

## Costanti

Se il dato non deve cambiare

costanti - tipo,nome per costanti

Cons nometipo identificatore=valore

Operazioni aritmetici/espressioni

% modulo o resto per le divisioni intere

>> operatore di output

<< operatore di input

cin>>variabile>>variabile2

feet=35 oppure cin>>feet; sapete che ti dà un valore interno e quando te lo dà aggiungi dentro questa variabile "feet".

"g++ area.cpp -o area"

## Funzioni

1 ottenere input

2 risolvere problema

3 generare output

In matematica: funzione prevede un argomento produce un risultato

Es  $f(x) = x^2 + 2$       $f(7) = 51$

In informatica

Funzione ha un tipo che è il tipo del valore che produce, ha un nome usato per definirla e invocarla( chiamarla)

Ha una lista di parametri: nomi e tipi di valori che prende in input

Un corpo: insieme delle istruzioni che specificano cosa fa, usa i parametri cose se fossero variabili.

Funzioni Possono avere effetti collaterali

Sostanzialmente di due tipi

1. Generare output verso cout o altro stream
2. Cambiare stato della programma

I primi tipi di funzioni che vedremo non cambiano lo stato.

La sintassi in c++

"matematica

Definizione della funzione

E invocazione della funzione"

Definizione di funzione: <tipo> <nome>(<lista parametri\_formali>){

<corpo>

}

<tipo> sia float,int etc

<Nome> è un identificatore scelto dal programmatore

(<lista parametri> e una lista del tipo <tipo1> <nome1>, <tipo2> <nome2> etc.  
<corpo> istruzioni.

Esempio: calcolo dell'area del rettangolo

```
Float area_rettangolo(float b,float h)
{
    Return Bxh; //effetto nullo,il valore deve essere restituito //
    Return //restituisce in output, buta fuori, ora funziona //
}
```

Chiamata di funzione ;  
<nome\_funzione>(<lista parametri\_attuali>)  
area\_rett(8.2,4.5)  
Posso anche scrivere  
Float base=8.2, altezza=4,5, area;  
area= altezza\_rettangolo(base,altezza);