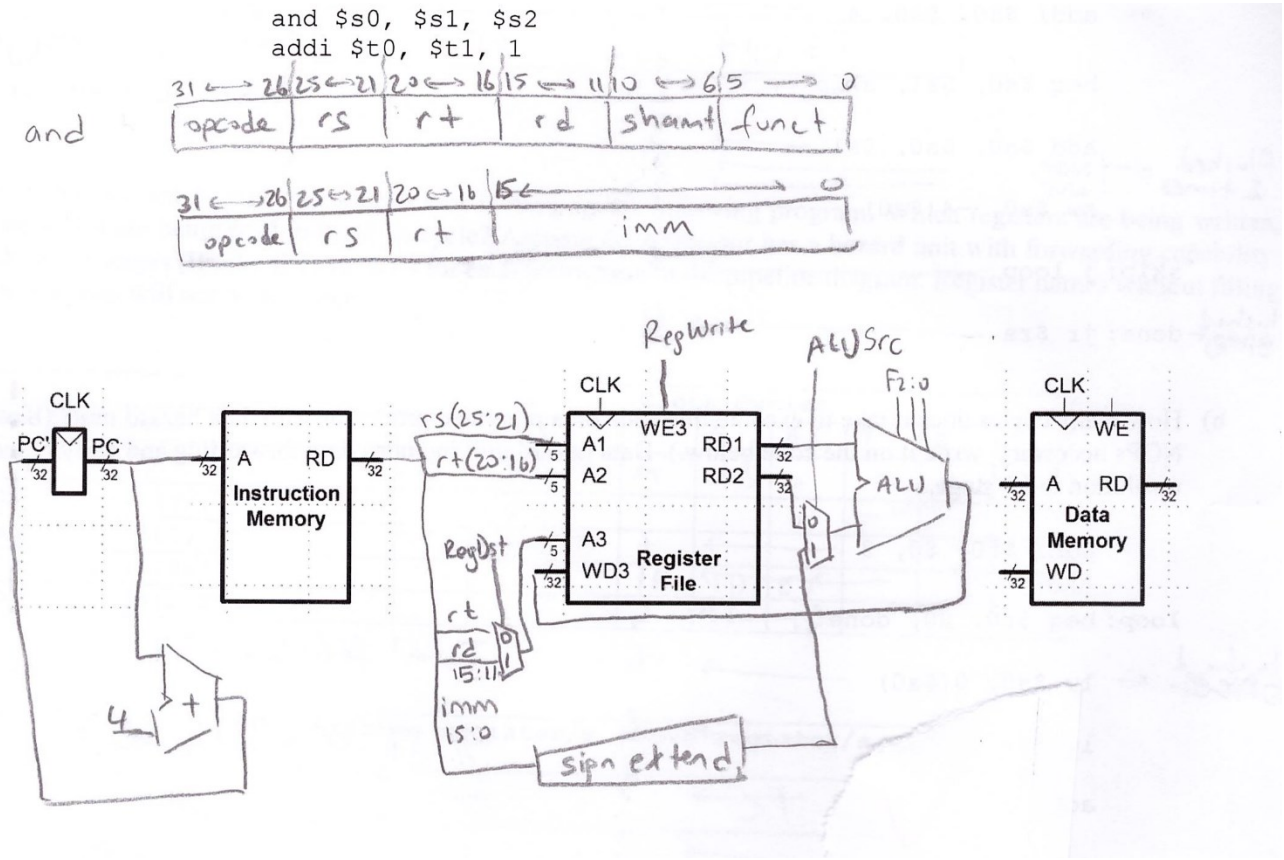
Name-Last Name: _____ **Student ID:** _____

Hacettepe University	Computer Engineering Department
BBM234 Computer Organization	Instructors: Assoc. Prof. Dr. Suleyman TOSUN
Midterm Exam 2	
Duration: 100 minutes	Exam Date: 03.05.2018

Questions	1	2	3	4	Total
Marks	30	30	20	20	100
Earned					

Q1. You have 32-bit Program Counter (PC), instruction memory, register file and a data memory.

Draw a single-cycle processor that can execute ONLY the following type of instructions. Define necessary control signals and write their values in the table. Note that you do not need some of the control signals in the table. Write NA to them if they are not used in your microarchitecture.



Tabel I: Control signals for **and** and **addi**

Inst.	$Op_{31:26}$	RegWrite	RegDst	AluSrc	Branch	MemWrite	MemtoReg	ALUOp
and	100100	1	1	0	NA	NA	NA	10 (or 1)
addi	001000	1	0	1	NA	NA	NA	00 (or 0)

Q2. You are given the following MIPS code. In the program, register `$a0` initially has the address of first array element. Array has the following data: **[5,5,8,8].**

a) How many cycles does it take to execute this code on a 5-stage pipelined processor without data hazard unit (no forwarding and no early branch resolution)? If any NOPs necessary, write it on the code below. Assume processor can flush wrong-fetched instructions when the branch or jump is resolved.
Hint: First, insert NOPs if necessary and then determine how many times each instruction is fetched.

	Instructions	Number of fetches	Explanation
	addi $t0, $0, 3	1	
	NOP	1	Data hazard on t0
	NOP	1	Data hazard on t0
loop:	beq $t0, $0, done	4	Loop iterates 3 times. Plus the last check.
	lw $s0, 0($a0)	4	Flushes at last fetch
	lw $s1, 4($a0)	4	Flushes at last fetch
	addi $t0, $t0, -1	4	Flushes at last fetch
	addi $a0, $a0, 4	3	
	beq $s0, $s1, skip	3	
	add $s0, $s0, $s1	3	Flushed twice.
	NOP	3	Data hazard on s0. Flushed twice.
	NOP	3	Data hazard on s0. Flushed twice.
	sw $s0, -4($a0)	1	
	skip: j loop	3	
done:	jr $ra	4	Flushed three times.
	Total fetches:	42	

Jr takes 4 more cc to finish its execution. Total cc= 42+4 =46 cc

b) How many cycles does it take to execute this code on a pipelined processor with data hazard unit? (If any NOPs necessary, write it on the code below.). Data hazard unit includes data forwarding and early branch resolution in ID stage.

	Instructions	Number of fetches	Explanation
	addi $t0, $0, 3	1	
	NOP	1	Data hazard on t0
loop:	beq $t0, $0, done	4	Loop iterates 3 times. Plus the last check.
	lw $s0, 0($a0)	4	Flushes at last fetch
	lw $s1, 4($a0)	3	
	addi $t0, $t0, -1	3	
	addi $a0, $a0, 4	3	
	beq $s0, $s1, skip	3	
	add $s0, $s0, $s1	3	Flushed twice.
	sw $s0, -4($a0)	1	
	skip: j loop	3	
done:	jr $ra	4	Flushed three times.
	Total fetches:	33	

Jr takes 4 more cc to finish its execution. Total cc= 33+4 =37 cc

Q3. a) The pipelined 5-stage MIPS processor is running the following program. Which registers are being written, and which are being read in the fifth cycle? Assume the processor has a hazard unit with forwarding capability. Show the stages (IF, ID, EX, M, WB) for each instruction in the pipeline diagram. Register names without filling the diagram will not be accepted.

	Clock Cycles								
Instructions	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
add $s0, $t0, $t1	IF	ID	EX	M	WB				
sub $s1, $t2, $t3		IF	ID	EX	M	WB			
and $s2, $s0, $s1			IF	ID	EX	M	WB		
or $s3, $t4, $t5				IF	ID	EX	M	WB	
slt $s4, $s2, $s3					IF	ID	EX	M	WB

Written register/s	**Read register/s**
s0 (first ins. Writes to s0 in WB stage)	t4 and t5 (or reads at ID stage)

b) The pipelined 5-stage MIPS processor is running the following program. Which registers are being written, and which are being read in the fifth cycle? Assume the processor has a hazard unit with forwarding capability. Show the stages (IF, ID, EX, M, WB) for each instruction in the pipeline diagram. Register names without filling the diagram will not be accepted.

	Clock Cycles								
Instructions	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
sw $t5, 72($t0)	IF	ID	EX	M	WB				
addi $s1, $s2, 5		IF	ID	EX	M	WB			
sub $t0, $t1, $t2			IF	ID	EX	M	WB		
lw $t3, 15($s1)				IF	ID	EX	M	WB	
or $t2, $s4, $s5					IF	ID	EX	M	WB

Written register/s	**Read register/s**
None (sw writes to memory in M stage. Nothing is written in the registers in WB stage)	s1 (lw reads s1 at ID stage to calculate the memory address)

Q4. You are given the following MIPS code running on a **single-cycle MIPS processor**. **Assume MemtoReg control signal has *stuck-at-zero* fault**, meaning that signal is always zero.

```
        lw $s0, 0($0)
        lw $s1, 4($0)
        addi $s3, $0, 5
        beq $s0, $s1, done
        sll $s3, $s3, 2
        addi $s3, $s3, 1
Done:   add $v0, $s3, $0
```

Memory

Address	Data
0x0000000C	45
0x00000008	35
0x00000004	25
0x00000000	15

a) In the following table, write down which instructions will malfunction. In other words, which instructions will not execute correctly? Write the registers that will be written by these instructions, their written values, and their expected values.

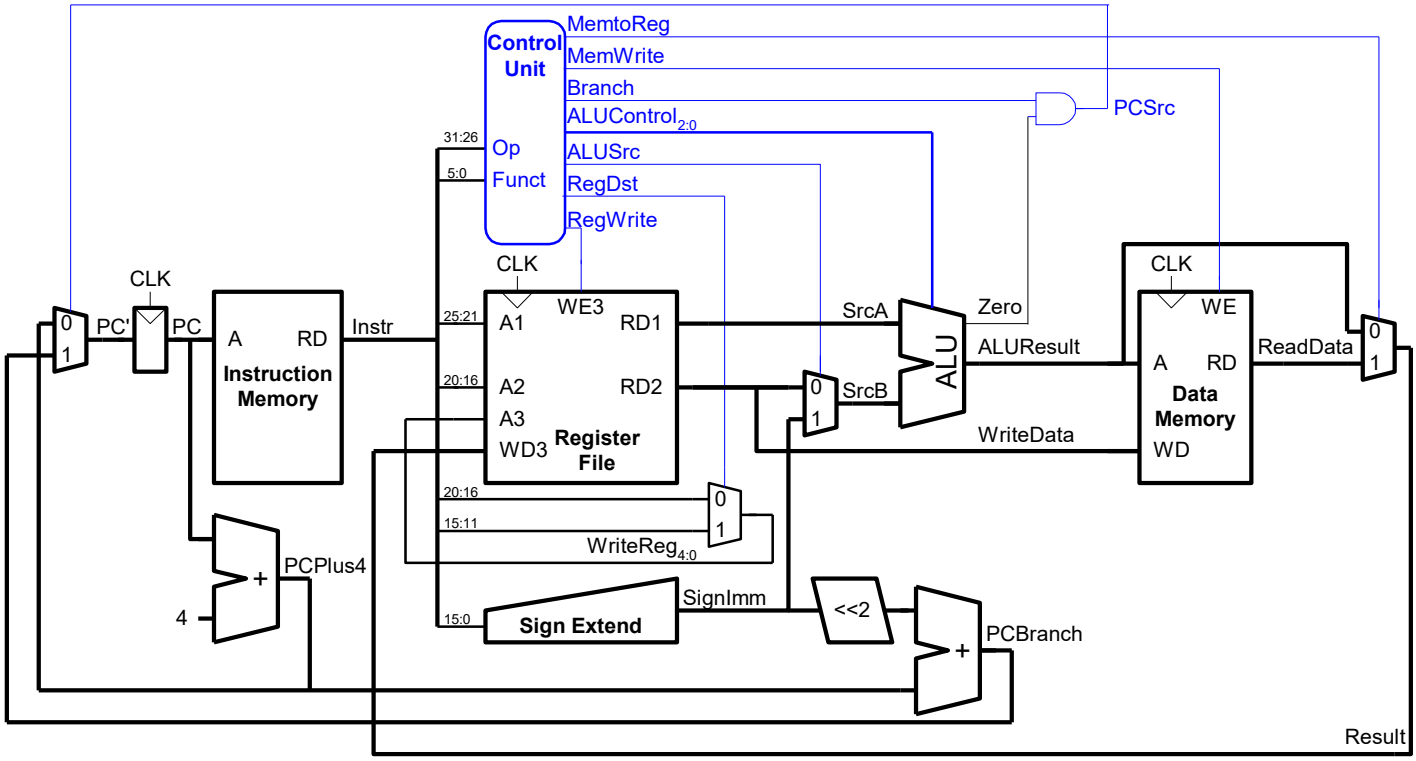
Malfunctioning instructions	**Registers written**	**Written value**	**Expected value**
lw $s0, 0($0)	s0	0	15
lw $s1, 0($0)	s1	4	25

Since lw cannot write the value from memory to register file, it writes the output of ALU, which is the calculated address.

b)

What value is returned by this function in v0 register?	**Is the returned result expected/correct value?**	**If no, what is the correct value?**
21	YES	-

Beq compares s1 and s2. We were expecting 15 and 25 in them, which are not equal. However, the written values, 0 and 4, are also not equal. Therefore, the code executes as expected although lw instructions malfunction.



Single-cycle MIPS processor