

- Per risolvere un problema complesso è necessario scomporlo in parti che siano facilmente maneggevoli.
- Il costrutto di funzione viene impiegato in C per realizzare questa metodologia di programmazione.
- Ogni programma deve essere pensato e realizzato come una sequenza di funzioni.
- Un programma è costituito da uno o più file, ognuno dei quali contiene zero o più funzioni, di cui una di nome `main( )`.
- L'esecuzione del programma inizia dalla funzione `main()`.
  
- La **dichiarazione** di una funzione avviene indicando il tipo del risultato, il nome della funzione e il tipo delle variabili che servono per poter operare.

Ad esempio

```
float area( float, float);
```

è la dichiarazione di una funzione che avuto in ingresso la base e l'altezza di un rettangolo restituisce l'area.

- La **definizione** di funzione è il codice C che descrive ciò che viene compiuto da una funzione. Ha la seguente forma:

```
tipo nome-della-funzione (lista-dei-parametri) {  
    dichiarazioni  
    istruzioni  
}
```

- l'intestazione della definizione di funzione è costituita da tutto ciò che precede la parentesi graffa aperta
- il corpo della definizione di funzione è formato da ciò che si trova racchiuso tra graffe. E' un'istruzione composta che può contenere dichiarazioni.
- l'elenco dei parametri è tutto quello che si trova tra le due parentesi rotonde. E' una lista in cui compare il tipo del parametro e il nome
- Se non compare la dichiarazione del tipo la funzione è considerata di tipo int
- Una funzione che non ha parametri contiene void come lista dei parametri
- Una funzione che non restituisce alcun valore è di tipo void

```

/* stampa una sequenza di caratteri */
void print-sequenza(void) {
    printf(“%s\n%s\n%s\n%s\n”,
           “*****”,
           “    Lorenzo Porcelli”,
           “e_mail: genna18@iol.it”,
           “*****”
    )
}

```

```

/* area di un rettangolo */

float area ( float lato1, float lato2) {
    return lato1 * lato2;
}

```

```

/* risolvere un'equazione di 1° grado */

float eq1grado( float a, float b ) { /* b diverso 0 */
    return -b / a;
}

```

```

/*
Fattoriale di un numero
*/
long fattoriale( int n ) {
    long fatt=1;

    for(;n > 1; n--)
        fatt *= n;
    return fatt;
}

```

```

/*
x^n con n intero positivo
*/

float esp( float base, int n ) {
    int i;
    float risultato=1;

    for(i=0; i < n; i++)
        risultato = risultato * base;
    return risultato;
}

/*
Determinare il volume di un parallelepipedo*/
*/
float leggi( void ) {
    float x;
    do {
        printf("Inserire la misura del lato[0 per finire] ");
        scanf("%f", &x);
    } while (x < 0);
    return x;
}

void scrivi( float a, float b, float c ) {
    printf("Lato 1 = %f\nLato 2 = %f\n
        Lato 3 = %f\n\nVolume = %f\n", a, b, c, a * b * c);
}

void main( void ) {
    float a,b, c;

    while( (a = leggi()) && (b = leggi()) && (c = leggi()) )
        scrivi(a, b, c);
}

```