**CS3375: Computer Architecture**
**Spring 2020**

**Review #8 Solution**

- Full name only: _____
- Release date: Apr 22nd, 2020 (Wednesday)
- Due date: **Apr 22nd, 2020 (Wednesday) before midnight**
- Total 5 points

1. The operation times for the major functional units are 200ps for memory access, 200ps for ALU operation, and 100ps for register file read or write. For example, in single-cycle design, the time required for every instruction is 800ps due to `lw` instruction (instruction fetch, register read, ALU operation, data access, and register write). Here, we only consider `lw` instruction for speedup comparison.

[2 pts]

a. [True/False]: If the time for an ALU operation can be shortened by 25%, it will affect the speedup obtained from pipelining.

a. False. Shortening the ALU operation will not affect the speedup obtained from pipelining. It would not affect the clock cycle.

b. [True/False]: If the ALU operation now takes 25% more time, it will affect the speedup obtained from pipelining.

b. True. If the ALU operation takes 25% more time, it becomes the bottleneck in the pipeline. The clock cycle needs to be 250ps.

2. Consider three branch prediction schemes: predict not taken, predict taken, and dynamic prediction. Assume that they all have zero penalties when they predict correctly and two cycles when they are wrong. Assume that the average predict accuracy of the dynamic predictor is 90%. If a branch is taken with 5% frequency, which predictor is the best choice? [ a ]

[1 pt]

a. Predict not taken

b. Dynamic prediction

c. Predict taken

3. Consider executing the following code on the pipelined datapath,

```
lw  $4, 100($2)
sub $6, $4, $3
add $2, $3, $5
```

How many clock cycles will it take to execute this code? [HINT]: You can use a data forwarding.

[2 pts]

- It will take 8 clock cycles to execute this code, including a bubble of 1 cycle due to the dependency between the `lw` and `sub` instructions.