

Nome, cognome, matricola

Calcolatori Elettronici (12AGA) – esame del 31.01.2024

Domande a risposta chiusa (è necessario rispondere correttamente ad almeno 6 domande).
Non è possibile consultare alcun tipo di materiale. Tempo: 15 minuti.

1	Quante righe ha la tavola di verità di un codificatore prioritario 4:2 senza enable?		
2	Si consideri un bus sincrono. Che cosa si intende per ciclo di wait durante un'operazione di lettura?	Un ritardo periodico sul segnale di Ready	A
		Un ritardo aggiuntivo sul ciclo di lettura dati	B
		L'aggiunta di cicli aggiuntivi di attesa per ottenere più tempo	C
		Un ritardo periodico sul segnale di Clock	D
3	Si consideri una cache con le seguenti caratteristiche • 128 linee da 16 byte • Meccanismo set associative a 4 vie con sostituzione LRU. Assumendo che gli indirizzi emessi dal processore siano su 32 bit, qual è la dimensione del campo tag associato a ogni linea?	21 bit	A
		23 bit	B
		25 bit	C
		28 bit	D
4	Quale vantaggio offre una memoria secondaria di tipo SSD basata su MLC Flash rispetto a una memoria secondaria di tipo HDD?	Minore consumo di potenza	A
		Minor costo	B
		Capacità di memoria maggiore	C
		Velocità di scrittura maggiore	D
5	Quale tra i fenomeni elencati a lato può causare uno stallo in un processore con pipeline?	Un miss nella cache dati	A
		Un errore in un'operazione aritmetica	B
		Un'operazione di I/O	C
		L'esecuzione di un'istruzione NOP	D
6	Quale dei seguenti dispositivi può diventare un master di un bus?	DMA controller	A
		Interfaccia di periferico	B
		Memoria	C
		Interrupt Controller	D
7	Si consideri il meccanismo della parità, utilizzato per aumentare l'affidabilità delle DRAM: quando viene calcolato il bit di parità associato ad una parola?	Solo quando la parola viene scritta in memoria	A
		Solo quando la parola viene letta dalla memoria	B
		Quando la parola viene scritta in memoria e quando viene letta dalla memoria	C
		Solo in momenti prefissati, in cui si calcola il bit di parità per tutte le parole presenti in quel momento in memoria	D
8	Si consideri un'unità di controllo microprogrammata basata su una memoria di microcodice composta da 536 parole da 18 bit: qual è il parallelismo del μ PC?	11	A
		10	B
		9	C
		Nessuno dei precedenti	D
9	Considerando il linguaggio MIPS a quale tipo di formato corrisponde l'istruzione <i>jal</i> ?	S-type	A
		R-type	B
		I-type	C
		J-type	D

10	<p>Dato il seguente frammento di codice in linguaggio C si scriva il corrispondente codice in linguaggio MIPS supponendo che \$s0 = i, \$s1 = sum:</p> <pre>int sum = 0; int i; for (i = 0; i != 10; i = i+1) { sum = sum + i; }</pre>	
----	--	--

Risposte corrette

1	2	3	4	5	6	7	8	9	10
16	C	B	A	A	A	C	B	D	

Esempio di risposta alla domanda 10:

```
#$s0 = i, $s1 = sum
    addi $s1, $0, 0
    add $s0, $0, $0
    addi $t0, $0, 10
for:   beq $s0, $t0, done
    add $s1, $s1, $s0
    addi $s0, $s0, 1
    j for
done:
```

Domande a risposta aperta (sino a 5 punti per ogni domanda) – Non è possibile consultare alcun materiale -
Tempo: 40 minuti.

11	<p>Si consideri un sistema a processore che include</p> <ul style="list-style-type: none">• una memoria di 1k byte• una cache direct-mapped composta da 4 linee da 8 byte ciascuna. <p>Si assuma che inizialmente la cache contenga i blocchi 0, 1, 2, 3. Si determini il numero del blocco presente in ciascuna linea della cache al termine della sequenza di accessi in memoria corrispondente agli indirizzi riportati in tabella e si riporti il risultato nella tabella apposita.</p> <p>Sequenza degli accessi alla memoria</p> <table border="1"><tr><td>Accesso 1</td><td>00 00 01 01 01</td></tr><tr><td>Accesso 2</td><td>11 00 01 01 11</td></tr><tr><td>Accesso 3</td><td>10 00 01 10 01</td></tr><tr><td>Accesso 4</td><td>00 00 00 00 00</td></tr><tr><td>Accesso 5</td><td>10 01 11 01 00</td></tr><tr><td>Accesso 6</td><td>10 01 01 01 11</td></tr><tr><td>Accesso 7</td><td>01 00 01 10 01</td></tr><tr><td>Accesso 8</td><td>10 10 11 11 00</td></tr></table> <p>Contenuto iniziale della cache</p> <table border="1"><tr><th>Linea</th><th>Blocco</th></tr><tr><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td></tr><tr><td>2</td><td>2</td></tr><tr><td>3</td><td>3</td></tr></table> <p>Contenuto finale della cache</p> <table border="1"><tr><th>Linea</th><th>Blocco</th></tr><tr><td>0</td><td></td></tr><tr><td>1</td><td></td></tr><tr><td>2</td><td></td></tr><tr><td>3</td><td></td></tr></table>	Accesso 1	00 00 01 01 01	Accesso 2	11 00 01 01 11	Accesso 3	10 00 01 10 01	Accesso 4	00 00 00 00 00	Accesso 5	10 01 11 01 00	Accesso 6	10 01 01 01 11	Accesso 7	01 00 01 10 01	Accesso 8	10 10 11 11 00	Linea	Blocco	0	0	1	1	2	2	3	3	Linea	Blocco	0		1		2		3	
Accesso 1	00 00 01 01 01																																				
Accesso 2	11 00 01 01 11																																				
Accesso 3	10 00 01 10 01																																				
Accesso 4	00 00 00 00 00																																				
Accesso 5	10 01 11 01 00																																				
Accesso 6	10 01 01 01 11																																				
Accesso 7	01 00 01 10 01																																				
Accesso 8	10 10 11 11 00																																				
Linea	Blocco																																				
0	0																																				
1	1																																				
2	2																																				
3	3																																				
Linea	Blocco																																				
0																																					
1																																					
2																																					
3																																					

12	<p data-bbox="121 69 1532 129">Si descrivano le principali caratteristiche dell'architettura di un processore basato su pipeline, mettendo in evidenza i vantaggi ed elencando i possibili fattori che ne riducono l'efficacia.</p>
----	---

13	<p>Si elenchino le operazioni che si susseguono in un sistema a processore che utilizza il meccanismo dell'interrupt vettorizzato dal momento in cui una periferica attiva una richiesta di interrupt al momento in cui la CPU esegue la prima istruzione della corrispondente procedura di servizio dell'interrupt (ISR).</p>
----	--

14	<p data-bbox="121 69 1528 163">Si consideri il meccanismo della memoria virtuale: si descriva la sequenza di operazioni eseguite per trasformare ciascun indirizzo logico generato dal processore nel conseguente indirizzo fisico. Per ciascuna operazione, si specifichi quali sono i moduli hardware o i componenti software coinvolti.</p>
----	--

Nome, Cognome, Matricola:.....

Esercizio di programmazione

sino a 12 punti – è possibile consultare solamente il foglio consegnato con l'istruzione set MIPS - tempo: 60 minuti

Sono date due matrici di numeri interi con segno espressi su halfword. Si scriva in linguaggio Assembly MIPS una procedura **contaUguali** che riceve i seguenti parametri (nell'ordine indicato):

- l'indirizzo della prima matrice
- il numero di colonne della prima matrice
- la riga da considerare nella prima matrice
- l'indirizzo della seconda matrice
- il numero di colonne della seconda matrice
- la riga da considerare nella seconda matrice.

I primi 4 parametri sono passati nei registri, gli ultimi 2 attraverso lo stack.

La procedura confronta gli elementi nella riga indicata per la prima matrice con gli elementi nella riga indicata per la seconda matrice e restituisce il numero di elementi uguali in valore assoluto (senza considerarne il segno).

Esempio

Matrice 1:

<i>riga 0:</i>	-1	2	3	4	-5
<i>riga 1:</i>	6	-7	8	-9	10
<i>riga 2:</i>	11	12	-13	14	15
<i>riga 3:</i>	16	-17	18	-19	20

Matrice 2:

<i>riga 0:</i>	-1	12	7	-24	-13	18
<i>riga 1:</i>	2	-10	-11	22	14	17
<i>riga 2:</i>	-3	8	9	20	-15	-16

Se nella chiamata alla procedura i parametri indicano di considerare la riga 2 nella prima matrice e la riga 1 nella seconda matrice, il numero di elementi uguali in valore assoluto è 2 (gli elementi uguali sono evidenziati in grassetto nell'esempio).

Di seguito un esempio di programma chiamante:

```
.data
matrice1: .half -1, 2, 3, 4, -5
          .half 6, -7, 8, -9, 10
          .half 11, 12, -13, 14, 15
          .half 16, -17, 18, -19, 20
matrice2: .half -1, 12, 7, -24, -13, 18
          .half 2, -10, -11, 22, 14, 17
          .half -3, 8, 9, 20, -15, -16

.text
.globl main
.ent main
main:  subu $sp, $sp, 4
       sw $ra, ($sp)
       la $a0, matrice1
       li $a1, 5
       li $a2, 2
       la $a3, matrice2
       subu $sp, $sp, 8
       li $t0, 6
       sw $t0, ($sp)
       li $t0, 1
       sw $t0, 4($sp)
       jal contaUguali
       addu $sp, $sp, 8
       lw $ra, ($sp)
       addiu $sp, $sp, 4
       jr $ra
.end main
```


Soluzione proposta

```
.ent contaUguali
contaUguali:
    li $v0, 0          # contatore elementi uguali
    mul $t0, $a1, $a2
    sll $t0, $t0, 1     # offset matrice1
    add $t0, $t0, $a0

    lw $t1, ($sp)
    lw $t2, 4($sp)
    mul $t2, $t1, $t2    # offset matrice1
    sll $t2, $t2, 1
    add $t2, $t2, $a3

cicloMatrice1:
    lh $t3, ($t0)
    bgtz $t3, positivo1
    sub $t3, $0, $t3
positivo1:
    move $t4, $t1
    move $t5, $t2
cicloMatrice2:
    lh $t6, ($t5)
    bgtz $t6, positivo2
    sub $t6, $0, $t6
positivo2:
    bne $t3, $t6, incrementaIndice
    addi $v0, $v0, 1
incrementaIndice:
    add $t5, $t5, 2
    sub $t4, $t4, 1
    bgtz $t4, cicloMatrice2

    add $t0, $t0, 2
    sub $a1, $a1, 1
    bgtz $a1, cicloMatrice1

    jr $ra
.end contaUguali
```