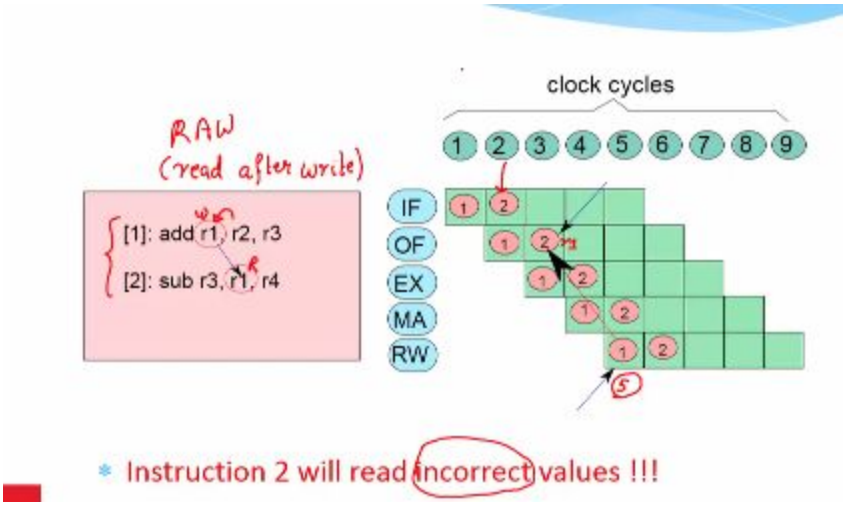
Assignment 5

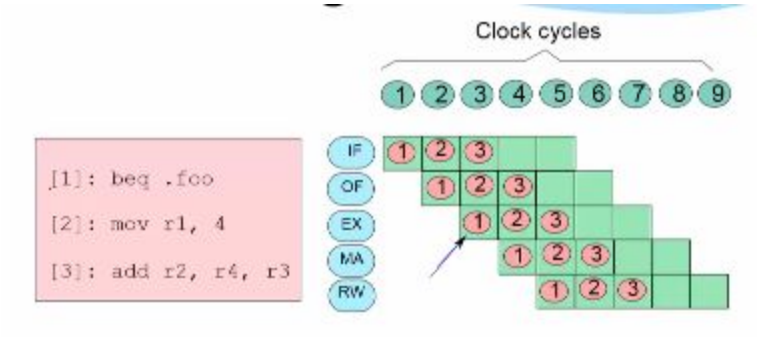
Question 1:
Hazard: a hazard is defined as possibility of erroneous execution of an instruction in a pipeline.

Data Hazards: data hazard represents the possibility of erroneous execution because of unavailability of data, or availability of incorrect data.
Types: RAW, WAR, WAW.



So we can say that it is the RAW(read after write) type of hazard as an update of value at r1 will be done in cycle 5 but the next statement tries to use that value in cycle 3 so it is going to use incorrect value so it is an example of **data hazard**.

Control hazard: control hazard represents the possibility of erroneous execution in a pipeline because instruction in the wrong path of a branch can possibly get executed and save their results in memory or in the register file.



The two instructions fetched immediately after branch instruction might have been fetched incorrectly. As .foo function should have been called if equal flag==1 but instruction 2,3 have already passed in the pipeline

Structural Hazard: A structural hazard may occur when two instructions have conflict on the same set of resources in a cycle.
Assume that we have an instruction that can read one operand from memory.

```
[1]: st r4, 20[r5]
[2]: sub r8, r9, r10
[3]: add r1, r2, 10[r3]
```

[3] will try to read 10[r3] in cycle 4 (OF stage)
[1] will try to write in 20[r5] in cycle 4 (MA stage)
They both try to access memory at the same time(same cycle) as the MA unit can't satisfy two operations at a time one has to wait.
This doesn't happen in Simple Risc.

As these hazards cause results to be inaccurate and to overcome these we have to spend more cycles. These degrade the performance of processing.

Question 2:
No, RAR is not a hazard as Read After Read doesn't fetch incorrect values and if data is available it is going to be available for the next instruction. So RAR is a valid hazard.