

Question 1: Forwarding Answer the following considering the 5-stage RISC-V pipeline discussed in class. Fill out the pipeline diagram for the set of instructions. Additionally, what should ForwardA and ForwardB be set to in each cycle? If there is no instruction in the execute stage, set them to XX.

```

1 add x2, x3, x4
2 or x5, x2, x7
3 sub x8, x9, x10
4 add x3, x5, x8
    
```

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
add x2, x3, x4															
or x5, x2, x7															
sub x8, x9, x10															
add x3, x5, x8															
<u>ForwardA</u>															
<u>ForwardB</u>															

Question 2: Load Use Hazard Answer the following considering the 5-stage RISC-V pipeline discussed in class. Fill out the pipeline diagram for the set of instructions. Additionally, what should ForwardA and ForwardB be set to in each cycle? If there is no instruction in the execute stage, set them to XX.

```

1 lw x3, 0(x8)
2 add x11, x3, x3
3 lw x4, 0(x11)
4 add x9, x11, x9
5 sub x10, x4, x9
    
```

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
<u>lw</u> x3, 0(x8)															
add x11, x3, x3															
<u>lw</u> x4, 0(x11)															
add x9, x11, x9															
sub x10, x4, x9															
<u>ForwardA</u>															
<u>ForwardB</u>															

Question 3: Performance Analysis Assume a pipelined implementation where: Taken branches take 2 cycles. Loads and stores take 2 cycles. Assuming approximately 10% of instructions executed are branches, and of those 80% of the time they are taken, and 15% of instructions executed are loads or stores, what sort of real speed-up do we expect?