Name: _____

Problem 1 (30%). Show each step of the pipeline machine (page 470 and 469) for the following instruction sequence:

Assume $1=9, $2=8; Mem[12]=16. Treat the "nop" instruction as "add $0,$0,$0".

```
sw    $1, 8($2)
lw    $1, 4($2)
nop
nop
```

Clock	<IF/ID> <PC, IR>	<ID/EX> <WB,M,EX,PC,A,B,S,Rt,Rd>	<EX/MEM> <WB,M,PC,Z, ALU, B, R>	<MEM/WB> <WB,MDR,ALU,R>
0	<0,?>	<?,?,?,?,?,?,?,?,?>	<?,?,?,?,?,?,?>	
1	**<4,"sw $1,8($2)">** observe that sw $rt=$1,8($rs=$2)	<?,?,?,?,?,?,?,?,?>	<?,?,?,?,?,?,?>	
2	**<8,"lw $1,4($2)">** observe that lw $rt=$1,4($rs=$2)	**<0X, 001, X001, 4, 8, 9, 8, $1,X>** observe that <wb=1X, m=001, ex=X001, pc=4, $rs→$2→8→A, $rt→$1→9→B, S=4, $rt=$1, $rd=X>	<?,?,?,?,?,?,?>	<?,?,?,?,?>
3	**<12, "nop">** observe that add $rd=0,$rs=0,$rt=0	**<11, 010, 0001, 8, 8, 9, 4, $1,X>** observe that <wb=11 ,m=010, ex=0001, pc=8, $rs→$2→8→A, $rt→$1→9→B, S=4, $rt=$1, $rd=X>	**<0X, 001, X or 36, X or 0, 16, 9, $1>** observe that <wb=1X, m=001, pc=pc+(S<<2)=4+(8<<2)=36, Z=0, **ALU=S+$2=8+8=16,** 9, $1> **Store to memory** **Mem[ALUOut] ← B** **Mem[16] ← 9**	<?,?,?,?,?>
4	**<16, "nop">** observe that add $rd=0,$rs=0,$rt=0	**<10, 000, 1100, 12, 0, 0, 0, $0, $0>** observe that <wb=10, m=000, ex=1100, pc=12, $rs=$0=0=A, $rt=$0=0=B, S=0, $rt=$0, $rd=$0>	**<11, 010, X or 24, X or 0, 12, 9, $1>** observe that <wb=11, m=010, pc=pc+(S<<2)=8+(4<<2)=24, Z=0, **ALU=S+$2=4+8=12**, 9, $1> **Load from memory** Mem[12]→16	**<0X, X, X or 16, X or $1>** **Nothing happens here!** observe that <wb=0X, mdr=X, ALU=16, R=$1>
5	**<20, "nop">** observe that add $rd=0,$rs=0,$rt=0	**<10, 000, 1100, 16, 0, 0, 0, $0, $0>** observe that <wb=10, m=000, ex=1100, pc=16, $rs=$0=0=A, $rt=$0=0=B, S=0, $rt=$0, $rd=$0>	**<10,000,X,0,0,0,$0>** observe that <wb=10, m=000, pc=X, Z=0, ALU=A+B=0+0=0, B=0, R=$0>	**<11, 16, 12, $1>** =>Reg[$1]←16 **finally register 1 will contain the load memory word of 4 + $12** observe that <wb=11, mdr=Mem[ALU] =Mem[12]=16, ALU=12, R=$1>

Both solutions ok
X = do not care
or
36

Problem 2 (50%). Assume a simple 5 stage pipeline with the following execution times

1	IF	Instruction fetch	3 ns
2	ID	Register Read	1 ns
3	EX	ALU	3 ns; Branch decision made here
4	MEM	Data Access	Memory Write time=5 ns, Read time=4 ns
5	WB	Register Write	2 ns

This computer has the following instructions:

Instruction		Operation
add	$rd, $rs, $rt	$rd = $rs + $rt
lw	$rt, addr16($rs)	$rt = Mem[addr16+$rs]
sw	$rt, addr16($rs)	Mem[addr16+$rs]=$rt
beq	$rs, $rt, disp16	pc = pc+2+($rs-$rt=0?disp16:0)

2a (10%) Fill in the following tables

Instruction	IM	ID	EX	MEM	WB	Total Time	Multi-Cycles	Instruction Mix
add	**3 ns**	**1 ns**	**3ns**		**2 ns**	**9 ns**	**4**	50%
lw	**3 ns**	**1 ns**	**3 ns @+$rs**	**4 ns**	**2 ns**	**13 ns**	**5**	20%
sw	**3 ns**	**1 ns**	**3 ns @+$rs**	**5 ns**		**12 ns**	**4**	10%
beq	**3 ns**	**1 ns**	**3 ns**			**7 ns**	**3**	20%

2b (2%) Fill the following table and show work.

Instruction	Clock frequency	CPI	MIPS
Single-cycle CPU	1/13 ns **= 76.9 MHz**	**=1**	76.9 MHz/1 **= 76.9 MIPS**
multi-cycle CPU	1 / 5 ns **= 200 MHz**	4*50%+5*20%+4*10%+3*20% **= 4**	200 MHz/4 **= 50 MIPS**

The following parts refer to the pipelined machine only

For the following code: Assume no forwarding.

2c (10%) Draw lines showing all the data dependencies in column 1.
and show the pipeline sequence (IF, ID, EX, M, WB) for the following code

Time	1	2	3	4	5	6	7	8	9	10	11	12
lw **$1**, 4($2)	**IF**	**ID**	**EX**	**M**	**WB**							
lw $3, 8(**$1**)		**IF**	**ID**	**ID**	**ID**	**EX**	**M**	**WB**				
sub $4, **$1**, $2			**IF**	**IF**	**IF**	**ID**	**EX**	**M**	**WB**			
add $5, $4, $2						**IF**	**ID**	**ID**	**ID**	**EX**	**M**	**WB**

2d (2%) How many clock cycles will a load take if there is a dependency?
Latency instruction time = 5+2 stalls = 7 or Pipeline or Throughput clocks = 1+ 2 stalls = 3

2e (2%) How many clock cycles will a load take if there is no dependency?
Latency instruction time = 5+0 stalls = 5 or Pipeline or Throughput clocks = 1 + 0 stalls = 1

Solution #1 **Solution #2**

2f (10%) Using forwarding, show the pipeline sequence (IF, ID, EX, M, WB) and draw lines showing the forwarding.

Time	1	2	3	4	5	6	7	8	9	10	11	12
lw $1, 4($2)	IF	ID	EX	M	WB							
lw $3, 8($1)		IF	ID	ID	EX	M	WB					
sub $4, $1, $2			IF	IF	ID	EX	M	WB				
add $5, $4, $2					IF	ID	EX	M	WB			

2g (10%) For the following code: Assume no forwarding and no branch prediction. Draw lines showing all the data dependencies and show the pipeline sequence (IF, ID, EX, M, WB) and draw lines showing the forwarding. **Note:** Branch decision is made in the EX stage.

Time	1	2	3	4	5	6	7	8	9	10	11	12
beq $1, $2, loop	IF	ID	EX	M	WB							
sub $3, $4, $5				IF	ID	EX	M	WB				
add $4, $4, $5					IF	ID	EX	M	WB			

No branch prediction: On IF stage: the Branch IF/ID stops fetching next instruction.

2h (2%) What is the pipeline clock? **= 1/5ns = 200 MHz (slowest pipeline stage)**

2i (2%) What is the pipelined speed up from a single-cycle processor?

speedup = single-cycle clock / pipelined clock = 13 ns / 5 ns = 2.6

Problem 3 (20%). Show all calculations for the following questions.

Assume a load word takes 2 cycle if no dependancy and if dependant then 5 clocks
Assume there is a 20% data dependency.

Assume a branch takes 2 cycle if true prediction and if false prediction then 5 clocks.
Assume that 25% of the branches are mispredicted.

3a (5%) What is the average load instruction time in clocks?

= 80% × (2 clocks) + 20% × (5 clocks) = 2.6 clocks

3b (5% What is the average branch instruction time in clocks?

= 75% × (2 clocks) + 25% × (5 clocks) = 2.75 clocks

3c (10%) Now fill in the table and show calculations.

Instruction	Pipeline Cycles	Instruction Mix
add	1	50%
lw	**2.6**	20%
sw	1	10%
beq	**2.75**	20%
Clock	300 Mhz	
CPI	**= 1× 50% + 2.6 × 20% + 1 × 10% + 2.75 × 20% = 1.67**	
MIPs	**300 Mhz/ 1.67 = 179.6 MIPS**	