

Non saranno corretti elaborati scritti confusamente o con grafia difficilmente leggibile

Nome:

Cognome:

Matricola:

2 Punti

1. Giustificando in modo preciso la risposta, si dica se le seguenti espressioni booleane sono tra loro equivalenti (ovvero, identificano la stessa funzione delle variabili booleane A, B, C):

$$\overline{A + B \cdot C} \quad \text{e} \quad \overline{A} \cdot (\overline{B} + \overline{C}) + \overline{A} \cdot \overline{B}$$

Solution: Usando l'algebra Booleana, si può dimostrare che le due espressioni sono equivalenti. Infatti applicando De Morgan risulta $\overline{A + B \cdot C} = \overline{A} \cdot (\overline{B} + \overline{C})$, che con ulteriori passaggi algebrici risulta uguale a $\overline{A} \cdot \overline{B} + \overline{A} \cdot \overline{C} = \overline{A} \cdot \overline{B} + \overline{A} \cdot \overline{C} + \overline{A} \cdot \overline{B} = \overline{A} \cdot (\overline{B} + \overline{C}) + \overline{A} \cdot \overline{B}$.

In alternativa, si può dimostrare l'equivalenza utilizzando le tabelle di verità. I valori di entrambe le espressioni per le 8 combinazioni degli ingressi, nell'ordine 000, 001, 010, 011, 100, 101, 110, 111 sono 1, 1, 1, 0, 0, 0, 0, 0.

3 Punti

2. Mostrando i passaggi fondamentali:

- rappresentare i numeri decimali -54 e -98 in complemento a due con 8 bit;
- eseguire la somma algebrica dei numeri ottenuti e riportare (separatamente) il risultato;
- indicare l'eventuale presenza di overflow e come rilevarla.

Solution: $54_{10} = 00110110_2$. $-54 = 11001010_{ca2}$.

$98_{10} = 01100010_2$. $-98 = 10011110_{ca2}$.

Somma:

$$\begin{array}{r} 11001010 \quad + \\ 10011110 \quad = \\ (1)01101000 \end{array}$$

Risultato: 01101000.

C'è overflow perchè si ha riporto dalla cifra più significativa ma non dalla seconda più significativa; in modo equivalente perchè dalla somma di due numeri negativi si ottiene un risultato positivo.

3 Punti

3. Mostrando i passaggi fondamentali, ricavare il valore decimale del seguente numero in virgola mobile rappresentato secondo lo standard IEEE 754 a 32 bit:

1 10000010 0010000000000000000000

Solution:

Segno negativo.

$E = 10000010_2 = 130_{10}$. Quindi $esp = 130 - 127 = 3$.

Di conseguenza il numero rappresentato è $-1.001 * 2^3 = -1001 = -9_{10}$.

2 Punti

4. Che cos'è il codice operativo di una istruzione macchina? E cosa sono gli operandi?

3 Punti

5. Nell'ambito dei sistemi operativi multitasking, che cos'è il salvataggio del contesto di un processo? Riportare gli elementi principali che fanno parte del contesto.

3 Punti

6. Illustrare il significato ed il ruolo degli indirizzi IP. Quali sono le differenze tra IP statico ed IP dinamico?

5 Punti

7. Scrivere un programma C che richiede all'utente l'inserimento di una serie di numeri interi (l'acquisizione termina quando l'utente inserisce il valore 0) e stampa la somma dei numeri compresi tra 10 e 20 estremi inclusi (se non ce ne sono il programma stampa 0).
Ad esempio, se l'utente inserisce la sequenza

7, 25, 13, 10, 20, 70, 0

il programma termina stampando 43.

Solution:

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 main()
5 {
6     int num, somma;
7     somma=0;
8     do{
9         scanf("%d", &num);
10        if (num>=10 && num<=20)
11            somma=somma+num;
12    } while (num!=0);
13    printf("somma= %d\n", somma);
14    system("PAUSE");
15 }
```

9 Punti

8. Scrivere un programma C che, ricevendo in ingresso una sequenza di lunghezza arbitraria di numeri interi diversi da zero, terminata da uno zero, produca in uscita la sequenza di tre elementi contigui la cui somma è massima (se sono presenti più sequenze con somma massima, è sufficiente stampare la prima). Per semplicità, si assuma che la sequenza inserita comprenda almeno tre numeri interi oltre lo zero.

Ad esempio, se l'utente inserisce la sequenza

7, 2, 19, 4, 45, 3, 7, 9, 3, 0

il programma termina e stampa 19, 4, 45.

Solution:

```
1
2 #include <stdio.h>
3 #include <stdlib.h>
4
5 main(){
6     int max1, max2, max3, num, ult1, ult2, ult3;
7
8     scanf("%d", &max3);
9     scanf("%d", &max2);
10    scanf("%d", &max1);
11    ult1=max1;
12    ult2=max2;
13    ult3=max3;
14
15    scanf("%d", &num);
16    while(num!=0){
17        ult3=ult2;
18        ult2=ult1;
19        ult1=num;
20        if(ult1+ult2+ult3 > max1+max2+max3){
21            max1=ult1;
22            max2=ult2;
23            max3=ult3;
24        }
25        scanf("%d", &num);
26    }
27
28    printf("Numeri: %d, %d, %d\n", max3, max2, max1);
29    system("PAUSE");
30 }
```