CSE/ECE 511
EndSem Solution

Q1

RAM AL = 200
With L1 AL = 10 (hit)
With L1 AL = 210 (Miss)
Let miss rate be m
AMAT = hitrate*hitlatency + missrate*misslatency
= (1-m)*10 + m*(210)

AMAT <= 200 (for cache to be useful)
m <= 19/20

Q2

The number of sets in the cache = (16 * 2^10) /(2*32) = 256
Since a word size is 4 bytes, int is word-sized and the size of a cache block is 32 bytes, the number of ints that would fit in a cache block is 8.

In the first loop:

Every miss followed by 3 successive hits and then this will go till i =1022 .
i = 0 ⇒access a[0] - compulsory miss
I = 2 ⇒ access a[2] - hit
i = 4 ⇒ access a[4] - hit
i = 6 ⇒ access a[6] - hit

i = 8 ⇒ access a[8] - Compulsory miss
……..
……..
……..
……..

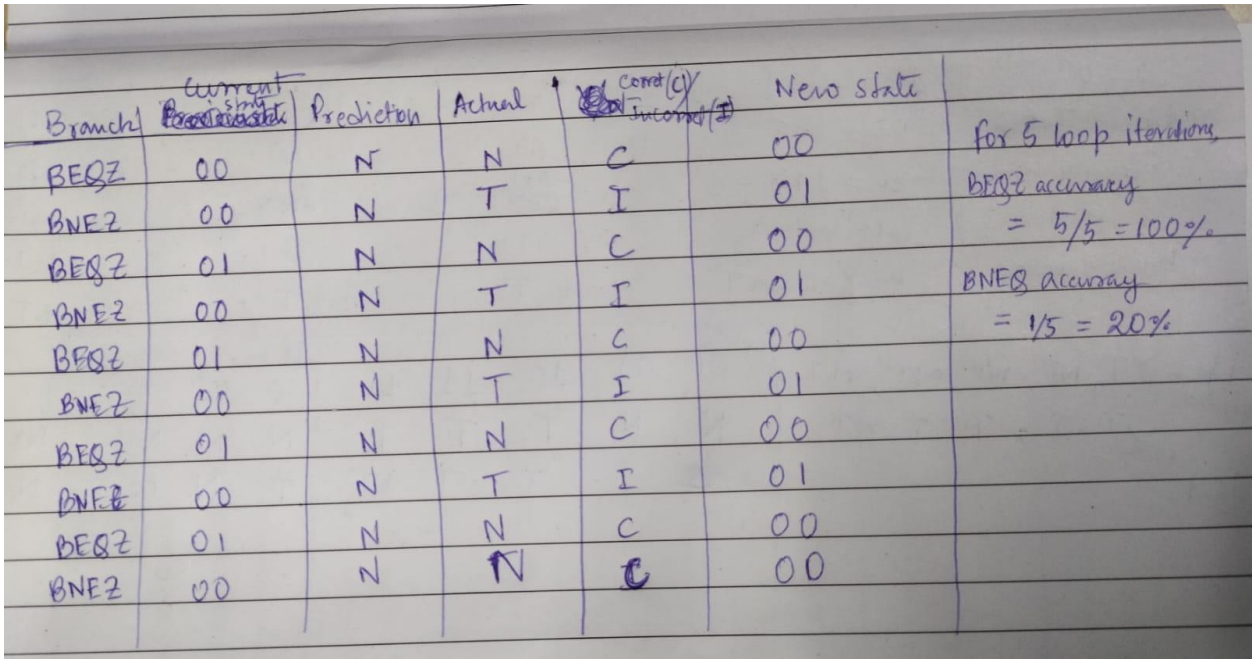
Therefore total miss in the first loop = 128
And hits = 384

In Second Loop:

Access to a[0] will result in a hit because it is still in cache.
Access to a[1024] will result in a compulsory miss and will map to one of the cache lines of set 128 access to a[1024*2] will result in a compulsory miss and will map to one of the cache lines of set 0 and this will carry on.
Therefore the total number of misses in the second loop will be 1024 - 1 and hit = 1

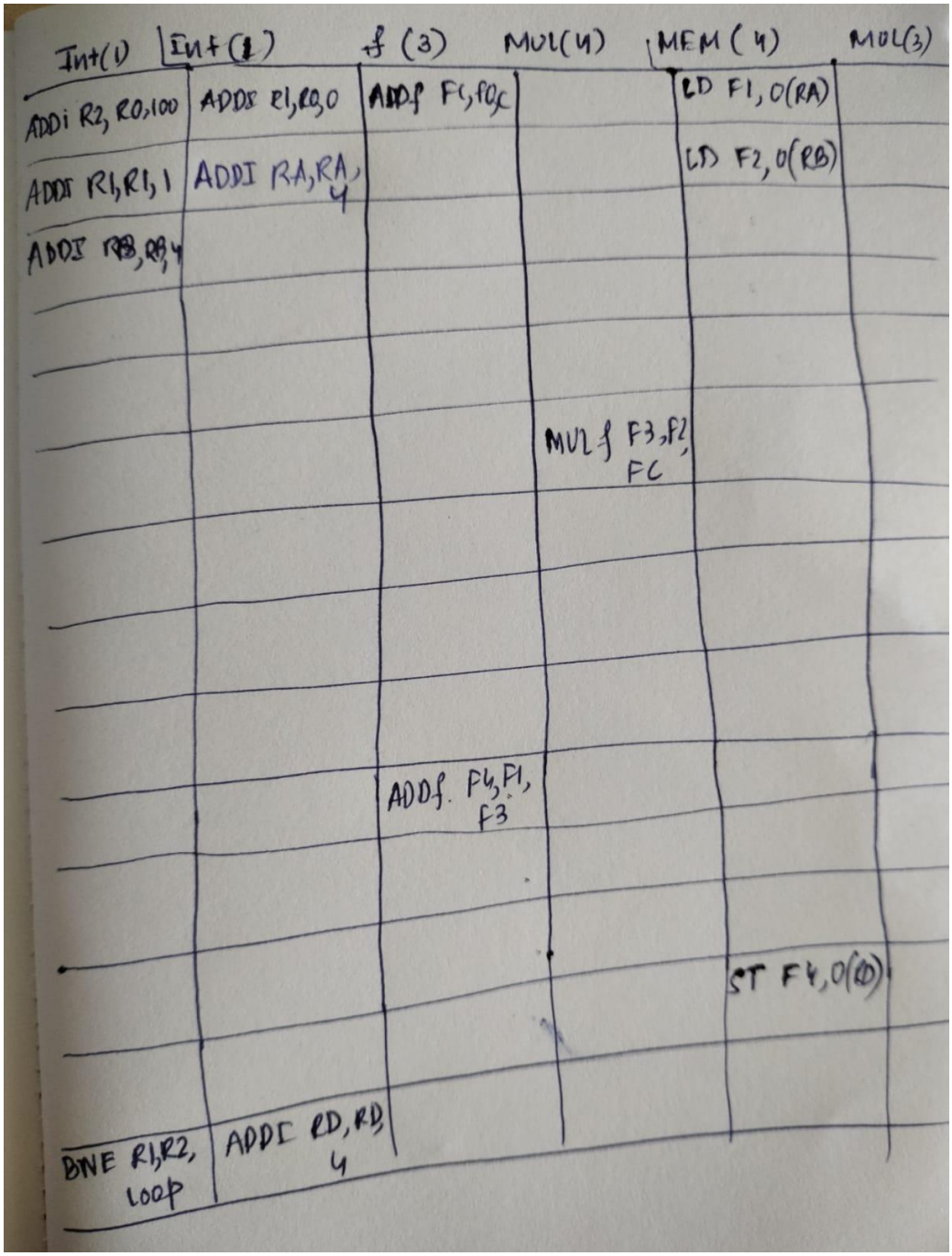
Total hits = 1 + 384 = 385
Total miss = 1023 + 128 = 1151

Q3.



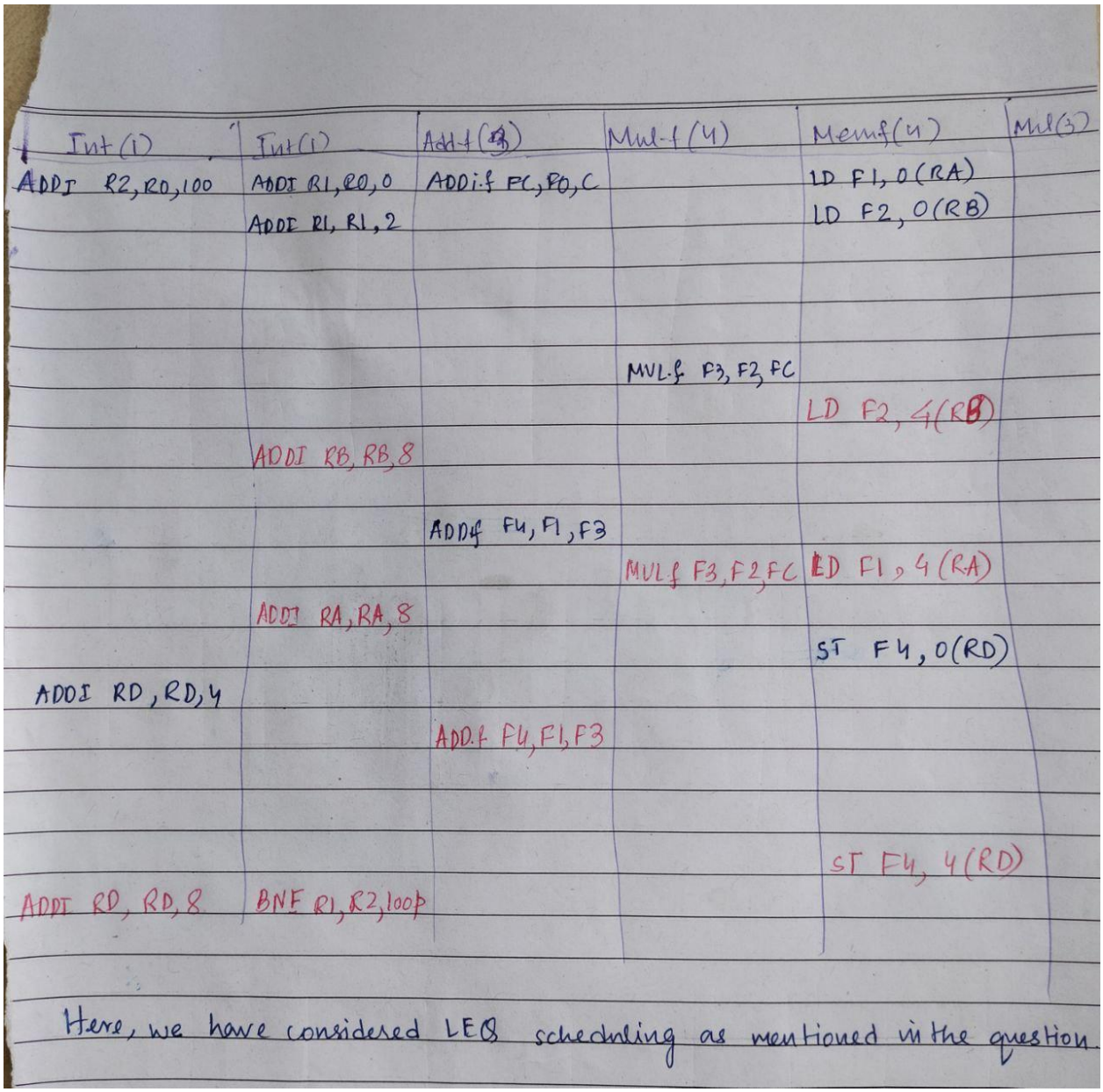
Branch	current state	Prediction	Actual	Correct (C) / Incorrect (I)	Next state	
BEQZ	00	N	N	C	00	for 5 loop iterations
BNEZ	00	N	T	I	01	BEQZ accuracy = 5/5 = 100%
BEQZ	01	N	N	C	00	
BNEZ	00	N	T	I	01	BNEQ accuracy = 1/5 = 20%
BEQZ	01	N	N	C	00	
BNEZ	00	N	T	I	01	
BEQZ	01	N	N	C	00	
BNEZ	00	N	T	I	01	
BEQZ	01	N	N	C	00	
BNEZ	00	N	T	C	00	

Q4. a)



0.6 for MULf; ADDf F4,F1,F3; ST; ADDI RD,RD,4; BNE R1, R2,loop
0.25 marks for rest instruction

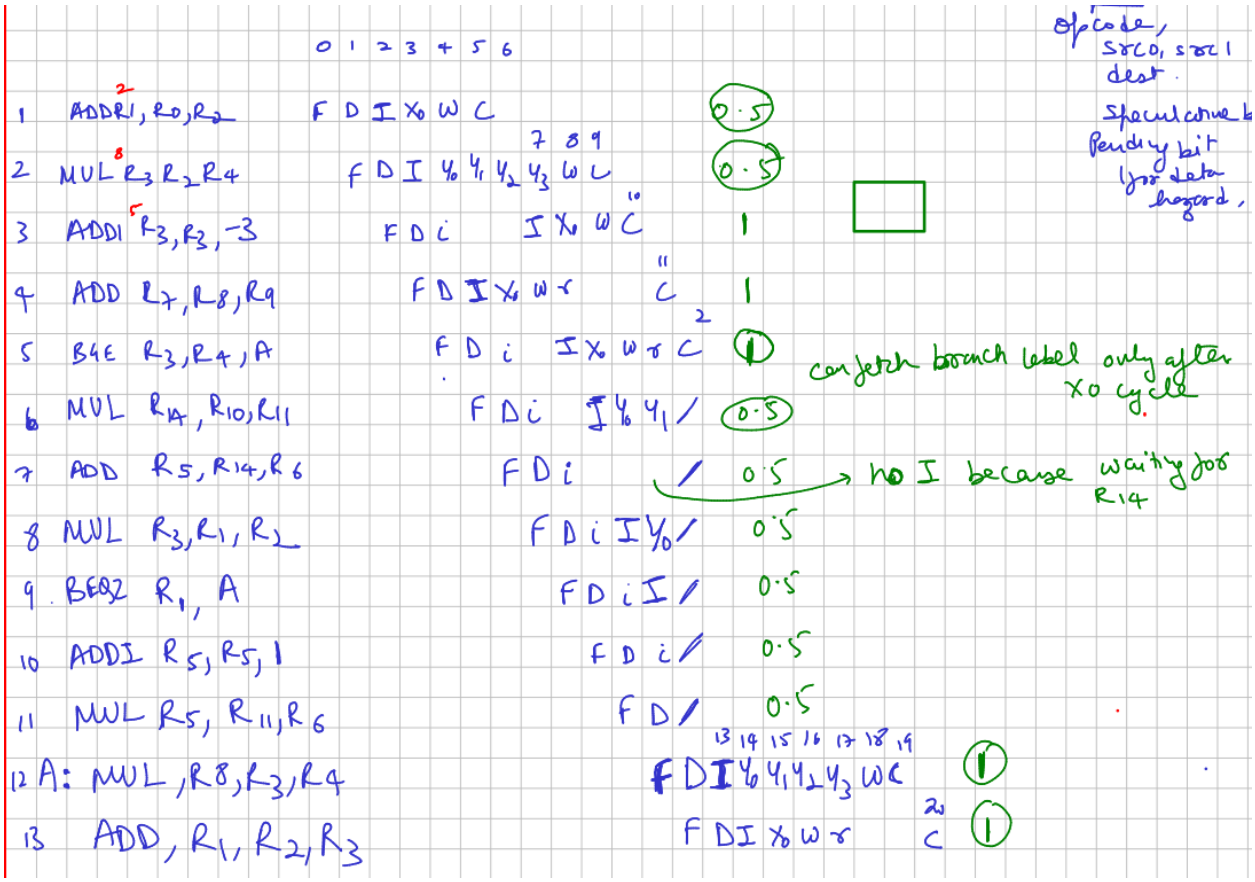
b)



Incremented by 8=0.25

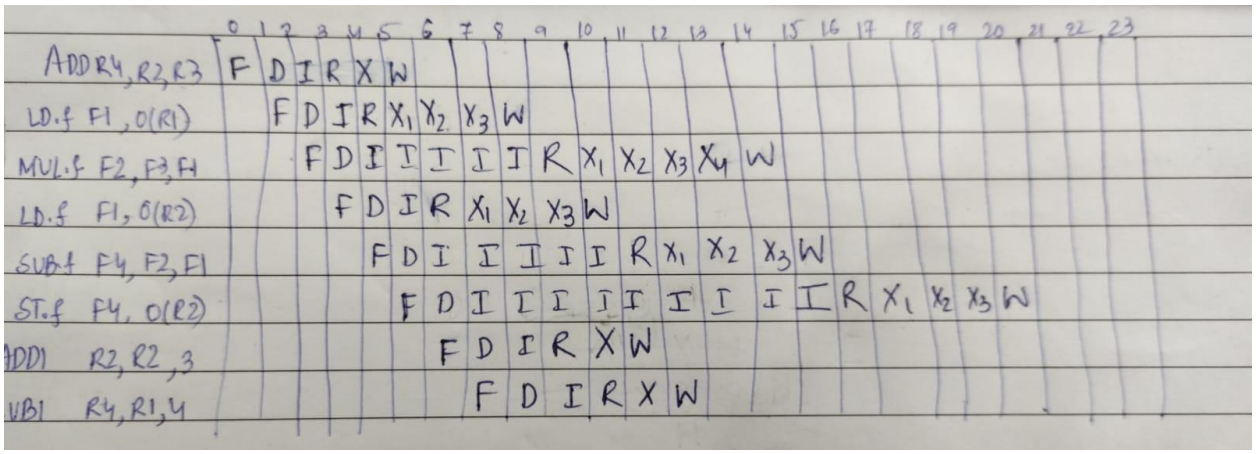
0.25 for each

Q5.

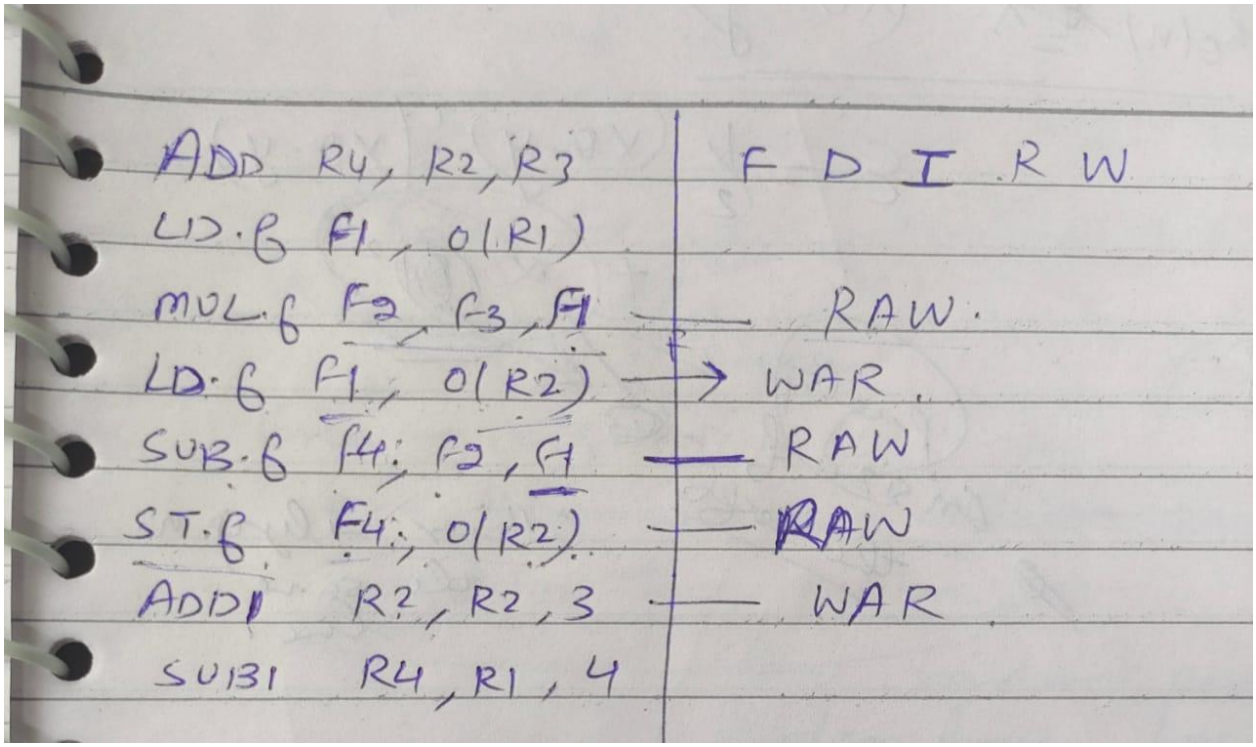

0 1 2 3 4 5 6
7 8 9
13 14 15 16 17 18 19
1 ADDR1, R0, R2
2 MUL R3 R2 R4
3 ADDI R3, R3, -3
4 ADD R7, R8, R9
5 BGE R3, R4, A
6 MUL R10, R11
7 ADD R5, R14, R6
8 MUL R3, R1, R2
9. BEQZ R1, A
10 ADDI R5, R5, 1
11 MUL R5, R11, R6
12 A: MUL, R8, R3, R4
13 ADD, R1, R2, R3
opcode, src0, src1 dest.
Speculative b
Pending bit for data hazard,
0.5
0.5
1
1
1
conjetch branch label only after X0 cycle
0.5
0.5
no I because waiting for R14
0.5
0.5
0.5
1
1

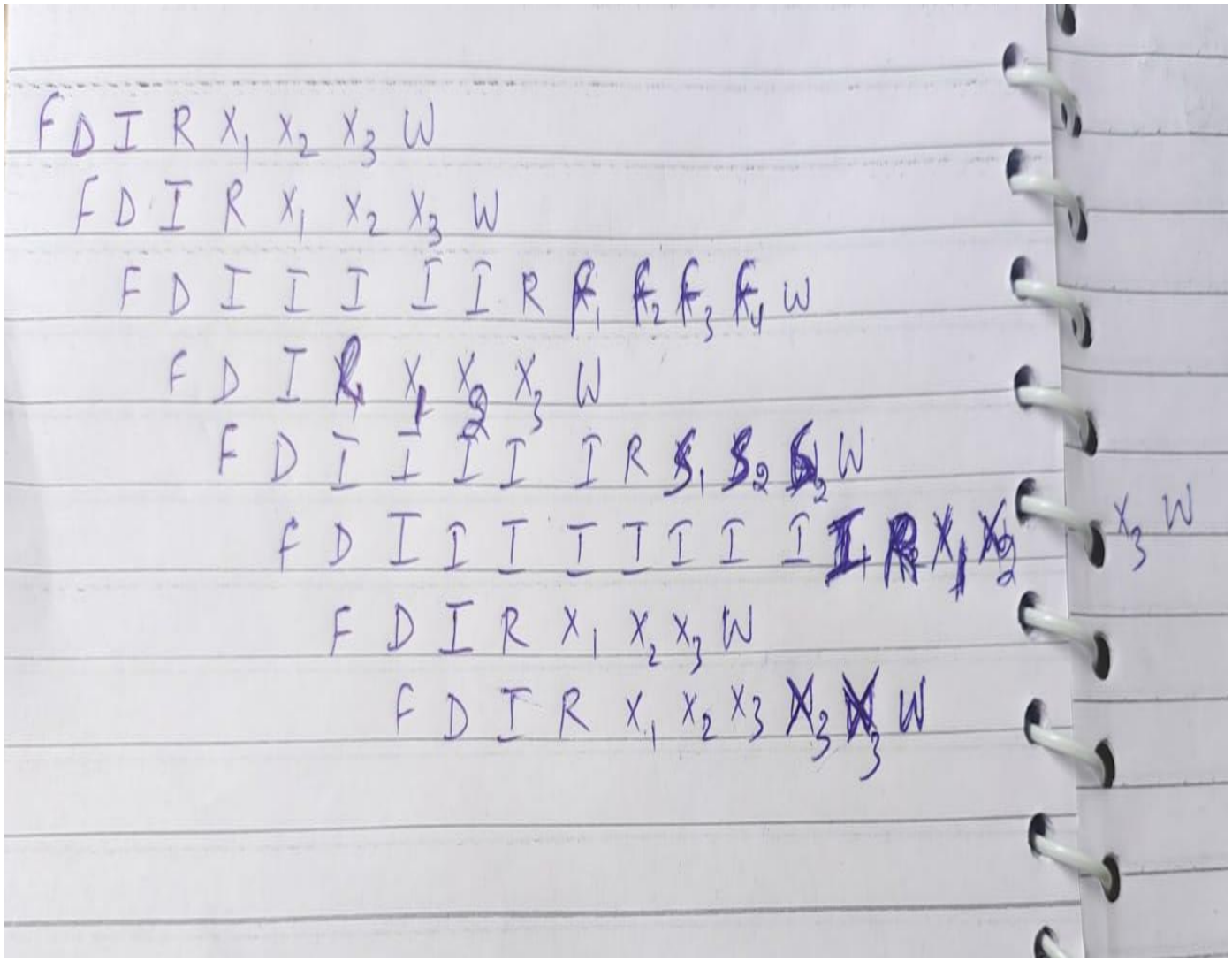
Q6.

By register renaming, we were able to remove WAW hazard between instr. 2 and instr.4, and execute instr. 4 early. Similarly, we were able to remove WAR hazard between instr.6 and instr. 7.



Or




F D I R X1 X2 X3 W
F D I R X1 X2 X3 W
F D I I I I I R F1 F2 F3 F4 W
F D I R X1 X2 X3 W
F D I I I I I R S1 S2 S3 W
F D I I I I I I I I R X1 X2 X3 W
F D I R X1 X2 X3 W
F D I R X1 X2 X3 X3 X3 W