COMP311 In-Class Practice 11/11/26 Name: _____

Question 1: Pipeline Stalls Assume the five-stage pipeline without forwarding. Register writes occur before reads in the same cycle.

```
1 add x5, x3, x4
2 or x8, x9, x10
3 xor x14, x15, x16
4 sub x7, x5, x2
```

1.1. Does this program need any stalls?

1.2. Fill out the pipeline diagram for the set of instructions.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
add x3, x2, x2															
or x4, x2, x5															
and x9, x4, x10															
sub x11, x4, x14															
add x11, x15, x18															

1.3. What is the CPI?

1.4. What is the speedup compared to a single-cycle datapath?
Assume pipelined clock period = 220ps and single-cycle clock period = 820ps

Question 2: ForwardA and ForwardB For each `sub` in Execute, choose ForwardA and ForwardB. Use `00` = ID/EX register value, `10` = EX/MEM ALU result, and `01` = MEM/WB writeback value.

Instruction sequence	ForwardA	ForwardB
`add x5, x3, x4` `sub x7, x5, x2`		
`add x5, x3, x4` `sub x7, x2, x5`		

`add x5, x3, x4`
`or x8, x9, x10`
`sub x7, x5, x2`		

Question 3: Load Use Hazard Use the five-stage pipeline with forwarding. Register writes occur before reads in the same cycle; each memory access takes one cycle. Fill in the table using F, D, X, M, and W. Show stalls by repeating the held stage, and draw arrows for forwarding.
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.

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

Question 4: CPI with Hazards A program executes 100 instructions on the five-stage pipeline. There are 8 load-use stalls (1 cycle each) and 6 taken branches (2 penalty cycles each). Assume no other penalties. How many total cycles are needed? What is the CPI?

Question 5: Performance Analysis Assume a pipelined clock period of 1 ns and an ideal CPI of 1. Each taken branch adds 1 stall cycle, each load/store adds 1 stall cycle. Branches make up 10% of instructions, and 80% of branches are taken. Loads/stores make up 15% of instructions.

5.1. What is the actual average CPI?

5.2. What percentage of the ideal pipeline's performance does this achieve, assuming the same clock period?

5.3. An unpipelined processor takes 5 ns per instruction. What speedup does the actual pipeline provide over this processor?