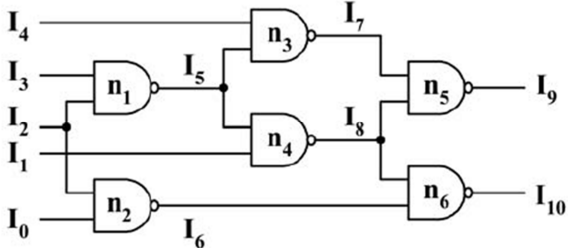


Calcolatori Elettronici (12AGA) – esame del 21.6.2016 - A**Domande a risposta chiusa** (è necessario rispondere correttamente ad almeno 6 domande).

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

1	A quanto ammonta il ritardo del circuito qui sotto, assumendo che ogni porta abbia un ritardo di 5ns?			
2	Quale vantaggio presenta un sommatore di tipo carry lookahead rispetto a un ripple carry adder?	È più veloce	A	
		Richiede un hardware più semplice	B	
		È più facile da progettare	C	
		È più robusto ai guasti	D	
3	Si consideri il meccanismo della Memoria Virtuale: che cosa contiene il TLB?	Le ultime pagine di memoria accedute	A	
		L'elenco delle pagine presenti in memoria principale	B	
		L'elenco delle pagine accessibili dal processo in corso di esecuzione	C	
		Le ultime entry della MAT accedute	D	
4	Si confronti una memoria RAM e una Flash: quale svantaggio presenta la seconda?	Non è volatile	A	
		Le operazioni di scrittura sono più lente di quelle di lettura	B	
		Non può essere riscritta	C	
		Il meccanismo di accesso non è casuale	D	
5	Che differenza c'è tra le istruzioni ADD e ADC?	ADD somma due valori senza modificare i flag; ADC invece li modifica sulla base del risultato prodotto	A	
		ADD somma due valori su 16 bit; ADC somma invece due valori su 32 bit	B	
		ADD somma due valori su 8 o 16 bit; ADC somma un'ulteriore unità se il carry flag vale 1	C	
		Nessuna differenza: le due istruzioni eseguono le stesse operazioni	D	
6	Qual è il numero massimo di periferiche che può gestire un singolo 8259?	1	A	
		3	B	
		8	C	
		64	D	
7	Quanti segnali sono necessari per gestire l'arbitraggio in un sistema composto da 20 moduli che utilizza la tecnica del polling?	3	A	
		7	B	
		22	C	
		41	D	
8	Quanti byte occupa il codice macchina dell'istruzione MOV AX, 300?	1	A	
		2	B	
		3	C	
		6	D	
9	Si assuma che ad un certo istante i registri di un processore x86 abbiano il seguente contenuto (espresso in esadecimale): DS=0010, BX=0010. Qual è l'indirizzo fisico del byte di memoria in cui scrive 0 il frammento di codice seguente? MOV SI, 8 MOV [BX+SI+2], 0			

Risposte corrette

1	2	3	4	5	6	7	8	9
15ns	A	D	B	C	C	B	C	282

Domanda 9

DS = 0000 0000 0001 0000

BX = 0000 0000 0001 0000

SI = 0000 0000 0000 1000

EA = 0000 0000 0001 0000 0000 +
 0000 0000 0001 0000 +
 0000 0000 0000 1000 +
 0000 0000 0000 0010 =
 0000 0000 0001 0001 1010 = 256 + 16 + 8 + 2 = 282 = 11Ah

Nome, cognome, matricola

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

10	Si progetti un circuito combinatorio minimo avente 4 ingressi a, b, c e d e un'uscita u che va a 1 se e solo se $a=c$ OR $b=d$.
11	Si disegni l'architettura di un'unità di controllo microprogrammata e si elenchino vantaggi e svantaggi rispetto alla soluzione cablata.

12	Si descriva il funzionamento di una porta di un 8255 programmata in input in modo 1.
----	--

13	<p>Si consideri una cache direct mapped composta da 8 linee</p> <p>Si identifichi per ogni linea il blocco contenuto nella cache ad un certo istante, assumendo che la memoria sia composta da 64 blocchi e nel periodo immediatamente precedente il processore abbia fatto accesso nell'ordine ai seguenti blocchi: 21, 28, 30, 10, 7, 60, 30, 31, 32, 33, 60, 61, 25, 28, 27, 14. Si scriva il risultato nel disegno sotto.</p>
----	---

0	
1	
2	
3	
4	
5	
6	
7	

Esercizio di programmazione

sino a 12 punti – è possibile consultare qualunque materiale cartaceo - tempo: 60 minuti

Siano date due matrici di interi positivi memorizzate per righe:

Matrice 1:

<i>riga 0:</i>	1	2	3	4
<i>riga 1:</i>	5	6	<u>7</u>	8
<i>riga 2:</i>	9	10	11	12
<i>riga 3:</i>	13	14	15	16

Matrice 2:

<i>riga 0:</i>	9	3	4	11
<i>riga 1:</i>	1	5	6	8
<i>riga 2:</i>	16	15	2	12
<i>riga 3:</i>	<u>7</u>	10	13	14

Data l'indicazione dell'indice di una riga per ciascuna matrice (ad esempio, 1 e 3, rispettivamente), si scriva in linguaggio Assembly 8086 una procedura `trova_num` in grado di determinare il valore dell'elemento presente in ciascuna delle due matrici alle righe indicate (in questo caso, 7).

La procedura riceve mediante lo *stack* gli indirizzi di due vettori di DIMxDIM *byte* (DIM dichiarato come costante) contenenti le due matrici, e gli indici di riga mediante AL e AH. Il risultato, ossia il valore trovato, deve essere restituito tramite lo *stack*.

Se non ci sono elementi in comune, la procedura deve restituire il valore -1. Se ci sono più elementi in comune, la procedura ne restituisca uno qualunque fra essi.

Di seguito un esempio di programma chiamante:

```

DIM          EQU 4
              .model small
              .stack
              .data
mat1          db 1, 2, 3, 4
              db 5, 6, 7, 8
              db 9, 10, 11, 12
              db 13, 14, 15, 16
mat2          db 9, 3, 4, 11
              db 1, 5, 6, 8
              db 16, 15, 2, 12
              db 7, 10, 13, 14

              .code
              .startup
push offset mat1
push offset mat2
sub sp, 2
mov al, 1      ; indice riga mat1
mov ah, 3      ; indice riga mat2
call trova_num
pop ax         ; risultato
add sp, 4

.exit

```