

Esercizio

- n Sapendo che la frequenza del clock è di 1 GHz, valutare le prestazioni della CPU relativamente al frammento di programma a lato ipotizzando che:
- *il ciclo viene eseguito 50 volte*
 - *l'istruzione sottolineata viene eseguita 20 volte.*
- n In particolare:
- determinare i valori di CI, CPI e T_{CPU}
 - motivare i valori assegnati ai CPI_i
 - mostrare in dettaglio le fasi di esecuzione per le istruzioni add, beq e lw.

```
sll $t1,$s2,2
add $t1,$t1,$s0
add $s1,$zero,$zero
loop: beq $t0,$t1,exit
      lw  $t3,0($t0)
      add $s1,$s1,$t3
L:    addi $t0,$t0,4
      j  loop
exit: sw  $s1,4($s4)
```

Soluzione dell'esercizio 2 (1)

- n Ricordiamo che per *prestazioni della CPU* si intende il *tempo di CPU riferito all'utente*, cioè tempo speso dalla CPU all'interno del programma.
- n Il tempo di CPU riferito all'utente può essere formulato come segue:

$$T_{CPU} = N_{cicli} * T_{ck} = CI * \frac{N_{cicli}}{CI} * T_{ck} = CI * CPI * T_{ck}$$

- n dove *CI* è il *conto istruzioni*, cioè il numero di istruzioni eseguite dal processo, e *CPI* è il *numero medio di cicli di clock per istruzione*, determinato come segue:

$$CPI = \frac{N_{cicli}}{CI} = \frac{\sum CPI_i * N_i}{CI}$$

Soluzione dell'esercizio 2 (2)

	<i>Cicli Per Istruzione</i>	<i>n° di esecuzioni dell'istruzione</i>	
	CPI_i	N_i	$CPI_i * N_i$
sll \$t1,\$s2,2	4 cicli	1	4 * 1 = 4
add \$t1,\$t1,\$s0	4 cicli	1	4 * 1 = 4
add \$s1,\$zero,\$zero	4 cicli	1	4 * 1 = 4
loop: beq \$t0,\$t1,exit	3 cicli	51	3 * 51 = 153
lw \$t3,0(\$t0)	5 cicli	50	5 * 50 = 250
beq \$t3,\$zero,L	3 cicli	50	3 * 50 = 150
add \$s1,\$s1,\$t3	4 cicli	20	4 * 20 = 80
L: addi \$t0,\$t0,4	4 cicli	50	4 * 50 = 200
j loop	3 cicli	50	3 * 50 = 150
exit: sw \$s1,4(\$s4)	4 cicli	1	4 * 1 = 4

"conto istruzioni"

$CI = \sum N_i = 275$	$N_{cicli} = \sum CPI_i * N_i = 999$
-----------------------	--------------------------------------

$T_{CPU} = N_{cicli} * T_{ck} = 999 * 10^{-9} \approx 1 \text{ microsec}$

$$CPI = \frac{N_{cicli}}{CI} = \frac{999}{275} = 3,63$$

Soluzione dell'esercizio 2 (3)

	CPI_i	<i>motivazioni dei valori dei CPI_i</i>
sll	4 cicli	prelievo, accesso al banco/decodifica, operazione ALU, scrittura nel banco
add	4 cicli	prelievo, accesso al banco/decodifica, operazione ALU, scrittura nel banco
beq	3 cicli	prelievo, accesso al banco/decodifica, test e aggiornamento PC
lw	5 cicli	prelievo, accesso al banco/decodifica, calcolo ind., lettura mem., scrittura reg
addi	4 cicli	prelievo, accesso al banco/decodifica, operazione ALU, scrittura nel banco
j	3 cicli	prelievo, accesso al banco/decodifica, aggiornamento PC
sw	4 cicli	prelievo, accesso al banco/decodifica, calcolo ind., lettura mem.

ogni fase si esegue in un ciclo $\rightarrow n^\circ$ cicli = n° fasi di esecuzione

Soluzione dell'esercizio 2 (4)

add	prelievo, accesso al banco/decodifica, operazione ALU, scrittura nel banco
beq	prelievo, accesso al banco/decodifica, test e aggiornamento PC
lw	prelievo, accesso al banco/decodifica, calcolo ind., lettura mem., scrittura reg

n Le prime due fasi sono uguali per tutte le istruzioni:

IR	=	Mem[PC]
PC	=	PC + 4

A	=	Reg[IR[25-21]]
B	=	Reg[IR[20-16]]
AluOut	=	PC + sign-ext(IR[15-0]) << 2

Soluzione dell'esercizio 2 (5)

add	prelievo, accesso al banco/decodifica, operazione ALU, scrittura nel banco
beq	prelievo, accesso al banco/decodifica, test e aggiornamento PC
lw	prelievo, accesso al banco/decodifica, calcolo ind., lettura mem., scrittura reg

n **Fasi 3 e 4 della add:**

AluOut = A op B

Reg[IR[15-11]] = AluOut

n **Fase 3 della beq:**

if (A == B) PC = AluOut

n **Fasi 3, 4 e 5 della lw:**

AluOut = A + sign-ext(IR[15-0])

MDR = Mem[AluOut]

Reg[IR[20-16]] = MDR