

# Corso sul linguaggio C

## Modulo 1

### 2 – La programmazione in C

M. Malatesta 2-La programmazione in C-02

1  
09/03/2014

## Prerequisiti

- Saper scrivere semplici programmi in C
- Saper usare un sistema operativo

M. Malatesta 2-La programmazione in C-02

2  
09/03/2014

# Introduzione

In questa lezione si descrivono le operazioni da svolgere per eseguire un programma in C al computer.

Ovviamente per esercitarsi è necessario che sul pc sia presente un compilatore C di qualche tipo (**Microsoft, Borland**, C standard sotto **Linux**, Dev-Cpp in ambiente Windows, ...)

# Compilazione

Il programma scritto (**programma sorgente**) mediante un **editor** deve essere salvato su disco (con estensione **.c**) e sottoposto successivamente alla compilazione per ottenere un programma eseguibile.

Il compilatore è un programma che:

- **rileva eventuali errori** nella sintassi delle istruzioni, ossia nel modo in cui le istruzioni sono state scritte e segnala il punto in cui si sono verificati. Il programmatore dovrà correggere gli errori e *compilare di nuovo il programma sorgente*.
- quando il programma non avrà più errori sintattici, **effettua la sua traduzione da programma sorgente a programma oggetto (.obj)**

# Compilazione

Il programma oggetto è la pura traduzione in binario del programma sorgente; in essa, mancano le espansioni delle funzioni utilizzate (ad esempio **printf()** e **scanf()**), per cui il programma non è ancora eseguibile dalla CPU.

È allora necessario collegare il programma **obj** con le librerie di sistema (tipicamente i file “.lib” di **UNIX**); per questo compito, interviene il *linker* che produce così il programma eseguibile (**a.out**).

A questo punto il programma è contenuto in un file scritto in linguaggio binario direttamente comprensibile dall’elaboratore.

# Compilazione

La compilazione può essere svolta:

- da linea di comando
- tramite i comandi di un IDE (ambiente di sviluppo, es Dev Cpp, Kdevelop, ecc)

Salvo diversa indicazione, nel seguito facciamo riferimento alla prima delle due tecniche.

# Compilazione

```
$ cc nomeprog.c <INVIO> /* in ambiente GNU/Linux */  
c:> gcc nomeprog.c      /* in ambiente Windows */
```

Se il programma non contiene errori, il compilatore produce il file eseguibile **a.out**.

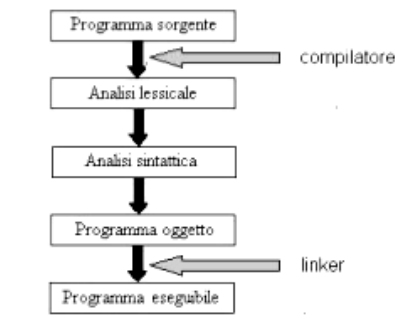
L'opzione "-o" consente di assegnare all'eseguibile un nome scelto dal programmatore.

## Esempi:

```
$ cc prog1.c          (compila prog1.c e crea l'eseguibile a.out)  
$ cc prog2.c -o prog2 (compila prog2.c e crea l'eseguibile prog2)
```

# Attività di sviluppo software

L'attività di sviluppo software può essere schematizzata come indicato



# Libreria **math.h**

Esistono **costanti predefinite** nelle varie librerie del C. Ad esempio, dovendo usare il valore di  $\pi$ , si deve scrivere in testa al programma **#include <math.h>**. In questo modo è disponibile il valore di  $\pi$  con il nome simbolico **M\_PI**

```
#include <stdio.h>
#include <math.h>
int main()
{ float r;
  printf("Immettere valore del raggio: ");
  scanf("%f", &r);
  printf("L'area e' %5.2f", M_PI*r*r);
  getchar();
  return 0;
}
```

Il programma legge la misura del cerchio in una variabile **float** e calcola e stampa il valore dell'area anch'essa **float**

M. Malatesta 2-La programmazione in C-02

9  
09/03/2014

# Libreria **math.h**

A titolo di esempio riportiamo le più comuni funzioni presenti in **math.h**

<i>Funzione</i>	<i>Effetto</i>
<b>pow(b, e)</b>	potenza $b^e$
<b>sqrt(x)</b>	radice quadrata di x
<b>abs(x)</b>	valore assoluto di x
<b>ceil(x)</b>	arrotonda x all'intero superiore
<b>floor(x)</b>	arrotonda x all'intero inferiore
<b>srand(t)</b>	inizializza il generatore di numeri casuali
<b>rand()</b>	estrae numero casuale

M. Malatesta 2-La programmazione in C-02

10  
09/03/2014

# Esempio

```
#include <stdio.h>
#include <math.h>
int main()
{
    int base, esponente, potenza;
    printf("Valore della base.....: ");
    scanf("%d", &base);
    printf("Valore dell'esponente: ");
    scanf("%d", &esponente);
    potenza = pow(base, esponente);
    printf("%d^%d = %d\n", base, esponente, potenza);
    getchar();
    return 0;
}
```

Il programma legge i valori di *base* e di *esponente* e calcola il valore della *potenza* tramite una funzione matematica.

# Argomenti

- Compilazione
- Attività di sviluppo software
- Libreria math.h
- Esempio

## Attività

- In un referendum si hanno i seguenti dati:
  - Numero degli iscritti a votare
  - Numero dei votanti
  - Numero dei SI'
  - Numero dei NO

Scrivere un programma che riproduca i risultati raccolti, stampando la percentuale dei votanti rispetto al totale degli iscritti e le percentuali dei SI' e dei NO rispetto al numero dei votanti.

## Altre fonti di informazione

- J. Purdum, C - ed. Jackson
- Romagnoli Ventura, C/C++ - Ed. Petrini