

1. Si stimi il numero di colpi di clock al di sotto del quale non è possibile scendere nell'esecuzione del codice assegnato, qualunque sia il numero di RS, CRB e stadi di Fetch e Decode disponibili, nell'ipotesi che esista una sola unità di tipo **A**, una sola di tipo **D** e una sola di tipo **M** (tre unità funzionali in tutto) e si motivi la risposta (*In questa stima non si tenga conto delle dipendenze nel codice*) (**1 punto**)

ID	Blocco di Codice assegnato		
1	fdiv.s	f2, f9, f3	
2	fmul.s	f1, f9, f3	
3	fadd.s	f4, f9, f3	
4	fdiv.s	f5, f1, f2	
5	fadd.s	f5, f3, f1	
6	fdiv.s	f4, f5, f3	

Si compilino i campi in arancione con le risposte

Nome	
Cognome	
Matricola	

Descrizione unità di esecuzione				Numero di unità funzionali nel datapath	Numero di reservation stations per unità funionale.
A:	FADD	2	T	1	2
M:	FMUL	3	T	1	2
D:	FDIV	4	T	1	2

Da completare con la soluzione dell'esercizio. Si stimi il numero di colpi di clock al di sotto del quale non è possibile scendere nell'esecuzione del codice assegnato. Si motivi la risposta nel campo "descrizione" sottostante.

Unità di Esecuzione / Tipo di istruzione	Numero di istruzioni per tipo	Numero di cicli spesi nell'unità di esecuzione	Numero di Cicli Extra	Numero di cicli totali.
A/FADD	2	2	3	7
M/FMUL	1	3	3	6
D/FDIV	3	4	3	15

Si compilili il campo seguente con la descrizione (risposta e motivazione)

2. Si disegni il grafo delle dipendenze e si deduca il numero minimo di periodi di clock necessario a eseguire il codice assegnato tenendo conto solo delle dipendenze trovate (1 punto).

ID	Blocco di Codice assegnato		Alee: (WAW, RAW, WAR) #si inseriscano le alee presenti per istruzione.	Si compilino i campi in arancione con le risposte	
1	fdiv.s	f2, f9, f3		Nome	
2	fmul.s	f1, f9, f3		Cognome	
3	fadd.s	f4, f9, f3		Matricola	
4	fdiv.s	f5, f1, f2	RAW F2,F1		
5	fadd.s	f5, f3, f1	RAW F2		
6	fdiv.s	f4, f5, f3	RAW F4		

Descrizione unità di esecuzione				Numero di unità funzionali nel datapath	Numero di reservation stations per unità funzionale.
A:	FADD	2	T	1	2
M:	FMUL	3	T	1	2
D:	FDIV	4	T	1	2

Da completare con la soluzione dell'esercizio. Si riporti il grafo delle dipendenze e si deduca il numero minimo di periodi di clock necessario a eseguire il codice assegnato tenendo conto solo delle dipendenze trovate.

Si codifichi il grafo delle dipendenze come una lista con il seguente formato:

si descrivano i nodi del grafo tra parentesi tonde. Separando il numero dell'istruzione ed il numero di cicli spesi in execute dell'istruzione con una virgola all'interno di parentesi tonde. Es.: (id_istruzione, numero_cicli)

Dipendenze	Si completi con il numero di cicli di clock spesi nei differenti stadi della pipeline per ciascun thread di istruzioni. Ciascun thread identifica un flusso indipendente di codice.				
(A,B): A è l'id dell'istruzione e B è il numero di cicli dell'EX	IF	ID	EX	WB	Tot
(1,4)->(4,4)->	1	1	8	2	12
(2,3)->(4,4)->	1	1	7	2	11
(2,3)->(5,2)->(6,4)->	1	1	9	3	14

Si compili il campo seguente con la descrizione (risposta e motivazione)

--

3. Si mostri la dinamica dell'esecuzione nel caso della CPU considerata nel punto 1 con 1 CRB, uno stadio di IF e uno di ID, e 2 RS per ognuna delle tre unità funzionali **(3 punti)**

(si ipotizzi che, in caso di conflitti sul CRB, la fase di WB dell'unità D abbia la priorità sulle altre).

Descrizione unità di esecuzione				Numero di unità funzionali nel datapath	Numero di reservation stations per unità funzionale.
A:	FADD	2	T	1	2
M:	FMUL	3	T	1	2
D:	FDIV	4	T	1	2
CDB				1	

Si compilino i campi in arancione con le risposte

Nome	
Cognome	
Matricola	

Si mostri la dinamica dell'esecuzione nel caso della CPU considerata per il blocco di codice riportato. (Si ipotizzi che, in caso di conflitti sul CRB, la fase di WB dell'unità M abbia la priorità sulle altre).																										
#Si riempiano le caselle con i seguenti simboli : {IF, ID, A0_W, A0_R, A0_E, A1_W, A1_R, A1_E, M0_W, M0_R, M0_E, M1_W, M1_R, M1_E, D0_W, D0_R, D0_E, D1_W, D1_R, D1_E, WB}																										
ID	Instr	Rd	Rs1	Rs2	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22
Blocco di codice					H	L	H	L	H	L	H	L	H	L	H	L	H	L	H	L	H	L	H	L	H	L
1	fdiv.s	f2	f9	f3		IF		ID	D0_E	D0_E	D0_E	D0_E	WB	WB												
2	fmul.s	f1	f9	f3			IF		ID	M0_E	M0_E	M0_E	WB													
3	fadd.s	f4	f9	f3				IF	ID	A0_E	A0_E	WB	WB	WB												
4	fdiv.s	f5	f1	f2					IF	ID	D1_W	D1_W	D1_W	D1_E	D1_E	D1_E	D1_E	WB								
5	fadd.s	f5	f3	f1						IF	ID	A1_W	A1_E	A1_E	WB											
6	fdiv.s	f4	f5	f3							IF	ID	ID	ID	D0_W	D0_R	D0_R	D0_E	D0_E	D0_E	D0_E	WB				

- aumento di RS da due a tre in una o più unità funzionali
- raddoppio o triplicazione del CDB per poter eseguire due o tre fasi di WB per clock
- aggiunta di una o più unità funzionali con due RS

(In caso di raddoppio di una unità funzionale, si ipotizzi di fare lo scheduling a rotazione sulle due unità funzionali uguali)

Si compilino i campi in arancione con le risposte

Nome	
Cognome	
Matricola	

numero CRB complessivo.

ID	Instr	Rd	Rs1	Rs2	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22
	Blocco di codice				H	L	H	L	H	L	H	L	H	L	H	L	H	L	H	L	H	L	H	L	H	L
1	fdiv.s	f2	f9	f3	IF	ID	DO_E	DO_E	DO_E	DO_E	WB															
2	fmul.s	f1	f9	f3		IF	ID	MO_E	MO_E	MO_E	WB															
3	fadd.s	f4	f9	f3			IF	ID	AO_E	AO_E	WB	WB														
4	fdiv.s	f5	f1	f2				IF	ID	X0_W	X0_W	X0_E	X0_E	X0_E	X0_E	WB										
5	fadd.s	f5	f3	f1					IF	ID	A1_W	A1_E	A1_E	WB												
6	fdiv.s	f4	f5	f3						IF	ID	DO_W	DO_W	DO_W	DO_E	DO_E	DO_E	DO_E	WB							

[illegible]