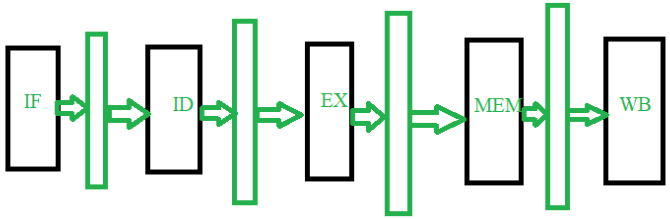
1.

For the following code, identify data hazard, structural hazard and control hazard, if any in ***consecutive instructions***.
Show the processing of following instructions on a 5-stage RISC processor (without data forwarding)
Show the processing of following instructions on a 5-stage RISC processor (with data forwarding). Compare above two cases.

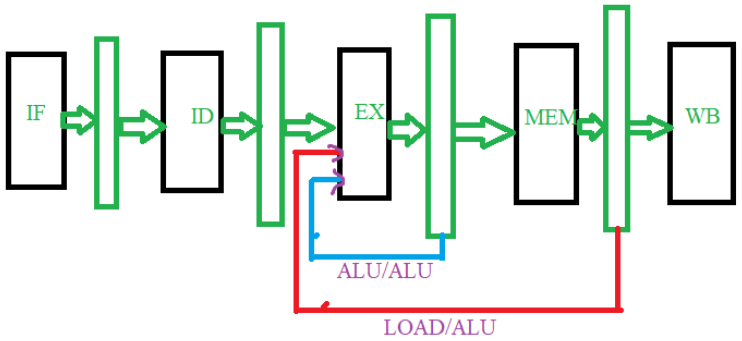
Inst-1: add r3, r1, r2 ; r3 destination
Inst-2: and r5, r3, r4 ; r5 result field ; true data dependency between inst-1 and inst-2
Inst-3: load r6, 24(r3) ; r6 destination
Inst-4: add r2, r6, r3 ; r2 result field ; true data dependency between inst-3 and inst-4
Inst-5: store r6, 12(r2) ; r6 source of data ; true data dependency between inst-4 and inst-5

without data forwarding



Instructions	1	2	3	4	5	6	7	8	9	10	11	12	13	1	15	16	17	18	
add r3, r1, r2	IF	ID	EX	ME	WB														
and r5, r3, r4		IF				ID	EX	MEM	WB										
load r6, 24(r3)						IF	ID	EX	MEM	WB									
add r2, r6, r3							IF				ID	EX	MEM	WB					
store r6, 12(r2)											IF				ID	EX	MEM	WB	

Show the processing of following instructions on a 5-stage RISC processor (with data forwarding). Compare above two cases.



Instructions	1	2	3	4	5	6	7	8	9	10	11	12	13	1	15	16	17	18	
add r3, r1, r2	IF	ID	EX	ME	WB														
and r5, r3, r4		IF	ID	EX	MEM	WB													
load r6, 24(r3)			IF	ID	EX	MEM	WB												
add r2, r6, r3				IF	ID		EX	MEM	WB										
store r6, 12(r2)					IF		ID	EX	MEM	WB									

2. For the following code, identify data hazard, structural hazard and control hazard, if any in consecutive instructions.

For the following code, identify data hazard, structural hazard and control hazard, if any in consecutive instructions.

Show the processing of following instructions on a 5-stage RISC processor (without data forwarding)

Show the processing of following instructions on a 5-stage RISC processor (with data forwarding). Compare above two cases.

load r6, 24(r3) ; r6 destination

add r3, r1, r6 ; r3 result field

And r5, r3, r4 ; r5 result field

add r2, r6, r5 ;r2 result field

store r6, 12(r2);

3.	For the code sequence below, state whether it must stall, can avoid stalls using only forwarding, or can execute without stalling or forwarding. (RISC processor)
	add R1,R0,1
	add R2,R0,2
	add R3,R0,2
	add R3,R0,4
	add R5,R0,5

4.	For the following code, identify data hazard, structural hazard and control hazard, if any in consecutive instructions. (RISC processor)
	ADD R1, R2, R3
	SUB R4, R1, R5
	AND R6, R1, R7
	OR R8, R1, R9
	XOR R10, R1, R11

5.	For the following code, identify data hazard, structural hazard and control hazard, if any in consecutive instructions. (RISC processor)
	ADD R1, R2, R3
	SUB R4, R1, R6

6.	For the following code, identify data hazard, structural hazard and control hazard, if any in consecutive instructions. (RISC processor)
	SUB R4, R1, R3
	ADD R1, R2, R3
	MUL R6, R1, R7

7.	Consider the following MIPS code that is being executed on the above 5-stage pipeline: Inst-1: lw $t1, 8($t0) Inst-2: addi $t2, $t1, 2 Inst-3: or $t3, $t1, $t2 Inst-4: lw $t4, -4($t3) Inst-5: sub $t5, $t3, $t4 Complete the following table showing the timing of the above code on the 5-stage pipeline MIPS processor (IF, ID, EX, MEM, WB). Draw an arrow showing forwarding between the stage that provides the data and the stage that receives the data. Show all stall cycles by placing an X in the box to represent a stall cycle
8.	Consider following part of an assembly–language program: Inst-1: LW $S5, 24($S3) ; $S5 = Mem[$S3+24] Inst-2: LW $S6, 8($S3) ; $S6 = Mem[$S3+8] Inst-3: MUL $S5, $S5, $S6 ; $S5 = $S5 * $S6 Inst-4: ADD $S3, $S3, $S2 ; $S3 = $S3 + $S2 Inst-5: SW $S5, 64($S3) ; Mem[$S3+64] = $S5

9.	Consider the following sequence of instructions being processed on the pipelined 5-stage MIPS RISC processor:
	Load R4, 100(R2) ; R4 is destination
	Add R5, R2, R3 ; R5 is result field
	Sub R6, R4, R5 ; R6 is result field
	And R7, R2, R5 ; R7 is result field
	
	Identify all the data dependencies in the above instruction sequence. For each dependency, indicate the two instructions and the register that causes the dependency.
	
	Assume that the pipeline does not use operand forwarding. Also assume that the only sources of pipeline stalls are the data hazards. Draw a diagram that represents instruction flow through the pipeline during each clock cycle. How long does it take for the instruction sequence to complete?
	
	Now, assume that the pipeline uses operand forwarding. There are separate forwarding paths from the outputs of stage-3 and stage-4 to the input of stage-3. Draw a diagram that represents the flow of instructions through the pipeline during each clock cycle. Indicate operand forwarding by arrows.

10.	For all following questions we assume that: Pipeline contains 5 stages: IF, ID, EX, M and W; Following program segment should be processed: NSU: LW R1, 0 (R4) ; Load first operand LW R2, 400 (R4) ; Load second operand ADD R3, R1, R2 ; Add operands SW R3, 0 (R4) ; Store result JUMP NSU Calculate how many clock cycles will take execution of this segment on the simple pipeline without forwarding or bypassing when the loop is repeated twice. Show timing of one loop cycle in Figure: Calculate how many clock cycles will take execution of this segment on the simple pipeline with forwarding or bypassing when the loop is repeated twice. Show timing of one loop cycle in Figure:
11.	Assume a machine using Five-stage pipelining (IF – ID — EX – MU- WR) runs a program. Instruction-3 is a Call instruction. CPU calls a function that starts at instruction 8 and ends at Instruction-11. Please note that Instruction-11 is a return instruction, means that the program control returns to immediately next to the Call instruction. Show at least 15-time steps of pipelining stages.

12.	The following program is to be run on a RISC pipeline processor of form IF-ID-EX-MA-WB. Compute the pipeline CPI: Using pipeline interlocks Using optimal pipeline forwarding from EX or MA stages to any other stage Inst-1: LOAD R2, 60(R1) Inst-2: LOAD R1, 40(R2) Inst-3: ADD R1, R1, R2 Inst-4: STORE R1, 20(R2)

13.	Show the processing of following instructions using a 5-stage pipelined processor having Fetch unit-Decode unit-Memory/operand read unit-Execution unit and Result store unit. Use a table to show the execution Consider a single port RAM is used with CPU
	ADD R1, R2, R3
	SUB R4, R1, R3
	SUB R5, R6, R4
	BNEQZ R5, NSU-1 ; assume the condition is evaluated True
	AND R1, R2, R3
	OR R4, R3, R6
	NSU-1: LOAD R2, M1 ; M1 IS A MEMORY LOCATION
	LOAD R3, M2 ; M2 IS A MEMORY LOCATION
	ADD R1, R2, R3
	SUB R4, R1, R2
	STORE M3, R4 ; M3 IS A MEMORY LOCATION
	Here is the expected solution of the problem as stated above.
	DATA DEPENDENCY with a single-port RAM; CPU can only read/store one data from/to RAM at a time.