2. Register Renaming [50 points]

In this problem, we will give you the state of the Register Alias Table (RAT), Reservation Stations (RS), and Physical Register File (PRF) for a Tomasulo-like out-of-order execution engine.

The out-of-order machine in this problem has the following characteristics:

- The processor is fully pipelined with four stages: Fetch, decode, execute, and writeback.
- For all instructions, fetch takes 1 cycle, decode takes 1 cycle, and writeback takes 1 cycle.
- The processor implements `ADD` and `MUL` instructions only. Both the adder and multiplier are fully pipelined. `ADD` instructions take 3 cycles and `MUL` instructions take 4 cycles in the execute stage. Note that the adder and multiplier have separate common data buses (CDBs), which allow both the adder and multiplier to broadcast results in the same cycle.
- An instruction always allocates the first reservation station that is available (in top-to-bottom order) at the required functional unit.

Suppose the pipeline is initially empty and the machine fetches exactly 5 instructions. The diagram below shows the snapshot of the machine at a particular point in time.

Register Alias Table

ID	V	Tag
R0	1	P1
R1	1	P8
R2	1	P12
R3	1	P4
R4	0	P7
R5	1	P5
R6	0	P11
R7	1	P14

Physical Register File

ID	V	Data	ID	V	Data
P0	0	2	P8	1	88
P1	1	11	P9	0	90
P2	0	2	P10	0	91
P3	0	30	P11	1	110
P4	1	3	P12	1	33
P5	1	50	P13	1	130
P6	0	5	P14	1	17
P7	1	70	P15	1	159

`ADD` **Reservation Station**

ID	V	Tag	V	Tag	Dest. Tag
A	1	P12	1	P7	P15
B	1	P5	0	P13	P11

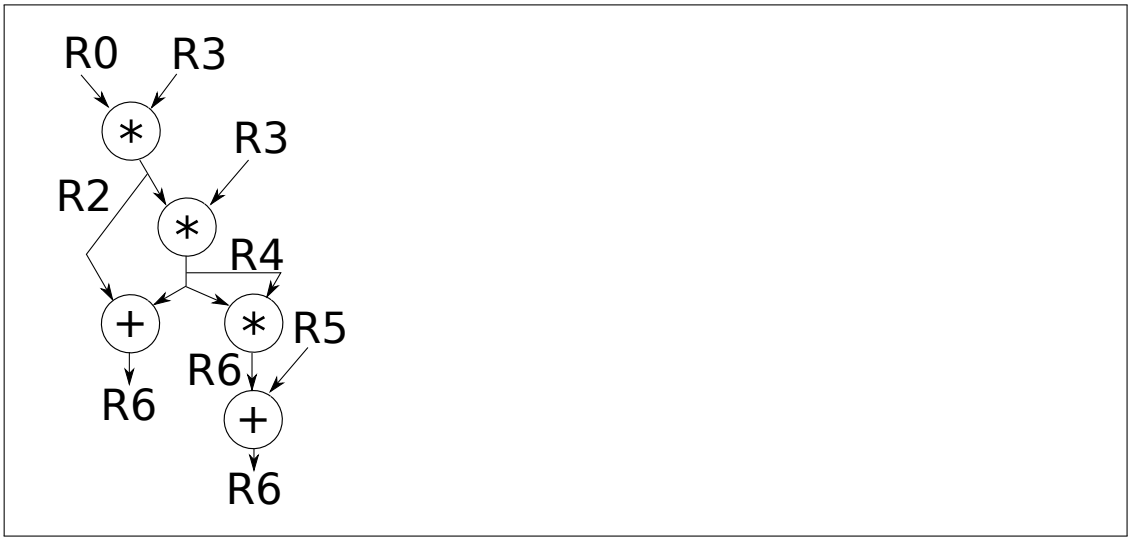
`ADD` **CDB**

`MUL` **Reservation Station**

ID	V	Tag	V	Tag	Dest. Tag
X	0	–	0	–	–
Y	1	P12	1	P4	P7
Z	1	P7	1	P7	P13

`MUL` **CDB**

(a) Your first task is to use only the supplied information to draw the data flow graph for the five instructions which have been fetched. Label nodes with the operation (+ or $\star$) being performed and edges with the architectural register alias numbers (e.g., R0).



(b) Now, use the data flow graph to fill in the table below with the five instructions being executed on the processor in program order. The source registers can be specified in either order. Give instructions in the following format: "opcode, source1, source2, *destination*."

OP	Src 1	Src 2	Dest
MUL	R0	R3	R2
MUL	R2	R3	R4
ADD	R2	R4	R6
MUL	R4	R4	R6
ADD	R5	R6	R6

(c) Now, show the full pipeline timing diagram below for the sequence of five instructions that you determined above, from the fetch of the first instruction to the writeback of the last instruction. Assume that the machine stops fetching instructions after the fifth instruction.

As we saw in class, use F for fetch, D for decode, En to signify the nth cycle of execution for an instruction, and W to signify writeback. You may or may not need all columns shown. Finally, identify the cycle after which the snapshot of the microarchitecture was taken. Shade the corresponding cycle in the last row of the table.

Cycle:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
Instruction 1	F	D	E1	E2	E3	E4	W												
Instruction 2		F	D				E1	E2	E3	E4	W								
Instruction 3			F	D							E1	E2	E3	W					
Instruction 4				F	D						E1	E2	E3	E4	W				
Instruction 5					F	D									E1	E2	E3	W	
Snapshot cycle										X									