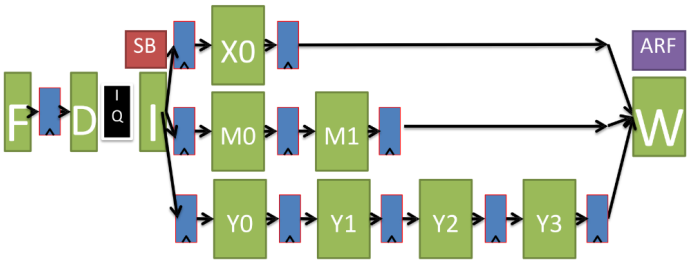
Formal Quiz-2 Rubric

Q1. Assume IO3 architecture with full bypassing and with in-order fetch, decode, out-of-order issue, out-of-order execute, and out-of-order writeback. The pipeline has 4 functional units: ALU (i.e. add, sub, beqz) (1 cycle X0), Loads-Stores (2 cycles, M0 and M1) and multiplies (4 cycles, Y0,Y1,Y2,Y3). **[15 Marks]**



Show the pipeline diagram for the following code.
Sub R9, R11, R6
ld R5, R9
mul R2, R3,R5
sub R14, R4, R4
mul R10,R1,R14
beqz R10, target
add R6, R7, R8
mul R9,R6, R5
ld R11,R12
sub R11,R12,R13
addi R1, R11,1
mul R1,R2,R3
add R4,R6,R7
target: mul R4, R5, R6

Ans:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
Sub R9, R11, R6	F	D	I	X0	W																
ld R5, R9		F	D	I	M0	M1	W														
mul R2, R3,R5			F	D	i	I	Y0	Y1	Y2	Y3	W										
sub R14, R4, R4				F	D	i	I	X0	W												
mul R10,R1,R14					F	D	i	I	Y0	Y1	Y2	Y3	W								
beqz R10, target						F	D	i				I	X0	W							
add R6, R7, R8							F	D	I	X0	X0	W									
mul R9,R6, R5								F	D	I	Y0	Y1	Y2	-							
ld R11,R12									F	D	I	M0	M1	-							
sub R11,R12,R13										F	D	I	X0	-							
addi R1, R11,1											F	D	i	-							
mul R1,R2,R3												F	D	-							
add R4,R6,R7													F	-							
target: mul R4, R5, R6														F	D	I	Y0	Y1	Y2	Y3	W

(1*5 + 2*1 + 1*8)

Grading:

1. In case commit stage is used in the pipeline, only half of the total marks will be awarded.
2. When the net pipeline is incorrect, full marks will be awarded till it is correct. For the remaining correct pipeline, only half of the remaining marks will be awarded.

Q2. Consider the following high-level language program, which is compiled into the following assembly language program.

High-level language

```
while (i < N)
{
        A[i] = B[i] * i;
        i++;
}
```

Assembly language:

R1 stores the base address of array B, and R2 stores the base address of array A

```
loop:   lw F1, 0(R1)
        addiu R1, R1, 4
        mul.s F2, R3, F1
        sw F2, 0(R2)
        addiu R2, R2, 4
        addiu R3, R3,1
        bne R3, R4,loop
```

a. Consider a VLIW EQ compiler with the processor having one integer unit, one floating point unit and one load/store unit. *addiu* takes one clock cycle, *mul.s* (floating point multiplication) takes three clock cycles, and *lw* and *sw* take four clock cycles. Assume branch instruction is executed in integer units. Assume the compiler does not reorder the instructions.Pack the instructions for the above program and determine the Floating Point Operation per second (FLOPS).
   [10 Marks]
b. Assuming 16 integer registers and 16 floating point registers, unroll the loop to perform three iterations at once. **[10 Marks]**

Ans.

a.

Clock Cycle	Integer Unit	Floating point	Load/Store Unit
1			lw F1, 0(R1)
2	addiu R1, R1, 4		
3			
4			
5		mul.s F2, F3, F1	
6			
7			

8			sw F2, 0(R2)
9	addiu R2, R2, 4		
10	addiu R3, R3,1		
11	bne R3, R4, loop		

1 + 1 + 2 + 2 + 1 + 1 + 1 = 9 Marks
(No mark will be awarded if the dependent instruction is scheduled before the previous one finishes execution)

FLOPS = 1/11 = 0.09 **[1 Mark]**

Note: All the instruction packing till 14 clock cycles are permissible. One mark will be deducted for each clock cycle exceeding it.

b.

```
lw F1, 0(R1)
lw F2, 4(R1)
lw F3, 8(R1)
addiu R1, R1, 12
mul.s F4, R3, F1
addiu R5, R3,1
mul.s F5, R5, F2
addiu R6, R3,2
mul.s F6, R6, F3
sw F4, 0(R2)
sw F5, 4(R2)
sw F6, 8(R2)
addiu R2, R2, 12
addiu R3, R3, 3
bne R3, R4, loop
```

(0.6*15 = 9 Marks)
(1 Mark for correct array increment)

Note: Different registers from the one specified can be used. However, the instructions should not be reordered.

Q3. There are 16 physical registers (p1-p16). Initially, the architectural registers R1 to R8 are mapped to physical registers p1 to p8. Rewrite the code below so that only true data dependencies remain. **[5 Marks]**

```
ADD   R1,R2, R3
ST    R1, R4
ADDi  R2, R5, 6
LD    R5, R6
ADD   R6, R7, R2
```

Ans:
Note: Any of the register can be used as long as RAW hazard is not removed.

```
ADD   p9, p2, p3
ST    p9, p4
ADDi  p10, p5, 6
LD    p11, p6
ADD   p12, p7, p10
```

(1*5 = 5 Marks)
1 mark deducted if RAW hazard removed