

Esercizio: pipeline

Sia data la seguente sequenza di istruzioni MIPS:

```

Loop:LD R1, 0(R2)      ; load di parola doppia;  $R1 \leftarrow \text{mem}[0 + [R2]]$ 
      DADDI R1, R1, #1 ;  $R1 \leftarrow R1 + 1$ 
      SD (0)R2, R1     ; store di parola doppia;  $\text{mem}[0 + [R2]] \leftarrow R1$ 
      DADDI R2, R2, #4 ;  $R2 \leftarrow R2 + 4$ 
      DSUB R4, R3, R2  ;  $R4 \leftarrow R3 - R2$ 
      BNEZ R4, Loop   ; salta a Loop se  $R4 \neq 0$ 

```

Si assuma che inizialmente $R3 = R2 + 396$. Si utilizzi la pipeline RISC a 5 stadi (IF, ID, EXE, MEM, WB) vista a lezione e si assuma che un qualunque accesso in memoria sia eseguito in 1 ciclo di clock.

- (a) assumendo che non sia possibile effettuare data-forwarding di alcun tipo, mentre i registri possono essere letti e quindi scritti in uno stesso ciclo di clock, si indichino le dipendenze relative alla sequenza di sopra e si disegni il diagramma di temporizzazione delle fasi di ogni istruzione relativamente ad un solo ciclo di esecuzione (una passata). Con l'aiuto di tale diagramma, ed assumendo che il salto sia trattato tramite svuotamento della pipeline, si calcoli il numero totale di cicli impiegati dalle esecuzione del codice.

Esercizio: pipeline

- (b) assumendo che sia possibile anche effettuare data-forwarding, si disegni il diagramma di temporizzazione delle fasi di ogni istruzione relativamente ad un solo ciclo di esecuzione (una passata). Con l'aiuto di tale diagramma, ed assumendo che il salto sia predetto come non preso (not taken), si calcoli il numero totale di cicli impiegati dalle esecuzione del codice.

Esercizio: pipeline

Soluzione parziale (a) (diagramma – per continuare: ricordarsi che il salto è deciso in fase MEM):
Per disegnare il diagramma, indichiamo le istruzioni con i seguenti codici:

Loop:**LD** R1, 0(R2) ; I1
DADDI R1, R1, #1 ; I2
SD (0)R2, R1 ; I3
DADDI R2, R2, #4 ; I4
DSUB R4, R3, R2 ; I5
BNEZ R4, Loop ; I6

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
I1		IF	ID	EXE	MEM	WB											
I2			IF	ID	ID	ID	EXE	MEM	WB								
I3				IF	IF	IF	ID	ID	ID	EXE	MEM	WB					
I4						IF	IF	IF	ID	EXE	MEM	WB					
I5									IF	ID	ID	ID	EXE	MEM	WB		
I6										IF	IF	IF	ID	ID	ID	EXE	MEM