

Nome, cognome, matricola

Calcolatori Elettronici (12AGA) – esame del 26.6.2019 - A

Domande a risposta chiusa (è necessario rispondere correttamente ad almeno 6 domande).

Non è possibile consultare alcun tipo di materiale. Tempo: 15 minuti.

!!!! Attenzione: il compito è su 2 facciate !!!!

1	Si consideri un banco di memoria di dimensioni pari a 16M parole, ciascuna da 32 bit, composto di moduli da 2 Mparole da 1 byte ciascuna. Quanti moduli compongono il banco?		
2	Si confrontino un contatore sincrono e uno asincrono. Quale delle seguenti affermazioni è <u>vera</u> ?	Il contatore sincrono ha un ritardo indipendente dal numero di bit	A
		Il contatore sincrono richiede una quantità minore di hardware	B
		Il contatore sincrono è più facile da progettare	C
		Il contatore sincrono può contare solo in avanti e non all'indietro	D
3	Il segnale di interrupt acknowledge è attivato...	quando la CPU termina la routine di servizio dell'interrupt	A
		quando la CPU ha rilevato la richiesta di interruzione	B
		al momento dell'inizio della routine di servizio dell'interrupt per inibire nuove richieste	C
		in nessuno dei casi precedenti	D
4	Si consideri la memoria di microcodice esistente in un'unità di controllo microprogrammata. A che tipo di memoria corrisponde ?	ROM	A
		SRAM	B
		DRAM	C
		Flash	D
5	Lo stack è...	un'area dati utilizzata per memorizzare gli indirizzi di inizio delle routine di servizio dell'interrupt	A
		un'area dati usata per memorizzare le variabili globali	B
		un'area dati usata per memorizzare le variabili locali	C
		un'area dati usata per memorizzare le strutture dati allocate dinamicamente (ad esempio tramite <i>malloc</i>)	D
6	Si consideri un sistema che utilizza il meccanismo del daisy chain per l'arbitraggio del bus tra 4 unità: si disegnano le connessioni tra l'arbitro e le 4 unità.		
7	Si consideri il meccanismo noto come write-back utilizzato nella gestione delle cache. Quale delle seguenti affermazioni è <u>vera</u> ?	Esiste un dirty bit per ciascuna linea	A
		Il dirty bit viene complementato ad ogni operazione di lettura	B
		Nel caso di write-back il tempo medio richiesto per gestire un miss è minore rispetto al write-through	C
		Tutte le operazioni di scrittura vengono eseguite sia sulla cache sia sulla memoria	D

8	Quale valore assume il parametro CPI (Clocks per Instruction) in un processore RISC?	Sempre inferiore a 1	A	
		Sempre uguale a 1	B	
		Sempre superiore a 1, con la possibilità di arrivare a 1 in alcuni casi particolari	C	
		Sempre significativamente superiore a 1	D	

9	Si consideri l'istruzione lw \$4, 16(\$2). A quale dei tipi a fianco appartiene?	R-type	A	
		I-type	B	
		J-type	C	
		Nessuno dei precedenti	D	

10	Si scriva un frammento di codice in Assembler MIPS che, date due variabili con segno su 32 bit VAR1 e VAR2, esegua lo scambio dei loro valori, in modo che al termine dell'operazione VAR1 contenga il valore originariamente in VAR2, e viceversa.	
----	---	--

Risposte corrette

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

Domanda 10 (esempio di soluzione)

lw \$t0, var1
lw \$t1, var2
sw \$t0, var2
sw \$t1, var1

Compito A

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 256 byte • una cache direct-mapped composta da 4 linee da 4 byte ciascuna.
----	--

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 figura.

Sequenza degli accessi alla memoria

Accesso 1	00010101
Accesso 2	10010011
Accesso 3	00011001
Accesso 4	00000100
Accesso 5	00111100
Accesso 6	01000111
Accesso 7	10001001
Accesso 8	01011100

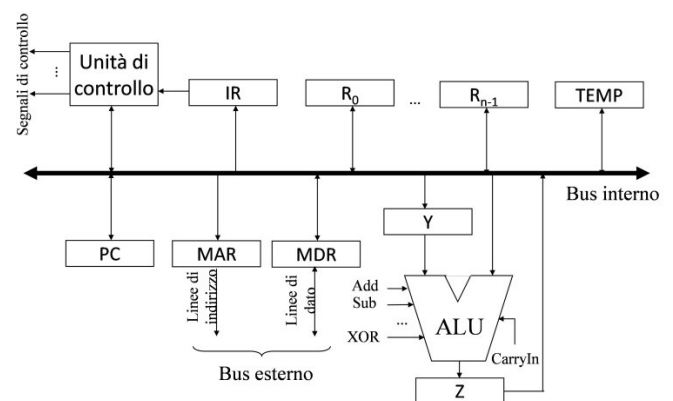
Contenuto iniziale della cache

Linea	Blocco
0	0
1	1
2	2
3	3

Contenuto finale della cache

Linea	Blocco
0	
1	
2	
3	

12	<p>Si scrivano le microistruzioni (inclusive della fase di fetch) eseguite da un processore avente l'architettura in figura durante l'esecuzione dell'istruzione SUB R2, R3, 24</p> <p>Tale istruzione sottrae al valore contenuto in R3 il valore 24, e scrive il risultato in R2.</p>
----	---



13	<p>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>
14	<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>

Esercizio di programmazione

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

La notazione polacca inversa permette di scrivere formule matematiche senza utilizzare parentesi e regole di precedenza degli operatori. Quando si incontra un operatore all'interno dell'espressione matematica, questo si applica ai due operandi precedenti. Ad esempio, l'espressione $18 + 25 * (10 - 7) - 13$ rappresentata con la notazione polacca inversa diventa $18\ 25\ 10\ 7\ -\ *\ +\ 13\ -$

Si vuole realizzare un programma in linguaggio MIPS per calcolare il valore di un'espressione in notazione polacca inversa. L'espressione è memorizzata in un array di word, in cui gli elementi sono così codificati:

- se il primo bit è 0, l'elemento dell'array è un numero positivo nell'intervallo $(0, 2^{31} - 1)$
- se il primo bit è 1, l'elemento dell'array è un operatore (+, -, *, /). Una possibile codifica è la seguente:
somma = -1; sottrazione = -2; moltiplicazione = -3; divisione = -4

Si supponga che la sintassi dell'espressione in notazione polacca inversa sia corretta e che nel calcolo del risultato non vi sia overflow. Si scriva una procedura `calcolaPolaccaInversa` che riceva in input come primo parametro l'indirizzo dell'array contenente l'espressione e come secondo parametro la lunghezza dell'array. La procedura restituisce il valore dell'espressione.

La procedura scandisce ogni elemento dell'array. Se è un operando, ne fa il push nello stack. Se è un operatore, preleva i due elementi in cima allo stack e chiama la procedura `eseguiOperazione` per ottenere il valore dell'operazione; questo valore è poi inserito in cima allo stack. La procedura `eseguiOperazione` riceve in input l'operatore, il primo operando, il secondo operando; restituisce il valore dell'operazione.

Tutte le procedure devono essere conformi allo standard e alle specifiche per quanto riguarda il passaggio dei parametri in input, del valore di ritorno e dei registri da preservare.

Di seguito un esempio di programma chiamante e della procedura `eseguiOperazione`:

```
.data
espressione:      .word 18, 25, 10, 7, -2, -3, -1, 13, -2
tabella:          .word somma, sottrazione, moltiplicazione, divisione

.text
.globl main
.ent main
main: subu $sp, $sp, 4
      sw $ra, ($sp)
      la $a0, espressione
      li $a1, 9
      jal calcolaPolaccaInversa

      lw $ra, ($sp)
      addu $sp, $sp, 4
      jr $ra
.end main

eseguiOperazione:
      subu $t0, $zero, $a0
      subu $t0, $t0, 1
      sll $t0, $t0, 2
      lw $t1, tabella($t0)
      jr $t1
somma: addu $v0, $a1, $a2
      b fine
sottrazione: subu $v0, $a1, $a2
      b fine
moltiplicazione: mulou $v0, $a1, $a2
      b fine
divisione: divu $v0, $a1, $a2
      b fine
fine: jr $ra
```