



Cognome e nome dello studente:

Matricola:

Anno di corso e turno (diurno o serale):

A.A. 2001-2002 – Prova d'esame del 14 ottobre 2002

ESERCIZIO 1 (6+2 punti)

Si traducano in linguaggio macchina in formato binario ed in linguaggio ad alto livello (usando le strutture di controllo appropriate), le seguenti istruzioni assembly:

```
L1:    beq $s3, $zero, L2          24:
        lui $at, 8                28
        ori $s1, $at, 16          32
        .....
        add $s3, $s1, $s2        40
        lw $t0, 4($s3)           48
        add $t0, $t0, $t0        52
        sw $t0, 4($s3)           56
        j L1                      60
        bne $s1, $zero, L1       64:
        j L3                      68
        .....
L2:    j L3                        80:
        .....
L3:    jr $s1                      108:
```

Facendo riferimento a: **lui** 0xf 0 rt imm, **ori** 0xd rs rt imm, **add** 0x0 rs rt rd 0x0 0x20, **lw** 0x23 rs rt offset, **sw** 0x26 rs rt offset, **bne** 0x5 rs rt offset, **beq** 0x4 rs rt offset, **j** 0x2 label, **jr** 0x0 rs 0x000008.

E ricordando che nel processore MIPS i registri: \$zero, \$at, \$s0, \$s1, \$s2, \$s3, \$s4, \$t0, \$t1 corrispondono rispettivamente ai registri \$0, \$1, \$16, \$17, \$18, \$19, \$20, \$8, \$9. Supponete che l'indirizzo di base di un vettore (*vett*) coincida con \$s3.

ESERCIZIO 2 (12 punti)

Si scriva in linguaggio assembly del processore MIPS un programma (main) che calcola la seguente funzione:

$$y = \text{multi}(a, b) = a * b * 2 - 1$$

La funzione multi sarà contenuta in una procedura richiamata dal main.

Utilizzare all'interno della procedura multi almeno una volta il registro \$s0.

Il main sarà costituito dalle funzini di lettura delle variabili e di scrittura del risultato. Inoltre, nel main compare il seguente pezzo di codice:

```
....
add $t0, $s1, $s2
add $t1, $s3, $s4
...
Chiamata della procedura multi
....
add $t2, $t0, $t1
....
```

Salvare in stack solamente le informazioni **strettamente necessarie** al corretto funzionamento. Fare attenzione alle convenzioni sul salvataggio dei registri in stack!!

ESERCIZIO 3 (3 + 3 punti)

Si consideri un calcolatore con una frequenza di clock pari a 1 Ghz. Dobbiamo valutare se sia conveniente un trasferimento DMA o mediante interrupt da un dispositivo I/O con le seguenti caratteristiche:

- Quantità di dati da trasferire: 2Mbyte.
- Quantità trasferita a controllo di programma: 1 parola (4 byte)
- Numero di cicli di clock per gestire l'interrupt: 500.
- Numero di cicli di clock per la gestione del controller DMA: 2000 (inizializzazione + terminazione)
- Numero di parole trasferite in DMA: 20.

Qual è la dimensione del blocco dati (frazionaria) trasferita in DMA che rende DMA ed I/O equivalenti in termini di tempo?

ESERCIZIO 4 (4+2 punti)

Gerarchie di memoria. La MISS_PENALTY è di 6 cicli di clock, mentre tutte le istruzioni impiegano 8.5 cicli di clock se si ignorano i MISS (stalli della memoria). Ipotizzando un MISS_RATE dell'11% e che vi siano in media 2 riferimenti alla memoria per ogni istruzione, determinare a) Qual è l'impatto sulle prestazioni quando viene inserita la cache reale rispetto ad una cache ideale? b) Qual è l'impatto sulle prestazioni tra il caso di cache reale e senza inserimento della cache?

DOMANDA (1 punto)

Quando è più conveniente un'architettura LOAD/STORE e quando è più conveniente un'architettura del tipo tradizionale?