



Fabio Scotti (2004-2009)

**Laboratorio di programmazione
per la sicurezza**



Valentina Ciriani (2005-2009)

Laboratorio di programmazione

NOTA IMPORTANTE

La preparazione del corso è basata sul seguente materiale didattico sempre aggiornato sui siti web dei corsi:

- lucidi delle lezioni (file pdf)
- note del corso (file pdf e pagine web)
- esercizi di programmazione (pagine web)

Le informazioni contenute nelle pagine web **non** sono incluse nei presenti lucidi

Lezione 1 e 2



Fabio Scotti (2004-2009)

**Laboratorio di programmazione
per la sicurezza**



Valentina Ciriani (2005-2009)

Laboratorio di programmazione



Fabio Scotti (2004-2009)
Laboratorio di programmazione per la sicurezza



Valentina Ciriani (2005-2009)
Laboratorio di programmazione

Lezione 1 e 2

Introduzione al linguaggio C e significato della compilazione

Obiettivi :

- Conoscere le caratteristiche di un linguaggio di programmazione
- Comprendere il significato della compilazione, del codice sorgente, del codice oggetto, di un programma eseguibile

Linguaggi di programmazione

Un programma è una sequenza di istruzioni scritte in un linguaggio di programmazione.

Esistono centinaia di linguaggi di programmazione:

C; C++; Java; Fortran; Assembly; Pascal; Cobol; Ada; VB; PERL; C#; Matlab; Modula2.

Vi sono linguaggi direttamente comprensibili dall'elaboratore.

Altri necessitano di traduzioni, cioè di sequenze di operazioni che rendono il programma eseguibile dall'elaboratore.

4

Categorie di linguaggi

I linguaggi di programmazione possono essere distinti in:

- linguaggi MACCHINA;
- linguaggi ASSEMBLY;
- linguaggi di ALTO LIVELLO.

5

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Linguaggio macchina

- **Ogni processore può direttamente comprendere il suo linguaggio macchina.**
- **Possiamo intenderlo come il linguaggio naturale del processore.**
- **Dipende strettamente dal processore.**
- **E' di difficile leggibilità.**

...

1010....10101
1111....10101
1011....10101
1111....10101

...

6

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

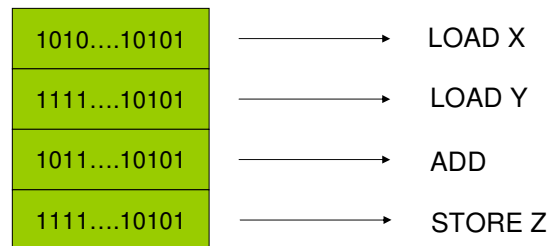
Linguaggio assembly (1)

- E' la traduzione del significato delle istruzioni del linguaggio macchina in un formato più leggibile.

- Esempio :

$z=x+y;$

...



...

7

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Linguaggio assembly (2)

- Ha semplificato molto il compito dei primi programmatori.
- Un semplice programma assembly è composto da moltissime istruzioni.
- Anche semplici operazioni fra variabili, come $z=x+y$, implicano la scrittura di più istruzioni.

8

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Linguaggio di alto livello

- **Permette di esprimere operazioni elementari con una sola istruzione (es.: funzioni matematiche).**
- **Permette di scrivere programmi facilmente comprensibili dal programmatore.**
- **E' un linguaggio molto vicino al linguaggio naturale del programmatore (inglese).**

Es: if done then print_results()

9

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Compilazione

PROGRAMMATORE

Programmi comprensibili dal programmatore
(ALTO LIVELLO)

ASSEMBLY

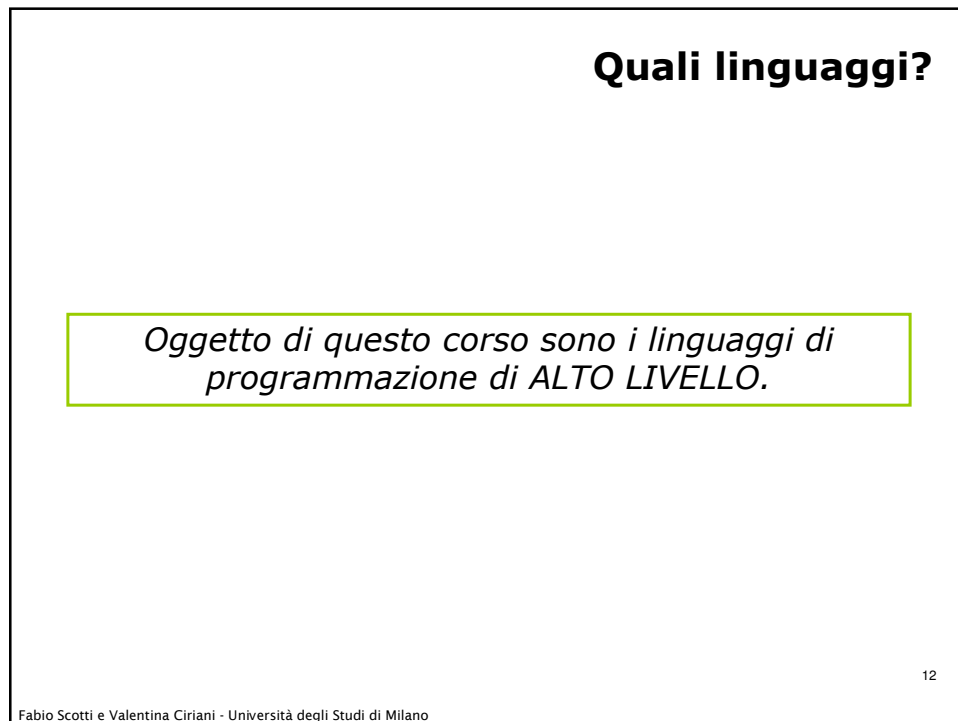
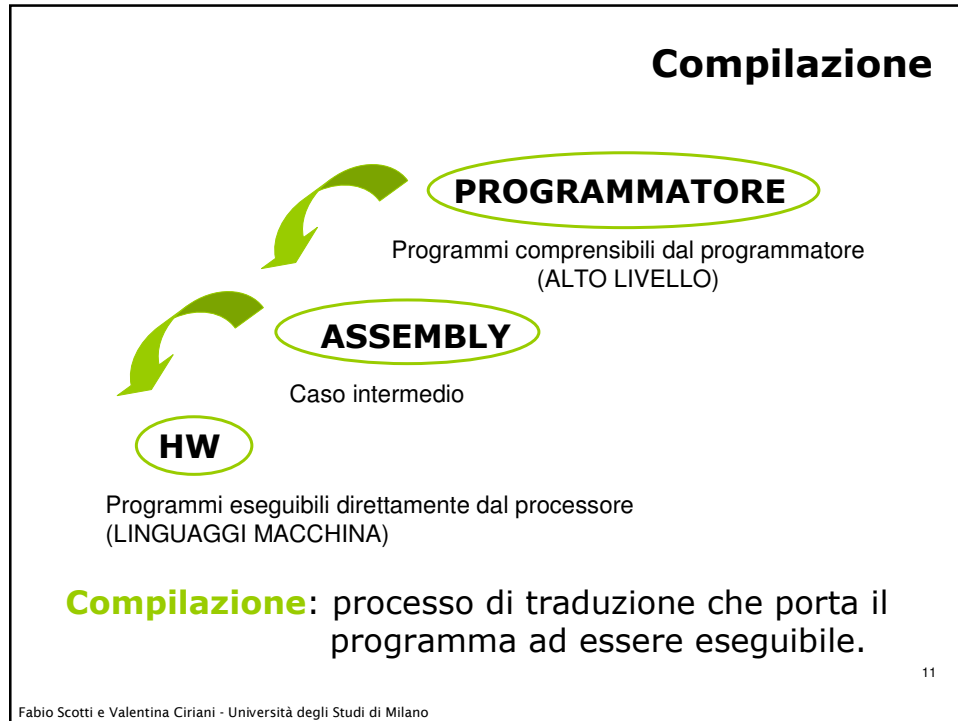
Caso intermedio

HW

Programmi eseguibili direttamente dal processore
(LINGUAGGI MACCHINA)

10

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano



Linguaggi di alto livello: esempi

```
# include <studio.h>
int main
{
    printf ("Hello World!");
    return 0;
}
```

C

```
class helloworld
{
    public static void main (String args [])
    {
        System.out.println ("Hello World!");
    }
}
```

JAVA

PROGRAM HELLOWORD

```
C   Le prime sei colonne sono vuote
    WRITE (UNIT=*,FMT=*) 'Hello World!'
    END
```

FORTRAN

File di testo
contenente
istruzioni,
commenti,direttive.

Ulteriori esempi in: www2.latech.edu/~acm/HelloWorld.shtml

13

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Quale il linguaggio migliore?

- Ogni linguaggio possiede le proprie peculiarità.
- Il programmatore esperto le conosce e sceglie il linguaggio più adatto per portare a termine, in modo efficace, il proprio compito.
- **NON** bisogna **AFFEZIONARSI** ad un linguaggio di programmazione.

**Il programmatore esperto deve essere
VERSATILE!!**

14

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Linguaggio C

- **Nasce da Dennis Ritchie nei laboratori BELL.**
- **E' stato implementato per la prima volta su un calcolatore DEC PDP-11 nel 1972.**
- **Il sistema operativo UNIX è stato scritto in C!**
- **E' indipendente dall'hardware ⇒ PORTABILE.**
- **E' stato standardizzato nel 1989 con il nome di ANSI/ISO9899:1980.**

15

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Perché il linguaggio C in questo corso?

- **Linguaggio di ALTO LIVELLO.**
- **Permette di studiare aspetti vicini all'HW.**

Essere in grado di astrarre dagli esempi in linguaggio C i concetti generali della programmazione.

16

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Rendere eseguibile un programma

Per rendere eseguibile un programma C su un calcolatore servono 6 passi:

1. editazione;
2. pre-elaborazione;
3. compilazione;
4. collegamento;
5. caricamento;
6. esecuzione.

17

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Editazione

```

#include <stdio.h>
int main()
{
    int x;
    int i;

    x = 1;
    i=x+3;

    printf("Il risultato di i=x+3 e' %d" , i);
    getchar(); // trucco per far rimanere aperta la finestra di testo
}

```

18

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Pre-elaborazione

```

#include <stdio.h>
#include "UnaMiaLIBRERIA.h"
#define LUN 10

int main()
{
    int x;
    int i;

    x = 1;
    i=x+3;
}

```

19

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Compilazione (1)

Partendo da questo esempio di programma C :

```

#include <stdio.h>
int main()
{
    int x;
    int i;

    x = 1;
    i=x+3;

    printf("Il risultato di i=x+3 e' %d" , i);
    getchar(); // trucco per far rimanere aperta la finestra di testo
}

```

20

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Compilazione (2)

L'istruzione `i = x + 3;` diventa:

```
mov eax, [x]
add eax, 3
mov [i], eax
```

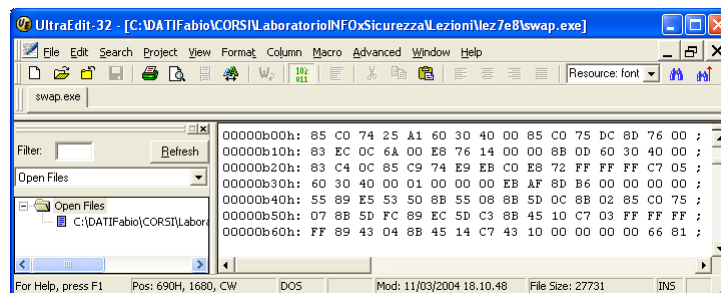
- `eax` è un registro del processore;
- `[x]` è il contenuto della cella di memoria di indirizzo `x`;
- (Microsoft assembler).

21

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Compilazione (3)

```
mov eax, [x]
add eax, 3
mov [i], eax
```



Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Collegamento (linking)(1)

Viene incluso il codice delle funzioni presenti nelle librerie:

```
#include <stdio.h>
int main()
{
    int x;
    int i;

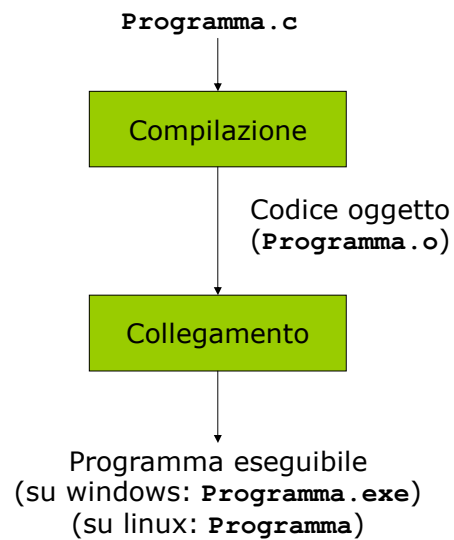
    x = 1;
    i=x+3;

    printf("Il risultato di i=x+3 e' %d" , i);
    getchar(); // trucco per far rimanere aperta la finestra di testo
}
```

23

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

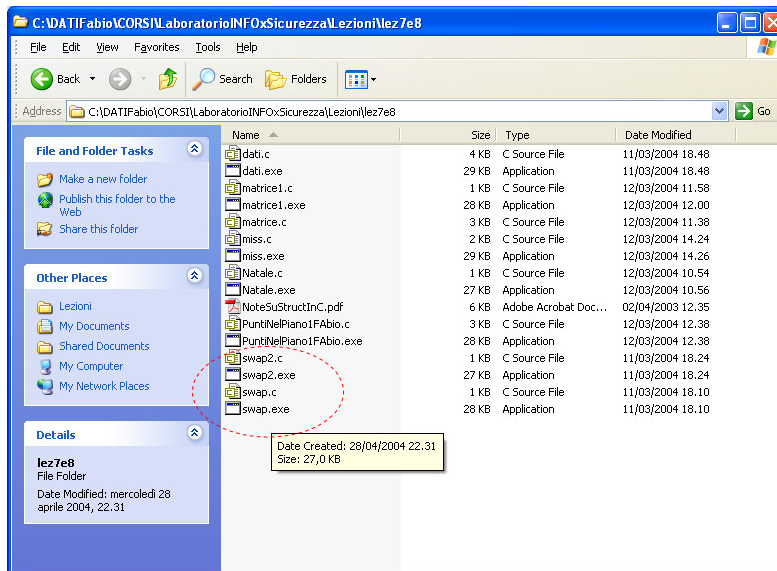
Collegamento (linking) (2)



24

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Caricamento ed esecuzione



25

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano



Fabio Scotti (2004-2009)

Laboratorio di programmazione per la sicurezza



Valentina Ciriani (2005-2009)

Laboratorio di programmazione

Lezione 1 e 2

Struttura di un programma C

Obiettivi :

- Essere in grado di identificare e modificare la struttura di programma in linguaggio C, attraverso la conoscenza delle componenti del programma

Esempio di programma C

```
#include <stdio.h>

int main()
{
    char sex;
    int eta;

    printf("Inserire il proprio sesso:\n M sta per maschio:\n F sta per femmina:\n");
    scanf("%c", &sex);

    printf("Inserire l'eta\n");
    scanf("%d", &eta);

    if((sex=='M') || (sex=='m')) printf("Sei un maschio e hai %d anni", eta);
    else printf("Sei una femmina e hai %d anni",eta);

    printf("\n\nPremi un tasto per uscire");
    fflush(stdin);
    getchar(); // trucco per far rimanere aperta la finestra di testo
}
```

27

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Struttura di un programma C

1. DIRETTIVE PER IL PRE-COMPILATORE

2. Funzione MAIN:

1. parte DICHIARATIVA;
2. dichiarazione delle COSTANTI;
3. dichiarazione delle VARIABILI;
4. parte ESECUTIVA.

28

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Direttive per il pre-compilatore

- Sono indirizzate al pre-compilatore e devono essere eseguite prima della compilazione.
- Iniziano con il carattere speciale "#" e sono moltissime.
- Quelle che ci interessano sono principalmente due:
 - includere le librerie
(es: "#include <stdio.h>");
 - definire delle etichette
(es: "#define DIMENSIONEMATRICE 5").
- Vanno inserite in testa al programma (per motivi di buona programmazione)

29

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Posizione direttive pre-compilatore

```
#include <stdio.h>
int main()
{
    char sex;
    int eta;

    printf("Inserire il proprio sesso:\n M sta per maschio:\n F sta per femmina:\n");
    scanf("%c", &sex);

    printf("Inserire l'eta\n");
    scanf("%d", &eta);

    if((sex=='M') || (sex=='m')) printf("Sei un maschio e hai %d anni", eta);
    else printf("Sei una femmina e hai %d anni",eta);

    printf("\n\nPremi un tasto per uscire");
    fflush(stdin);
    getchar(); // trucco per far rimanere aperta la finestra di testo
}
```

30

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Funzione main()

- **Contiene il programma vero e proprio che deve essere eseguito ed è divisa in più parti.**

```
#include <stdio.h>

int main()
{
    char sex;
    int eta;

    printf("Inserire il proprio sesso:\n M sta per maschio:\n F sta per femmina:\n");
    scanf("%c", &sex);

    printf("Inserire l'eta\n");
    scanf("%d", &eta);

    if((sex=='M') || (sex=='m')) printf("Sei un maschio e hai %d anni", eta);
    else printf("Sei una femmina e hai %d anni",eta);

    printf("\n\nPremi un tasto per uscire");
    fflush(stdin);
    getchar(); // trucco per far rimanere aperta la finestra di testo
}
```

31

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Parte dichiarativa

- **Qui si elencano tutte le variabili e le costanti che si intendono utilizzare.**
- **Lo scopo principale della parte dichiarativa è effettuare una **diagnostica** per trovare errori nell'uso delle variabili.**
- **Questa parte si divide ulteriormente in due parti:**
 - dichiarazione delle Variabili
 - dichiarazione delle Costanti

32

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Posizione della parte dichiarativa

```
#include <stdio.h>

int main()
{
    char sex;
    int eta;

    printf("Inserire il proprio sesso:\n M sta per maschio:\n F sta per femmina:\n");
    scanf("%c", &sex);

    printf("Inserire l'eta\n");
    scanf("%d", &eta);

    if((sex=='M') || (sex=='m')) printf("Sei un maschio e hai %d anni", eta);
    else printf("Sei una femmina e hai %d anni",eta);

    printf("\n\nPremi un tasto per uscire");
    fflush(stdin);
    getchar(); // trucco per far rimanere aperta la finestra di testo
}
```

33

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Sintassi della dichiarazione di variabili

- Si deve specificare il **nome** e il **tipo** di tutte le variabili che si intendono utilizzare all'interno del programma.
- La sintassi è la seguente:
 - tipo della variabile
 - nome della variabile
 - infine un ";"

Esempio:

```
int indice;
char lettera;
```

34

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Sintassi della dichiarazione di costanti

- **La sintassi è la seguente:**

- parola chiave "const"
- tipo della costante
- nome della costante
- "="
- valore della costante
- infine un ";"

Esempio:

```
int indice;
char lettera;
const long PI = 3.14;
```

35

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Parte esecutiva

```
#include <stdio.h>

int main()
{
    char sex;
    int eta;

    printf("Inserire il proprio sesso:\n M sta per maschio:\n F sta per femmina:\n");
    scanf("%c", &sex);

    printf("Inserire l'eta\n");
    scanf("%d", &eta);

    if((sex=='M') || (sex=='m')) printf("Sei un maschio e hai %d anni", eta);
    else printf("Sei una femmina e hai %d anni",eta);

    printf("\n\nPremi un tasto per uscire");
    fflush(stdin);
    getchar(); // trucco per far rimanere aperta la finestra di testo
}
```

- **In essa sono contenute le istruzioni.**
- **Alcune di queste sono funzioni predefinite, altre appartengono a librerie standard, altre ancora sono definite dal programmatore.**

36

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Distinzione tra etichette e costanti

```
#include <stdio.h>
#define ETA_PENSIONE 61

int main()
{
    char sex;
    int eta;
    const int MAGGIORE_ETA = 18;

    printf("Inserire il proprio sesso:\n M sta per maschio:\n F sta per femmina:\n");
    scanf("%c", &sex);

    printf("Inserire l'eta\n");
    scanf("%d", &eta);

    if((sex=='M' || (sex=='m')) printf("Sei un maschio e hai %d anni", eta);
    else printf("Sei una femmina e hai %d anni",eta);

    if (eta >= MAGGIORE_ETA) printf(" e sei maggiorenne");

    printf("\n\nl'eta' della pensione e' %d anni", ETA_PENSIONE);

    printf("\n\nPremi un tasto per uscire");
    fflush(stdin);
    getchar(); // trucco per far rimanere aperta la finestra di testo
}
```

istruzione per il precompilatore:
crea una etichetta (ETA_PENSIONE) associata al valore 61

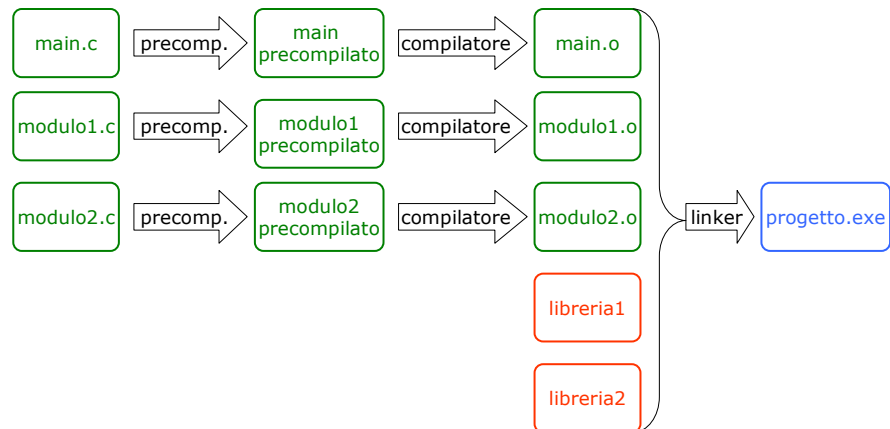
dichiarazione di costante:
viene allocata una variabile intera (MAGGIORE_ETA)
non più modificabile inizializzata a 18

etichetta:
il precompilatore sostituisce
l'etichetta ETA_PENSIONE con
61

37

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Cosa succede durante la compilazione?



38

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano



Fabio Scotti (2004-2009)

Laboratorio di programmazione per la sicurezza



Valentina Ciriani (2005-2009)

Laboratorio di programmazione

Lezione 1 e 2

Cosa serve per programmare; ambienti di sviluppo

Obiettivi :

- Essere in grado di compilare programmi negli ambienti di sviluppo: DEVC++, gcc, .Net
- Riuscire a trovare e comprendere la documentazione delle funzioni nelle librerie



Fabio Scotti (2004-2009)

Laboratorio di programmazione per la sicurezza



Valentina Ciriani (2005-2009)

Laboratorio di programmazione

Lezione 1 e 2

DEV++

Piattaforma DEVC++

- **E' un ambiente semplice e diffuso.**
- **E' utile per muovere i primi passi nella programmazione.**
- **E' un software freeware.**
- **E' scaricabile da:**
<http://www.bloodshed.net/dev/devcpp.html>
(<http://www.bloodshed.net/dev/devcpp.html>)

41

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano



Fabio Scotti (2004-2009)

Laboratorio di programmazione per la sicurezza



Valentina Ciriani (2005-2009)

Laboratorio di programmazione

Lezione 1 e 2

gcc

Compilatore gcc

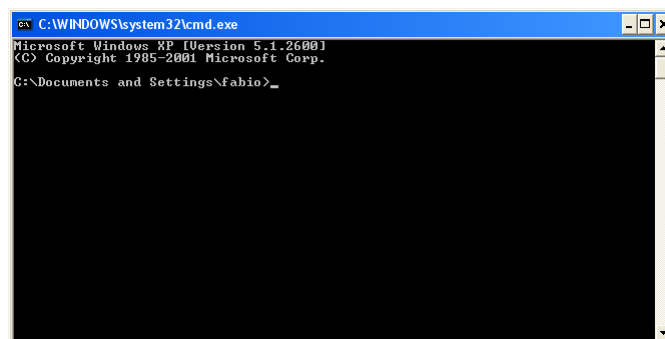
- **E' un compilatore freeware.**
- **E' compreso nel pacchetto DEVCC++.**
- **E' scaricabile al seguente URL :**
<http://gcc.gnu.org> .
- **E' multiplatforma (DOS, WINDOWS, LINUX, ALPHA, POWERPC, SUN, OS/2).**

43

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Caratteristiche del gcc

- **Non è un ambiente grafico: si usa da riga di comando.**



44

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano



Fabio Scotti (2004-2009)
Laboratorio di programmazione per la sicurezza



Valentina Ciriani (2005-2009)
Laboratorio di programmazione

Lezione 1 e 2

.Net

Piattaforma .Net

- **Prodotta dalla Microsoft (non freeware).**
- **Piuttosto diffusa.**
- **Impiegata per progetti di medie-grandi dimensioni.**
- **Integrata con notevoli librerie.**
- **Può essere impiegata per scopi didattici in questo corso.**

46

Progetto: significato in .Net

- **Non si può compilare un file .c in un passaggio, occorre CREARE UN PROGETTO.**
- **Un progetto è un insieme di FILE ed una struttura di DIRECTORY per la compilazione e l'editazione.**
- **Ciò deriva dal fatto che di solito programmi composti da migliaia di righe vengono divisi in MODULI SEPARATI.**
- **La piattaforma .Net può creare eseguibili per diverse applicazioni.**

47

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Vista della piattaforma .Net

```

seweta.c
#include <stdio.h>

int main()
{
    char sex;
    int eta;

    printf("Inserire il proprio sesso:\n M sta per maschio:\n F sta per femmina:\n");
    scanf("%c", &sex);

    printf("Inserire l'eta:\n");
    scanf("%d", &eta);

    if((sex=='M') || (sex=='m')) printf("Sei un maschio e hai %d anni", eta);
    else printf("Sei una femmina e hai %d anni", eta);

    printf("\n\nPremi un tasto per uscire");
    fflush(stdin);
    getch(); // trucco per far rimanere aperta la finestra di testo
}

```

Output

```

seweta.exe: Loaded 'C:\Documents and Settings\fabio\My Documents\Visual Studio Project\seweta\Debug\seweta...
seweta.exe: Loaded 'C:\WINNT\system32\NTDLL.DLL', Cannot find or open a required DBG file.
seweta.exe: Loaded 'C:\WINNT\system32\USER32.DLL', Cannot find or open a required DBG file.
The program '[1192] seweta.exe: Native' has exited with code 0 (0x0).

```

48

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Esempio

```
# include <stdio.h>
int main() {

printf("Hello World!");
getchar();
}
```

Salvare nel file **helloworld.c**

49

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano

Compilazione sotto . Net

- Selezionare dal menù file **nuovo progetto**.
- Scegliere un progetto **“Visual C++ per WIN32”** anche se programmeremo solo in **ANSI C**.
- Selezionare nel menù che appare **“Applicazione vuota per CONSOLE”**.
- Aggiungere il file **helloworld.c** al nostro progetto (dal menù file) e compilare.

50

Fabio Scotti e Valentina Ciriani - Università degli Studi di Milano