

Corso di Laurea in Ingegneria Elettronica, a.a. 2001-02
Modulo di "Informatica 1"

Prova d'esame del 17 aprile 2002 - Fila unica

durata della prova: 2h

- Domande a risposta libera -

1. Cosa è la *Macchina di von Neumann*?
2. Che differenza c'è tra la memoria centrale e la memoria di massa?
3. A cosa servono i *linguaggi di programmazione*?
4. Descrivere i tipici *flussi di un programma*
5. Assumendo che 0=falso e 1=vero e posto

a=0, b=1, c=0

x=2, y=2

Determinare i valori delle seguenti espressioni logiche:

NOT(a AND c) OR NOT(b)

(x<y) AND (y==2)

- Esercizi sul linguaggio 'C' -

N.B. Nel caso si ritenga necessario formulare ipotesi aggiuntive è importante indicarle chiaramente e motivarle.

Scrivere, soprattutto il codice, in modo leggibile

1. Scrivere un programma che legga da tastiera una serie di numeri interi terminata da 0 e stampi un asterisco per ogni numero inserito multiplo di 3.
2. Descrivere l'output generato dal seguente programma:

```
#include <stdio.h>

main()
{
    int i,x;
    i=0; x=0;
    for (i=1; i<10; i++) {
        if (i%2 == 1)
            x=x+i;
        else
            x--;
        printf("%d ", x);
    }
    printf("\nx = %d", x);
}
```

3. Siano dati in input le coordinate (x,y) di punti (P1, P2, ..., Pn) di un piano. Definire una struttura per rappresentare le coordinate di tali punti e una funzione `check` che decida se tre punti passati per argomento sono allineati. Sviluppare un programma completo che utilizzi questa funzione.
Osservazione: Tre punti P1(x1,y1), P2(x2,y2) e P3(x3,y3), risultano allineati se verificano la seguente condizione: $(y2-y1)/(x2-x1) = (y3-y1)/(x3-x1)$.
4. Scrivere un programma che scriva sul file `out.txt` un numero di asterischi pari ad un numero letto da tastiera.
5. Scrivere un frammento di codice per gestire la memorizzazione di una sequenza di numeri interi la cui lunghezza deve poter essere specificata dall'utente in fase di esecuzione del programma.