

Nome, cognome, matricola .....

## Calcolatori Elettronici (12AGA) –esame del 1.7.2020

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

Tempo: 15 minuti.

1	Si consideri un circuito sequenziale sincrono avente 30 ingressi, 20 uscite e 40 stati. Quanti flip flop sono necessari per la sua implementazione?		
2	Quanti flip flop di tipo D sono necessari per costruire un contatore sincrono su 32 bit?	0	A
		1	B
		32	C
		64	D
3	In un sistema dotato di DMA Controller, chi si fa carico della sua programmazione?	L'Interrupt Controller	A
		La CPU	B
		Le periferiche	C
		L'arbitro del bus	D
4	Si consideri il meccanismo della parità, utilizzato per aumentare l'affidabilità delle DRAM: quando viene calcolato il bit di parità associato ad una parola?	Quando la parola viene scritta in memoria	A
		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
5	Considerando il sistema di arbitraggio distribuito del bus utilizzato dallo standard SCSI, quale delle seguenti affermazioni è vera?	Il sistema richiede $\log n$ linee per la gestione di $n$ unità master	A
		Il sistema non prevede la presenza di un arbitro	B
		Il sistema permette di cambiare con facilità la priorità delle unità master	C
		L'arbitro è molto semplice	D
6	Si consideri un processore dotato di una cache con le seguenti caratteristiche · 128 linee da 16 byte ciascuna · direct mapping · write-through. Assumendo che il processore (avente uno spazio di indirizzamento di $2^{32}$ parole) voglia accedere alla parola di indirizzo 0011 1100 1101 1111 0011 0100 0001 0111, a quale linea della cache farà accesso nel caso in cui si sia prodotto un hit?		
7	Considerando i processori RISC, quale delle seguenti disuguaglianze non è <u>mai</u> vera, essendo K il numero di stadi della pipeline?	$CPI < 1$	A
		$CPI = 1$	B
		$CPI > 1$	C
		$CPI = K$	D

8	In un processore MIPS, a quale tipo appartiene l'istruzione <code>beq</code> ?	Al tipo J	A	
		Al tipo R	B	
		Al tipo I	C	
		Al tipo B	D	

9	Nella programmazione in linguaggio assembler MIPS si fa spesso uso di pseudo-istruzioni: chi definisce l'insieme delle pseudo-istruzioni disponibili?	Il progettista del processore	A	
		Lo sviluppatore dell'assemblatore	B	
		Lo sviluppatore del debugger	C	
		Il programmatore	D	

10	Quale valore (in esadecimale) sarà presente in <code>\$t0</code> dopo l'esecuzione dell'istruzione <code>ori \$t0, \$t1, 12</code> assumendo che <code>\$t1</code> contenga il valore 1?	
----	--	--

# Risposte corrette

1	2	3	4	5	6	7	8	9	10
6	c	b	c	<u>b</u>	65	a	c	b	D

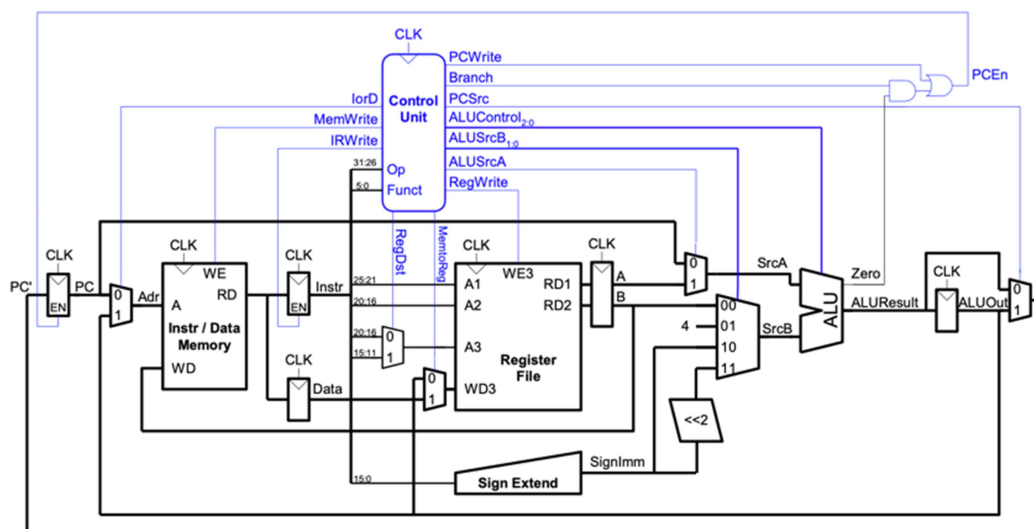
Nome, cognome, matricola .....

**Compito A**

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

11	Si descrivano le funzionalità offerte dal DMA Controller e si illustrino le modalità e i passaggi attraverso i quali avviene un trasferimento in DMA, partendo dalla fase di programmazione.
----	--

12	Utilizzando la tabella riportata, si elenchino le micro-operazioni eseguite da un processore MIPS durante la fase di fetch e l'esecuzione dell'istruzione lw \$s1, 12(\$s2) che assegna al registro destinazione \$s1 il valore contenuto all'indirizzo di memoria (\$s2 + 12)
----	--

[illegible]

13	Si descrivano le differenti soluzioni di arbitraggio di un bus elencandone vantaggi e svantaggi.
14	Si illustri un esempio di articolazione in stadi di un'architettura pipeline (spiegando la funzione di ciascuno stadio) e si descrivano le principali cause di stallo e le principali contromisure software.

## Esercizio di programmazione

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

Si scriva una procedura `media` in linguaggio Assembly MIPS32 che esegua una elaborazione degli elementi contenuti in un vettore di interi `veta` di dimensione nota `DIM` in cui sono memorizzati i valori numerici conseguenti al campionamento di un segnale analogico.

Il programma dovrà svolgere un'elaborazione sui dati contenuti in `veta` e scrivere il risultato in un vettore di interi `vetb` di uguale dimensione, effettuando le seguenti operazioni.

1) I primi due elementi in `vetb` rimangono invariati rispetto ai corrispondenti elementi in `veta`:

```
vetb[0] = veta[0]
```

```
vetb[1] = veta[1]
```

2) L'elemento in posizione  $i$ -esima di `vetb` (con  $1 < i < DIM-1$ ) è posto pari alla media aritmetica calcolata sui due valori precedenti di indice  $i-2$  e  $i-1$ , sul valore corrente di indice  $i$  e sul valore successivo di indice  $i+1$  in `veta`.

```
vetb[i] = (veta[i-2] + veta[i-1] + veta[i] + veta[i+1]) / 4
```

3) L'ultimo elemento di `vetb` rimane invariato rispetto al corrispondente elemento in `veta`:

```
vetb[DIM-1] = veta[DIM-1]
```

La procedura deve correttamente gestire i casi particolari in cui il vettore `veta` ha 1, 2 o 3 elementi.

Esempio:

<i>posizione</i>	<i>veta</i>	<i>vetb</i>
0	2	2
1	14	14
2	8	19
3	54	19
4	0	26
5	42	26
6	9	18
7	24	18
8	0	31
9	91	34
10	23	23

I parametri sono passati alla procedura attraverso i registri:

- **\$a0**: contiene l'indirizzo di `veta`
- **\$a1**: contiene l'indirizzo di `vetb`
- **\$a2**: contiene il numero di elementi di ciascun vettore (valore intero strettamente maggiore di 0).

Si lavori nell'ipotesi di non avere *overflow* durante i calcoli.

Di seguito un esempio di programma chiamante:

```

DIM = 11
.data
veta:    .word 2, 14, 8, 54, 0, 42, 9, 24, 0, 91, 23
vetb:    .byte 4*DIM
.text
.globl main
.ent main
main:    [...]
        la $a0, veta
        la $a1, vetb
        li $a2, DIM
        jal media
        [...]
        .end main

```