



**Programmazione 1 – CdS in Matematica  
Appello d'esame 20 Luglio 2007**

**Nome** .....  
**Cognome** .....  
**Matricola** .....

**INDICARE NOME, COGNOME, E MATRICOLA SU OGNI FOGLIO UTILIZZATO**

**NON è permesso (pena espulsione) usare la calcolatrice e consultare appunti e libri.  
Scrivere le risposte e commentare i programmi CHIARAMENTE (la chiarezza sarà un  
criterio determinante nella valutazione degli esercizi).**

**----- PARTE 1 -----**

**Esercizio 1.1**

Scrivere l'output del seguente programma C. (Indicare con '\_' underscore gli eventuali spazi e con '<n>' la riga vuota)

```
#include<stdio.h>

int main() {

    double x_begin = 1.0, x_end = 2.0;
    for ( ; x_begin < x_end; x_begin += .2)
        printf("%d ", x_begin);
    printf("\n");

    int b=3;
    if (b=1) b++;
    printf("b= %d\n", b);

    int i=1, j=2, k=3;
    int e=(i+j+k== -2*-k)-1;
    printf("e= %d\n", e);

    char mia_stringa[] = "voglio andare al mare!";
    int n;
    for ( n=20 ; n>=0 ; n-=2)
        printf("%c", mia_stringa[n]);
    printf("\n");
}
```

## Esercizio 1.2

Descrivere la sintassi e l'utilizzo del costrutto IF (risp. WHILE).

## Esercizio 1.3

Descrivere la "chiamata per valore" in C.

## Esercizio 1.4

Descrivere (anche graficamente) l'effetto del seguente frammento di programma.

```
int n=2;
float *ptr;
float *pippo = malloc(n*sizeof(float));
ptr = pippo++;
```

## Esercizio 1.5

Descrivere le funzionalità (cosa calcola) e discutere le assunzioni che devono essere verificate sui parametri delle seguenti funzioni. Esempificare il loro utilizzo.

**A)**

```
void Aldo(char *a) {
    int n=strlen(a);
    char *p = a;
    int i=0;
    for (i=0; i<n; i++)
        printf("%c", *p++)
    while (p > a)
        printf("%c", *(--p));
}
```

**B)**

```
float Osvaldo(float *a, int n) {
    if (n==0) return 0;
    else return a[0]+Osvaldo(++a, n-1);
}
```

## Esercizio 1.6

Realizzare un programma che richiede un intero (n) e stampi la sua "tabellina".

Esempio con n=3:

3 6 9 12 15 18 21 24 27 30

## ----- PARTE 2 -----

### Esercizio 2.1

Dare un programma

```
double CalcolaSerie(int n);
```

il più efficiente possibile (rispetto al numero di operazioni effettuate) per calcolare con precisione arbitraria il valore di convergenza della serie:

$$S_n = \sum_{i=1}^n \frac{\sqrt{i+1} - \sqrt{i}}{\sqrt{i^2 + i}}$$

### Esercizio 2.3

Descrivere l'algoritmo e dare un esempio di ordinamento BUBBLESORT.

### Esercizio 2.4

Descrivere una possibile implementazione e dare le strutture dati per il gioco del "Filetto".

### Esercizio 2.5

Sia data la seguente struttura dati LISTA

```
struct ELEM {
    int dato;
    struct ELEM *prox;
};

typedef struct ELEM ElementoLista;
typedef ElementoLista *Lista;
```

Si implementino le seguenti funzioni:

```
Lista InserisciInCoda(Lista lista, int dato);
/* Inserisce un nuovo dato alla fine della lista data come
parametro e ritorna la nuova lista*/

Lista CancellaUltimo(Lista lista);
/* Cancella l'ultimo elemento della lista e ritorna la nuova
lista*/
```

```
Lista CercaInLista(Lista lista, int dato);  
/* Ritorna il puntatore al primo elemento dove occorre il dato o  
NULL se non c'è */  
  
void StampaLista(Lista lista);  
/* Stampa i dati contenuti nella lista */  
  
int MoltLista(Lista lista);  
/* Ritorna la moltiplicazione degli elementi della lista */
```

## Esercizio 2.6

Si supponga che N sia stata precedentemente definita con una #define, si implementino le seguenti funzioni:

a) Trasposizione di una matrice

```
void Trasposizione(float mat[N][N], float tr_mat[N][N]);
```

b) Prodotto scalare

$$s = v1*v2$$

```
float ProdScalare(float v1[N], float v2[N]);
```