

- 1) (POINTS 14/40) Consider the following snippet of code running on a single-dispatch using Tomasulo's algorithm to perform the dynamic scheduling of instructions. The program performs a search inside a vector. Initially, R1 = 0.

```

etic:  LW    R2, 0(R1)    ; read Xi
        ADDI  R2, R2, 1    ; increment R1
        SW    R2, 0(R1)    ; store Yi
        ADDI  R1, R1, 4    ; update R1
        BNE  R2, R0, etic ; continue to loop if false
    
```

Working hypothesis:

- the loop executes speculatively in terms of direction (always taken) but not regarding the branch condition; **case A) no-speculation on branch condition; case B) speculation on branch condition;**
- the issue stage (I) calculates the address of the actual reads and writes;
- reads require 1 clock cycles; writes take 0 cycles (they are written in a write-buffer + split-cache)
- **when accessing memory (M), writes have precedence over reads and must be executed in-order**
- there's only one CDB
- dispatch stage (P) and complete stage (W) require 1 clock cycle
- there are separated integer units for the calculation of the actual address, for arithmetic and logical operations, for the evaluation of the branch condition
- the functional units do not take advantage of pipelining techniques **internally (reservation stations are busy until the end of CDB-write, except for Stores)**
- the load queue has 5 slots; the queue buffer has 5 slots (writes wait for the operand in the queue buffer, i.e., in the execution stage)
- the rest of the processor and has the following characteristics

Type of Functional Unit	No. of Functional Units	Cycles for stage I	No. of reservation stations
Integer (effective addr.)	1	1	2
Integer (op. A-L)	1	1	2
Integer (branch calc.)	1	1	2

Complete the following chart until the end of the third iteration of the code fragment above, both in the case of no speculation on branch condition (case A) that in the case of speculation on branch condition (case B).

Iteraz.	Instruction	P: Dispatch (clock)	I+X: Issue+Exec (start-stop)	M: MEM ACCESS (clock)	W: CDB-write (clock)	C: Complete (clock)	Commenti
1	LD R2, 0(R1)						
...	...						
...	...						

- 2) Given the sequence P1: R, P2: R, P3: R, P1: W, P2: W, P3: W (Px:R = read by the processor Px, Px:W write by the processor Px), with respect to a certain variable 'a', show for each processor the sequence of states, and transactions on the bus that occur in a multiprocessor UMA with write-back caches for each processor and PSCR coherence protocol.