



# Dichiarazione di funzioni

Corso di Informatica  
Laurea in Fisica

prof. ing. Corrado Santoro

# Quando ci servono le funzioni...



- Calcoliamo il coefficiente binomiale:

$$\binom{n}{k} = \frac{n!}{k! (n-k)!}$$

- Dobbiamo calcolare 3 volte il fattoriale di 3 numeri diversi
- Sappiamo scrivere l'algoritmo del fattoriale
- Allora lo inseriamo 3 volte nel nostro flow-chart
- **NO! SCRIVIAMO UNA *FUNZIONE GENERALE* PER CALCOLARE IL FATTORIALE E LA RICHIAMIAMO 3 VOLTE**

# Supponiamo di possedere una funzione "fatt(n)"



```
#include <stdio.h>

/* fatt(n) richiede un float e restituisce un float */

main()
{
    float n,k;
    printf("Inserisci il valore di 'n' ");
    scanf("%f", &n);
    printf("Inserisci il valore di 'k' ");
    scanf("%f", &k);
    getchar();
    if ((n < 0) || (k < 0) || (n < k))
        printf("Errore! n e k devono > 0, e n >= k\n");
    else
        printf("Il coefficiente binomiale calcolato e' %f\n",
                fatt(n) / (fatt(k) * fatt(n-k)));
    getchar();
}
```

# Dichiariamo la funzione "fatt(n)"



```
/* fatt(n) richiede un float e restituisce un float */  
  
float fatt(float n)  
{  
    float risultato = 1;  
    while (n > 1) {  
        risultato = risultato * n;  
        n = n - 1;  
    }  
    return risultato;  
}
```

- In generale una funzione si dichiara con:
  - **tipo\_di\_ritorno nome\_funz(tipo var1, tipo var2, ...)**
- **var1, var2**, etc. saranno assegnate dal chiamante e sono i parametri della funzione
- Il risultato della funzione, restituito al chiamante, si specifica con l'istruzione **return**

# Funzioni di più variabili "power(a, n)"



```
float power(float a, int n)
{
    float potenza;
    int esponente;

    if (n < 0) esponente = -n;
    else esponente = n;

    potenza = 1;
    while (esponente > 0) {
        potenza = potenza * a;
        esponente = esponente - 1;
    }

    if (n < 0)
        return 1.0 / potenza;
    else
        return potenza;
}
```

# Dichiarare prima di usare!



- In generale, in un programma C, una funzione deve essere scritta **prima** che essa venga utilizzata
- A volte però capita che ciò non sia possibile e che la funzione debba venir scritta **dopo** il suo utilizzo: come fare?
  1. Si inserisce, prima dell'utilizzo, il **prototipo della funzione**
    - **tipo\_di\_ritorno nome\_funz(tipo var1, tipo var2, ...);**
  2. si usa la funzione
  3. si inserisce la sua **implementazione**

# Esempio



```
#include <stdio.h>

float fatt(float n);

main()
{
    float n,k;
    printf("Inserisci il valore di 'n' ");
    scanf("%f", &n);
    ...
    if ((n < 0) || (k < 0) || (n < k))
        printf("Errore! n e k devono > 0, e n >= k\n");
    else
        printf("Il coefficiente binomiale calcolato e' %f\n",
                fatt(n) / (fatt(k) * fatt(n-k)));
    getchar();
}

float fatt(float n)
{
    float risultato = 1;
    ...
    return risultato;
}
```