1. Schedule the following instructions with NO FORWARDING on the MIPS 5-stage pipeline from the start of instruction A to the end of instruction I, mark the stages as F, D, X, M, W. Use the last column to indicate the number of stalls for each instruction.

Number of cycles =

#	Inst	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	Stall cycles
A	lw $1, 0($2)	**F**	**D**	**X**	**M**	**W**																					**0**
B	lw $3, 0($1)																										
C	add $4, $3, $9																										
D	add $5, $8, $9																										
E	lw $4, 0($8)																										
F	lw $9, 0($7)																										
G	add $10, $9, $11																										
H	sub $12, $4, $10																										
I	sw $12, 0($14)																										

2. Same as above but with FORWARDING .

Number of cycles =

#	Inst	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	Stall cycles
A	lw $1, 0($2)																										
B	lw $3, 0($1)																										
C	add $4, $3, $9																										
D	add $5, $8, $9																										
E	lw $4, 0($8)																										
F	lw $9, 0($7)																										
G	add $10, $9, $11																										
H	sub $12, $4, $10																										
I	sw $12, 0($14)																										

3. Assume the following instruction frequencies:

R-type	BEQ	JMP	LW	SW
40%	25%	5%	25%	5%

You are evaluating three branch prediction strategies: (1) predict always taken, (2) predict always not taken and (3) use the 2-bit branch predictor. Experimentally, your measurements show that 60% of all branches are taken. The 2-bit predictor has an accuracy of 85%.

Assuming that the penalty of a misprediction is 3 cycles (because the branch outcome is know in the EX stage) and 0 cycles for a correct prediction, compute the average stall cycles of the strategies.

prediction strategy	stall cycles if branch is actually
always taken	
always not taken	
2-bit predictor	

4. In the MIPS ISA, a branch instruction can be detected as such and its target known inn the ID stage. However, the condition (taken or not taken) is known only in the EX stage. For the two static strategies (always predict taken and always predict not taken) determine the number of stall cycles when the branch is actually taken or not taken. Show the execution schedules and write the stall cycles in the table below.

	stall cycle if branch is actually	
	taken	**not taken**
predict not taken		
predict taken		

predict not taken - branch not taken												
cycle	1	2	3	4	5	6	7	8	9	10	11	12
branch inst	F	D	X	M	W							
next inst												
target inst												

predict not taken - branch taken												
cycle	1	2	3	4	5	6	7	8	9	10	11	12
branch inst	F	D	X	M	W							
next inst												
target inst												

predict taken - branch not taken												
cycle	1	2	3	4	5	6	7	8	9	10	11	12
branch inst	F	D	X	M	W							
next inst												
target inst												

predict taken - branch taken												
cycle	1	2	3	4	5	6	7	8	9	10	11	12
branch inst	F	D	X	M	W							
next inst												
target inst												