

Nome e Cognome:

Matricola:

Corso di Architettura degli Elaboratori 1

Anno Accademico 2005/2006

Esempio di compito

Istruzioni

- Scrivere *Nome*, *Cognome* e *Matricola* su **ogni** foglio (solo pagine **dispari**).
- Scrivere la risposta nello spazio bianco al di sotto della domanda; Non è possibile allegare fogli aggiuntivi, quindi cercate di essere chiari e non prolissi.
- In caso di errori indicate chiaramente quale parte della risposta deve essere considerata; annullate le parti non pertinenti.
- Assicuratevi che non manchi alcun foglio al momento della consegna.

Esercizi a risposta multipla

es1

Sia data la seguente sequenza di istruzioni assembler, dove i dati immediati sono espressi in esadecimale

```
LB   R3, 80(R0)
ADD  R2, R0, R0
LB   R1, 800(R2)
ADDI R1, R1, 3
ADDI R2, R2, 4
SB   R1, 108(R2)
SUB  R4, R3, R2
```

Si consideri la pipeline MIPS a 5 stadi vista a lezione, senza possibilità di data-forwarding, ma con possibilità di scrittura e successiva lettura dei registri in uno stesso ciclo di clock. L'esecuzione completa del codice avviene in un numero di cicli di clock pari a

- a) 13
- b) 15
- c) 9
- d) 17
- e) in nessuno dei circuiti elencati precedentemente;

es2

Una cache set-associativa consiste di 128 linee, divise in insiemi di 8 linee. La memoria centrale contiene 4K blocchi di 128 parole ciascuno. Il formato degli indirizzi della memoria centrale è:

- a) Etichetta = 8 bit; Set = 6 bit; Parola = 6;
- b) Etichetta = 9 bit; Set = 5 bit; Parola = 7;
- c) Etichetta = 11 bit; Set = 3 bit; Parola = 6;
- d) Etichetta = 8 bit; Set = 4 bit; Parola = 7;
- e) nessuna delle risposte precedenti è corretta;

es3

Si consideri una cache di 4KB con associazione a gruppi a 4 vie (4-way set associative) in congiunzione con una memoria centrale di 256KB, indirizzabile al byte. Le locazioni di memoria con indirizzi (in esadecimale) 30E5C e 1A62C hanno la possibilità di essere caricate all'interno dello stesso set di linee se la dimensione del blocco è di:

- a) 64KB;
- b) 128KB;
- c) 32KB;
- d) nessuna delle risposte precedenti è corretta;

es4

In un disco CD-ROM, i dati possono essere memorizzati

- a) esclusivamente in campi di 2048B;
- b) esclusivamente in campi di 512B;
- c) in campi di 2048B o di 2336B;
- d) nessuna delle risposte precedenti è corretta;

Esercizi a risposta libera

es5

Dire quali sono le motivazioni sottostanti al RAID e spiegarlo in dettaglio

es6

Se si parla di “finestre di registri”, a quale tipo di architettura di CPU sia fa riferimento ?
Spiegare in dettaglio a cosa servono le “finestre di registri” e come funzionano.

es7

Nel contesto di una pipeline spiegare in dettaglio le tecniche viste a lezione per gestire le dipendenze dai controlli

es8

Spiegare in dettaglio le differenze fra un modulo di memoria DRAM ed un modulo di memoria SDRAM

Esercizi pratici

es9

Sia data la seguente sequenza di istruzioni assembler, dove i dati immediati sono espressi in esadecimale ed il registro R0 contiene il valore 0:

```

LB   R3, 0(R0)      ! load byte da mem[0+[R0]]
ADD  R2, R0, R0     ! R2 = R0 + R0
LB   R1, 8(R2)      ! load byte da mem[8+[R2]]
ADDI R1, R1, 3      ! R1 = R1 + 3
ADDI R2, R2, 4      ! R2 = R2 + 4
SB   R1, 8(R2)      ! store byte in mem[8+[R2]]
SUB  R4, R3, R2     ! R4 = R3 - R2
BGTZ R4, -6         ! PC = PC - 6 se [R4] > 0
                        ! cioe' salta alla istruzione LB   R1, 8(R2)

```

Si assuma la presenza di due cache, una dati ed una istruzioni. La cache dati, in particolare, è di ampiezza 8B, con dimensione di blocco 2B, inizialmente vuota, ed associazione 2-way (con politica di rimpiazzo LRU e politica di scrittura write-back). Si assuma che la memoria abbia il contenuto esadecimale mostrato di seguito (si esprimano gli indirizzi su 8 bit):

Indirizzo	byte	byte	byte	byte
00	0C	00	07	02
04	00	00	00	00
08	AE	13	A1	23
0C	A1	42	90	75
10	B9	16	00	00
14	0A	07	03	71

Si mostri come sia il contenuto della cache dati che il contenuto della memoria cambia a causa della esecuzione del codice assembler.

Nome e Cognome:

Matricola:

Pagina 9

es10

Considerando la pipeline MIPS vista a lezione, si consideri il seguente frammento di codice:

```
loop:  LW      R1, 0(R2)    ! R1 = mem[0+[R2]]
        ADDI   R1,R1, 1    ! R1 = [R1] + 1
        SW     R1, 0(R2)    ! mem[0+[R2]] = [R1]
        ADDI   R2, R2, 4    ! R2 = [R2] + 4
        SUB    R4, R3, R2   ! R4 = [R3] - [R2]
        BENZ   R4, loop     ! if([R4] != 0) PC = indirizzo(loop)
```

Mostrare come evolve la pipeline durante l'esecuzione del codice per le prime 6 istruzioni eseguite, assumendo:

- possibilità di forwarding, così come visto a lezione per la pipeline MIPS;
- che il salto condizionale (BENZ) sia trattato con stallo della pipeline fino al calcolo dell'indirizzo target.

