 100.

In execution cycle, PC will become 100.

If PC is incremented at end of execution cycle, then it will become 102 (assuming that the instruction length is 2 bytes).

So it will next fetch the instruction from address 102, which is wrong.

**Comments on student work:**

1. **Many are saying that it would hurt performance due to pipelining issues. Performance is much less important than correctness. Here there is a correctness problem.**
2. **In computers not using pipelining, there won’t be performance problem. But correctness problem will be there.**