

**FONDAMENTI DI INFORMATICA I**  
**Corso di Laurea in Ingegneria Elettronica, Informatica, delle Telecomunicazioni**  
**Università La Sapienza - Consorzio Nettuno**  
**Appello del 08-03-2003**  
**Parte Pratica (20 punti)**

**0)**

Scrivere un programma C che legga da input una sequenza di almeno 2 numeri double positivi, interrotta da 0.0 e calcoli la massima lunghezza delle sottosequenze strettamente decrescenti di numeri.

Ad esempio, se la sequenza di input è

7.1 3.2 6.0 5.1 4.9 3.0 2.1 2.2 2.5 3.0 2.2 13.1 9.0 8.0 1.0 0.0

le sottosequenze decrescenti sono quelle sottolineate e la lunghezza massima è 5.

**1)**

Scrivere una funzione C `eser1()` che, ricevendo un intero `n`, legge da standard input una sequenza di `n` parole (stringhe di caratteri prive di spazi bianchi) e ne stampa solo quelle di lunghezza massima.

Ogni parola è di al più 30 caratteri.

La soluzione deve essere proposta in due modi, sia facendo uso della funzione `strlen()`, sia non facendone uso.

Esempio,

**sequenza di input:**

porta montata male che non si chiude proprio bene

**output:**

montata, proprio

**2)**

Scrivere una funzione C `eser2()` che,

- ricevendo un intero `n`, un array `arr` di `n` interi e quanti altri parametri si ritenga opportuno,
- restituisca
  - o la massima distanza tra elementi contigui dell'array;
  - o un elemento mediano dell'array, cioè tale da avere tanti elementi minori quanti maggiori (+/- 1 nel caso sia `n` dispari)

Scrivere poi un programma C che, usando la funzione `eser2()`, e tutte le funzioni ulteriori che si ritenga opportuno definire, legga da input un array di interi, `arr`, lo stampi per controllo e lo passi ad `eser2()`, stampando poi i dati restituiti da `eser2()`.

Si richiede una specifica esauriente di dati di test suggeriti per provare il funzionamento del programma.

**Es.** se l'array è **32 24 51 29 25 49 33 55**

la massima distanza tra due elementi contigui è **27**;

l'elemento mediano è **32**

**FONDAMENTI DI INFORMATICA I**  
**Corso di Laurea in Ingegneria Elettronica, Informatica, delle Telecomunicazioni**  
**Università La Sapienza - Consorzio Nettuno**  
**Appello del 08-03-2003**

**Parte teorica (10 punti)**

**I) (1/2)**

Siano  $N=121$ ,  $M=591$ .

- 1) Stabilire il numero minimo  $B$  tale che sia  $N$  che  $M$  possano essere rappresentati come numeri binari puri con  $B$  cifre;
- 2) scrivere la rappresentazione di  $N$  ed  $M$  come numeri binari puri con  $B$  cifre ed eseguire la somma  $N+M$ ;
- 3) Stabilire il numero minimo di cifre necessarie per rappresentare tanto  $N$  quanto  $-M$  in complemento a due e mostrare tali rappresentazioni, dettagliando il procedimento usato per ottenerle;
- 4) infine eseguire l'operazione  $N-M$ , e scrivere il risultato usando 16 bit.

**II) (1/2)**

Descrivere il comportamento del seguente programma, spiegando anche dettagliatamente la successione delle chiamate ricorsive.

```
#include <stdio.h>
#define MAX 8

int che(int *, int);

int main() {
    int a[MAX], i;

    printf("inserire le prime %d cifre del numero di matr.", MAX );
    for(i=0; i<MAX; i++)
        scanf("%d", a+i);
    printf("--- %d ---\n", che(a,2));
    printf("\nFINE\n");
    return 0;
}

int che(int arr[], int i) {
    if (i >= MAX)
        return 0;
    else
        return (arr[i] + che(arr, i+1));
}
```