1. Consider execu¸c~ to the following section of c' Odigo a processor com- ISA pat'ıvel with MicroBlaze (ie, corresponding to the processor laborat' the river). O processor has 5 story *pipeline (* IF, ID, EX, MEM and WB) and addresses a mem' oria with 8 bit words and organiza¸c~ to *big endian.*

C'odigo Assembly	Results of control instru¸c~oes (T - *taken;* NT - *Not Taken)*
CALC: ...	
BGE R4 CALC	T, T, T, T, NT T, T, T, NT T, T, T, NT NT
. . .	
BEQ R1, R2, NEXT	NT T NT NT
. . .	
NEXT: ...	
BLT R2 CALC	T T T NT
SW 8 (R5) R6	

1.0 val. (The) Whereas R5 = R6 = 00010128h and 0AB01234h state which position c~oes of mem' oria modi fi ed by the instruction SW 8 (R5) R6 and what its new value.

5.0 val. (B) For each of the predi¸c~ mechanisms the jump indicated below, write down with a c'ırculo correctly predicted the jumps and the success rate in Predi dog jump.

(Note: Consider all simpli fi c Woes deemed appropriate by noting them in the statement.)

i. predictor est' atomic type *not Taken*

BGE R4 CALC	T, T, T, T, NT T, T, T, NT T, T, T, NT NT			Success rate
BEQ R1, R2, NEXT	NT	T	NT NT	
BLT R2 CALC	T	T	T NT	

ii. predictor din amico with *Branch Predit Bu ff ether (* BPB) of 1 bit.

BGE R4 CALC	T, T, T, T, NT T, T, T, NT T, T, T, NT NT			Success rate
BEQ R1, R2, NEXT	NT	T	NT NT	
BLT R2 CALC	T	T	T NT	

iii. predictor din amico with *Branch Predit Bu ff ether (* BPB) of 2 bits.

BGE R4 CALC	T, T, T, T, NT T, T, T, NT T, T, T, NT NT			Success rate
BEQ R1, R2, NEXT	NT	T	NT NT	
BLT R2 CALC	T	T	T NT	

2. Consider execu¸c~ to the following section of c' Odigo a processor com- ISA pat'ıvel with MIPS.

120h:	loop: LD		F0,0 (R1)	; F0 ← M [R1 + 0]
124H:		LD	F2,0 (R2)	; F2 ← M [R 2 + 0]
128h:		MUL.D	F2, F0, F2	; F2 ← F0 × F2
12CH:		DADD.D F2, F2, F4		; F4 ← F2 + F4
130h:		SD	0 (R4), F2	; M [R 4 + 0] ← F4
134h:		DADD	R1, R1, # - 8	; R1 ← R 1 - 8
138H:		DADD	R2 R2 # - 8	; R2 ← R2 - 8
13CH:		DADD	R4 R4 # - 8	; R4 ← R4 - 8
140h:		BNE	R1, R0, loop	; PRAÇA ← loop if R1 = R0

Solve the following al'ıneas considering all simpli fi ca¸c~oes deemed appropriate (where the fi zer write them down in the statement). (The)

3.0 val.

Assuming a processor with 5 est' Agios of *pipeline (* IF, ID, EX, MEM, WB) without *forwarding* data and resolu¸c~ the heels on the floor EX state in Table execution of the steps 1 iterates c~ao the section of c' Odigo.

Instru¸c~ to		Cycle rel' ogen IF	ID	EX MEM	WB Comments		
loop: LD	F0,0 (R1)						
LD	F2,0 (R2)						
MUL.D	F2, F0, F2						
DADD.D F2, F2, F4							
SD	0 (R4), F2						
DADD	R1, R1, # - 8						
DADD	R2 R2 # - 8						
DADD	R4 R4 # - BNE 8						
	R1, R0, loop						

3.0 val.

(B) Considering the existence of *forwarding* data in the table indicate the steps of execu¸c~ 1 to the iteration section of C' Odigo.

Instru¸c~ to		Cycle rel' ogen IF	ID	EX MEM	WB Comments		
loop: LD	F0,0 (R1)						
LD	F2,0 (R2)						
MUL.D	F2, F0, F2						
DADD.D F2, F2, F4							
SD	0 (R4), F2						
DADD	R1, R1, # - 8						
DADD	R2 R2 # - 8						
DADD	R4 R4 # - BNE 8						
	R1, R0, loop						

3.0 val.

(W) Consider a super-scalar processor with:

- din schedule amico using Tomasulo algorithm;
- execu¸c~ the speculative (jump predictor with a 100% success rate);
- *issue* Simultaneous aneo two instru¸c~ oes;
- n'humerus suf fi ciently large esta¸c~ booking oes and entries in the ROB;
- 1 CBD and *commit* Simultaneous aneo 2 instru¸c~ oes;
- functional units with the following latencies:

1 × INT ALU 1 cycle

1 × LOAD / STORE 1 cycle for the address c'alculo CO + 1 cycle to access the `mem'oria

1 × FP ADD 3 cycles

1 × FP MULT 5 cycles

List the steps execu¸c~ to the section of c' Odigo for 2 iterates c~oes.

(fa ca all simpli fi ca¸c~oes deemed appropriate, indicating them with the answer.)

Instru¸c~ to	Cycle rel' ogen					Comments
	IF	issue	EX	CDB	Commit	
loop: LD F0,0 (R1)						
LD F2,0 (R2)						
MUL.D F2, F0, F2						
DADD.D F2, F2, F4						
SD 0 (R4), F2						
DADD R1, R1, # - 8						
DADD R2 R2 # - 8						
DADD R4 R4 # - BNE 8						
R1, R0, loop						
loop: LD F0,0 (R1)						
LD F2,0 (R2)						
MUL.D F2, F0, F2						
DADD.D F2, F2, F4						
SD 0 (R4), F2						
DADD R1, R1, # - 8						
DADD R2 R2 # - 8						
DADD R4 R4 # - BNE 8						
R1, R0, loop						

3. Consider the following section of c' Odigo in *W:*

```
for (i = 1; i <100; i ++) {
    There]     = B [i] * C [i] - D [i];      / * * S1 /
    Bi]        = D [i] + A [i];              / * S2 * /
    A [i-1] = E [i] - F [i];                 / * * S3 /
    D [i + 1] = A [i-1] + G [i];             / * S4 * /
}
```

3.0 val.

(The) Assuming *arrays* s the distinct en~ the overlapping presents a graph with all existing inter-dependencies.

2.0 val.

(B) Say, justi fi cating if oc' Odigo shown'and paraleliz'avel n'ıvel the cycle, ie 'and if poss'ıvel run each iteration c~ao this cycle independently and para lelo. If the rmativo fi, rewrite it so that its execution can be c~ao performed in parallel.