

## Funzioni e Parametri di default

Molte funzioni includono argomenti ai quali si vorrebbe assegnare, se non diversamente specificato, un valore di **default**.

A questi argomenti, che possono essere omessi nella chiamata alla funzione, può essere assegnato un valore di default nel prototipo della funzione.

### Esempio

Definisco la funzione

```
int f(int a, int b){  
    return a+b;  
}
```

Prima del main:

```
int f(int a, int b=0); // prototipo di f
```

Se nel main scrivo:

```
a = f(3,2); // a = 5
```

Ma se scrivo

```
a = f(3); // a=3 perché b se non specificato vale 0
```

**N.B.** L'indicazione del valore di default va messa **SOLO** nel **prototipo**

**N.B.** Gli argomenti di default **DEVONO** essere gli ultimi sulla destra e vanno omessi ordinatamente da quello più a destra a scalare, senza lasciare "buchi".

## Funzioni e Variabili *Static*

Se in una funzione si dichiara una variabile “static” questa, dopo l’uscita del programma dalla funzione, non verrà cancellata ma resterà a disposizione per la chiamata successiva della funzione.

### Esempio

```
int f(int a, int b){  
    static int n;  
    n++;  
    cout << n << endl;  
    return a+b;  
}
```

se non inizializzata diversamente verrà inizializzata a zero

## Puntatori a Funzioni

### Esempi di utilizzo

- vogliamo che una funzione accetti un’altra funzione come argomento in input
- definito un puntatore voglio farlo puntare ad una tra più funzioni disponibili

### Esempio

Ho diverse successioni. Di ognuna voglio calcolare la somma dei primi M termini. Costruisco una funzione che, preso in ingresso N e la funzione che descrive il termine n-esimo, calcoli la somma.

(vedi esempio)