



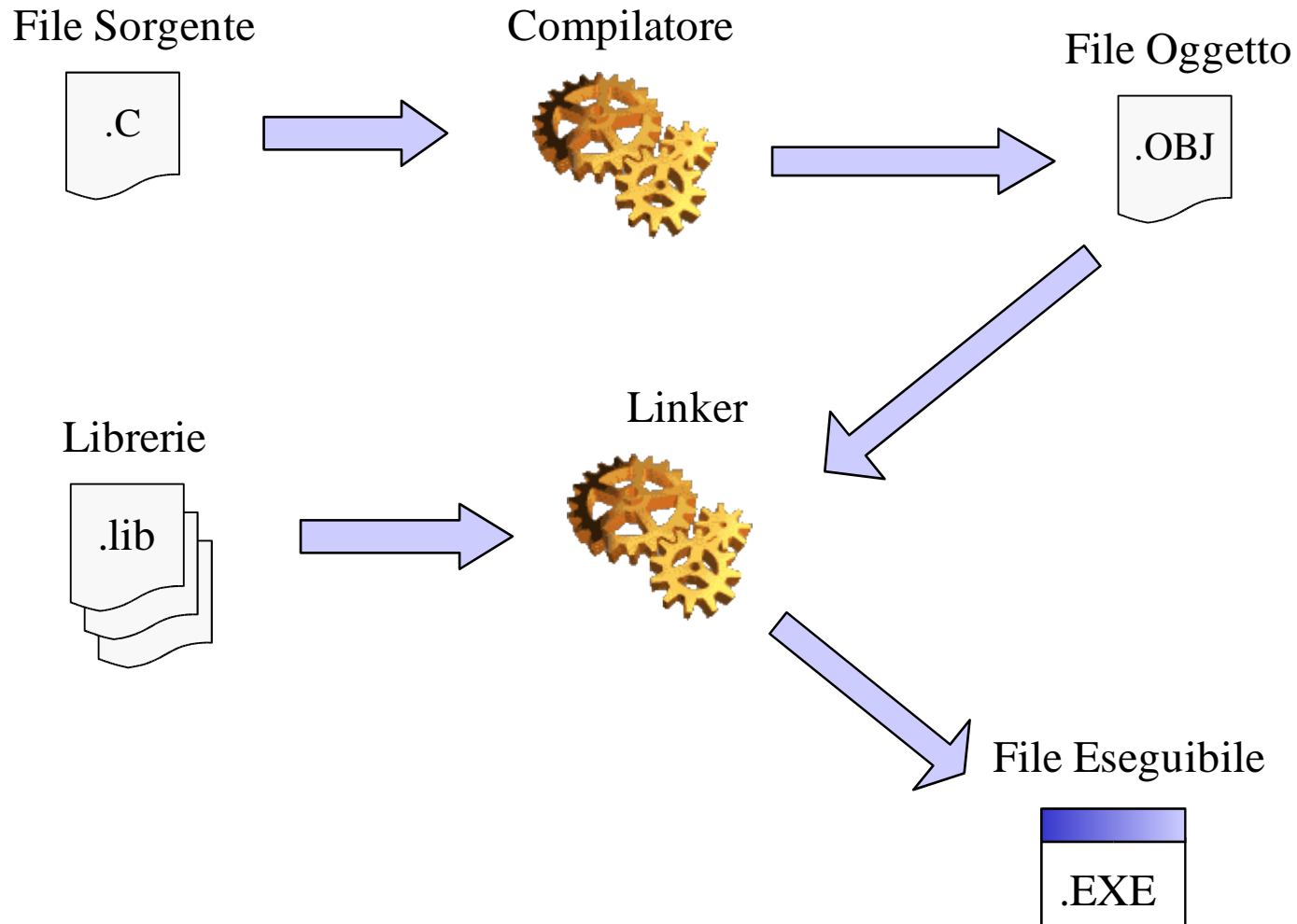
# Informatica “B”

## *Breve introduzione all’ambiente di programmazione*

Politecnico di Milano  
IV Facoltà di Ingegneria  
2003

# Dal sorgente all'eseguibile

---



# Gli strumenti

---

- Per generare il sorgente serve un *editor di testo*
- Per produrre l'oggetto serve un *compilatore*
- Per produrre l'eseguibile serve un *linker*

**Disponiamo di un ambiente integrato**

**LCC (lcc.exe)**

*disponibile gratuitamente a*

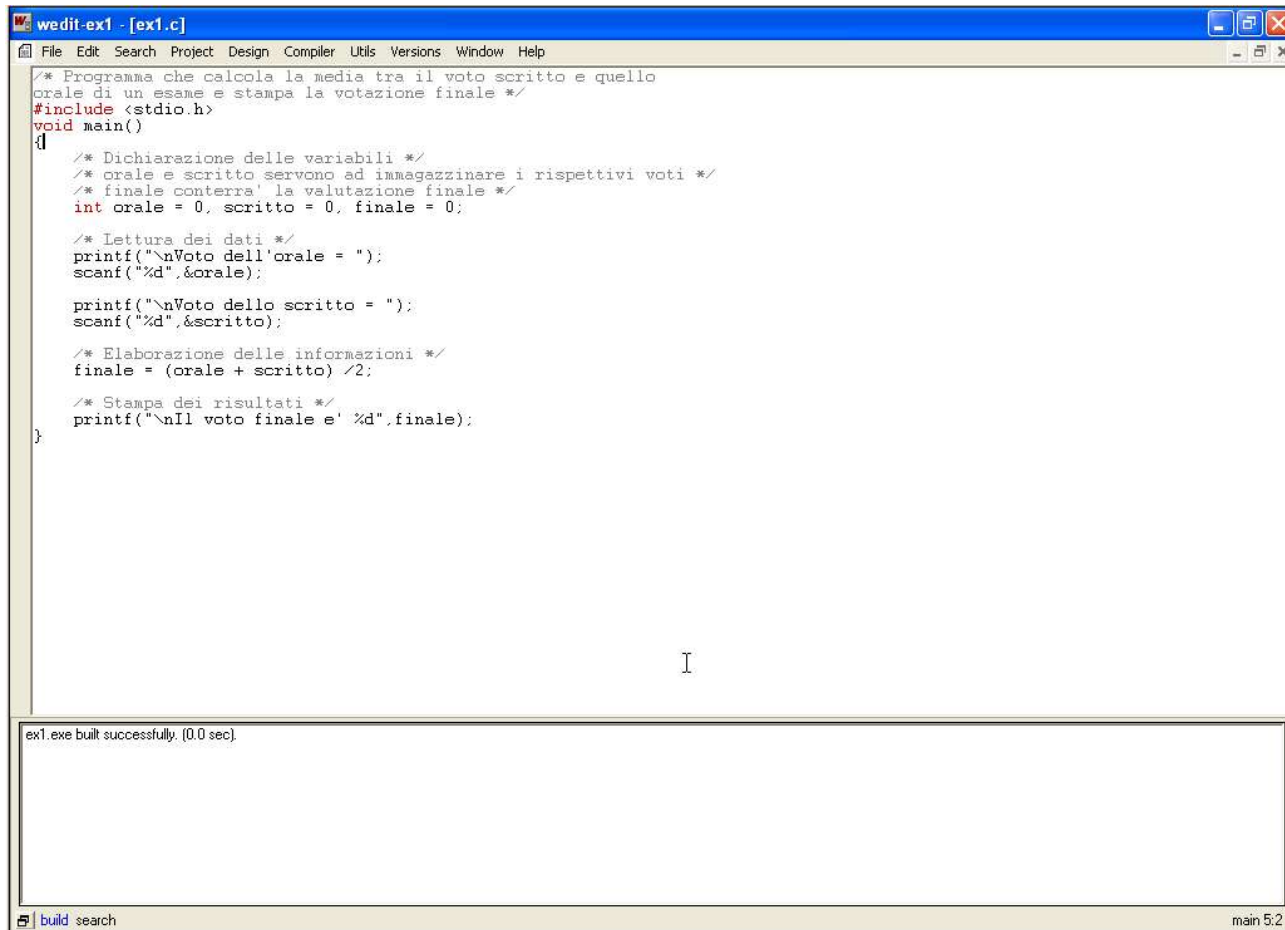
<http://www.cs.virginia.edu/~lcc-win32/>

# L'ambiente integrato

---

- Cosa permette di fare:
  - Scrivere un programma (ricordate di salvarlo!!)
  - Compilare il sorgente
  - Linkare le librerie
  - Eseguire il programma in un ambiente di 'debugging' simbolico

# L'ambiente integrato



The screenshot shows the wedit IDE window titled "wedit-ex1 - [ex1.c]". The menu bar includes File, Edit, Search, Project, Design, Compiler, Utils, Versions, Window, and Help. The main editor area contains the following C code:

```
/* Programma che calcola la media tra il voto scritto e quello
orale di un esame e stampa la votazione finale */
#include <stdio.h>
void main()
{
    /* Dichiarazione delle variabili */
    /* orale e scritto servono ad immagazzinare i rispettivi voti */
    /* finale conterra' la valutazione finale */
    int orale = 0, scritto = 0, finale = 0;

    /* Lettura dei dati */
    printf("\nVoto dell'orale = ");
    scanf("%d",&orale);

    printf("\nVoto dello scritto = ");
    scanf("%d",&scritto);

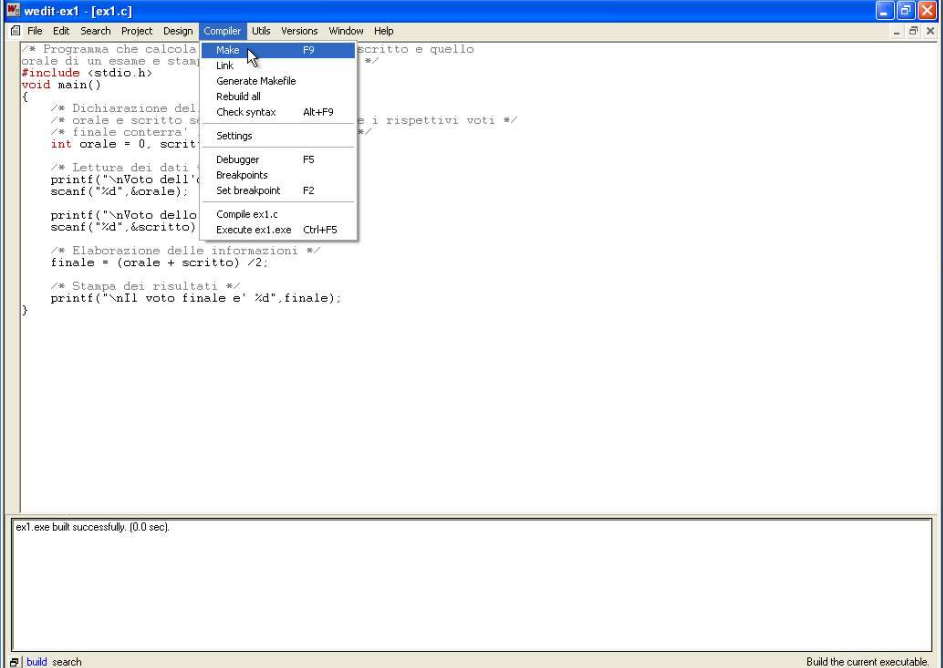
    /* Elaborazione delle informazioni */
    finale = (orale + scritto) /2;

    /* Stampa dei risultati */
    printf("\nIl voto finale e' %d",finale);
}
```

Below the code editor, a status bar indicates "ex1.exe built successfully. (0.0 sec)". At the bottom left, there is a "build search" button, and at the bottom right, it says "main 5:2".

# Compilazione

- Per compilare si attiva la voce “**Make**” del menu “**Compiler**” oppure premendo **F9**
  - Correggere eventuali errori
  - Controllare i ‘warning’



```
/* Programma che calcola
orale di un esame e stampa
il risultato scritto e quello
scritto e quello */
#include <stdio.h>
void main()
{
    /* Dichiarazione del
    /* orale e scritto e
    /* finale conterra'
    /* finale conterra'
    int orale = 0, scritto;

    /* Lettura dei dati
    printf("\nVoto dell'
    scanf("%d",&orale);

    printf("\nVoto dello
    scanf("%d",&scritto)

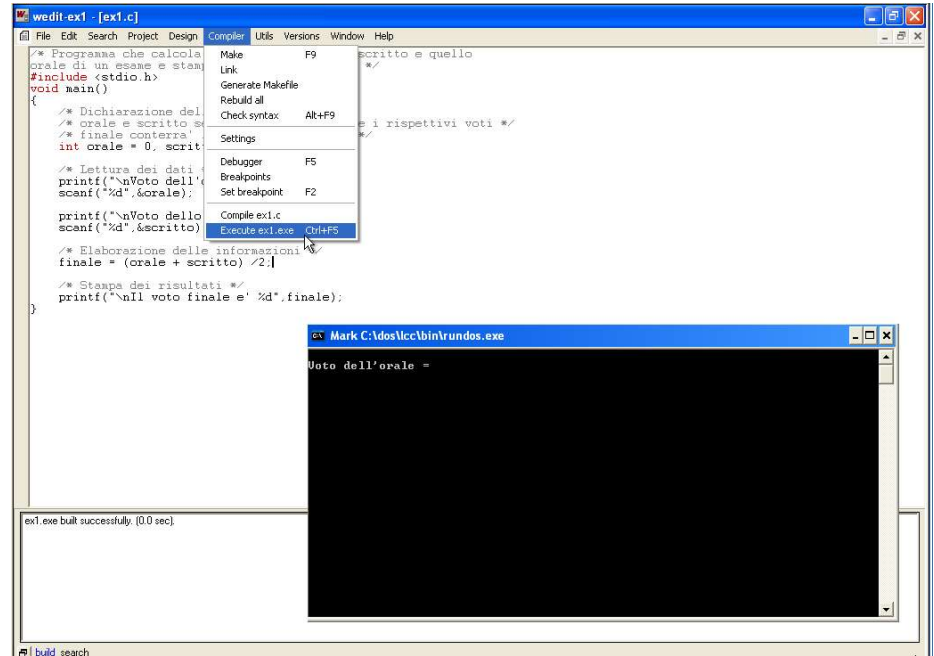
    /* Elaborazione delle informazioni */
    finale = (orale + scritto) / 2;

    /* Stampa dei risultati */
    printf("\nIl voto finale e' %d",finale);
}
```

ex1.exe built successfully. (0.0 sec).

# Esecuzione del programma

- Per eseguire il programma si seleziona la voce “Execute” dal menu “Compiler” oppure con la combinazione di tasti **ctrl-F5**



# Debugging (1)

---

- Quando un programma è molto complesso si commettono sicuramente errori.  
Il debugging è la ricerca e correzione di questi errori
- Il debugging serve a:
  - trovare errori nella codifica dell'algoritmo



# Debugging (2)

---

- Durante il debugging è possibile:
  - seguire il flusso di elaborazione eseguendo una operazione alla volta
  - ispezionare il contenuto di una variabile
  - modificare temporaneamente il valore di alcune variabili
- Il debugger si utilizza tramite i menu “Run” e “Debug”

# Debugging (3)

The screenshot shows a debugger window titled "wedit-ex1 - [ex1.c]". The code being debugged is a C program that calculates the average of two exam scores. A breakpoint is set at the line `scanf("%d",&scritto);`, which is highlighted in yellow. The program has stopped at this point. In the bottom-left corner, the current variable values are displayed: `orale = 5`, `scritto = 0`, and `finale = 0`. An inset window titled "z:\data\didattica\info03-04\laboratorio\20031009\lcclex1.exe" shows the output of the program: "Voto dell'orale = 5". The status bar at the bottom indicates the program is "Stopped" and the current function is "main 15:5".

```
/* Programma che calcola la media tra il voto scritto e quello
orale di un esame e stampa la votazione finale */
#include <stdio.h>
void main()
{
    /* Dichiarazione delle variabili */
    /* orale e scritto servono ad immagazzinare i rispettivi voti */
    /* finale conterra' la valutazione finale */
    int orale = 0, scritto = 0, finale = 0;

    /* Lettura dei dati */
    printf("\nVoto dell'orale = ");
    scanf("%d",&orale);

    printf("\nVoto dello scritto = ");
    scanf("%d",&scritto);

    /* Elaborazione delle informazioni */
    finale = (orale + scritto) /2;

    /* Stampa dei risultati */
    printf("\nIl voto finale e' %d",finale);
}
```

orale = 5  
scritto = 0  
finale = 0

Voto dell'orale = 5

auto locals stack events search Stopped main 15:5

# Qualche utile combinazione di tasti

---

- Parentesi graffa aperta:  
**ALT + 123** oppure **ALT\_GR + SHIFT + [**
- Parentesi graffa chiusa:  
**ALT + 125** oppure **ALT\_GR + SHIFT + ]**
- Quando il mouse fa i capricci :

**Shift+Frecce** : selezione testo    **Alt+Lettera** : apre i menu

**Ctrl+C** : copia                      **Ctrl + F5** : finestra di output

**Ctrl+V** : incolla                      **F5** : debugger

**F9** : compila                      **Ctrl + S** : salva il file

(le abbreviazioni sono riportate nei menu accanto alle rispettive voci)

# Elementi della sintassi

---

- Le righe che iniziano con un `#` sono direttive al **Preprocessore**

```
#include <stdio.h>
#define MAXNUM 255
```

- Programma principale:

```
void main() {...}
```

- Tutte le istruzioni terminano *sempre* con un `;`
- I blocchi di codice sono racchiusi tra parentesi graffe (dopo la graffa chiusa non ci vuole il `;`)
- Le stringhe stanno fra “ ”
- Il C è case-sensitive !!

# Ricordate

---

- Il codice deve essere **Commentato !!**  
Ci sono due modi per inserire un commento:
  - Racchiudere i commenti tra `/*` e `*/`
  - Iniziare la riga con `//`
- Il codice deve essere **Indentato !!**  
(in lcc Edit->Reformat)
- Per convenzione i nomi delle funzioni si scrivono con la lettera minuscola

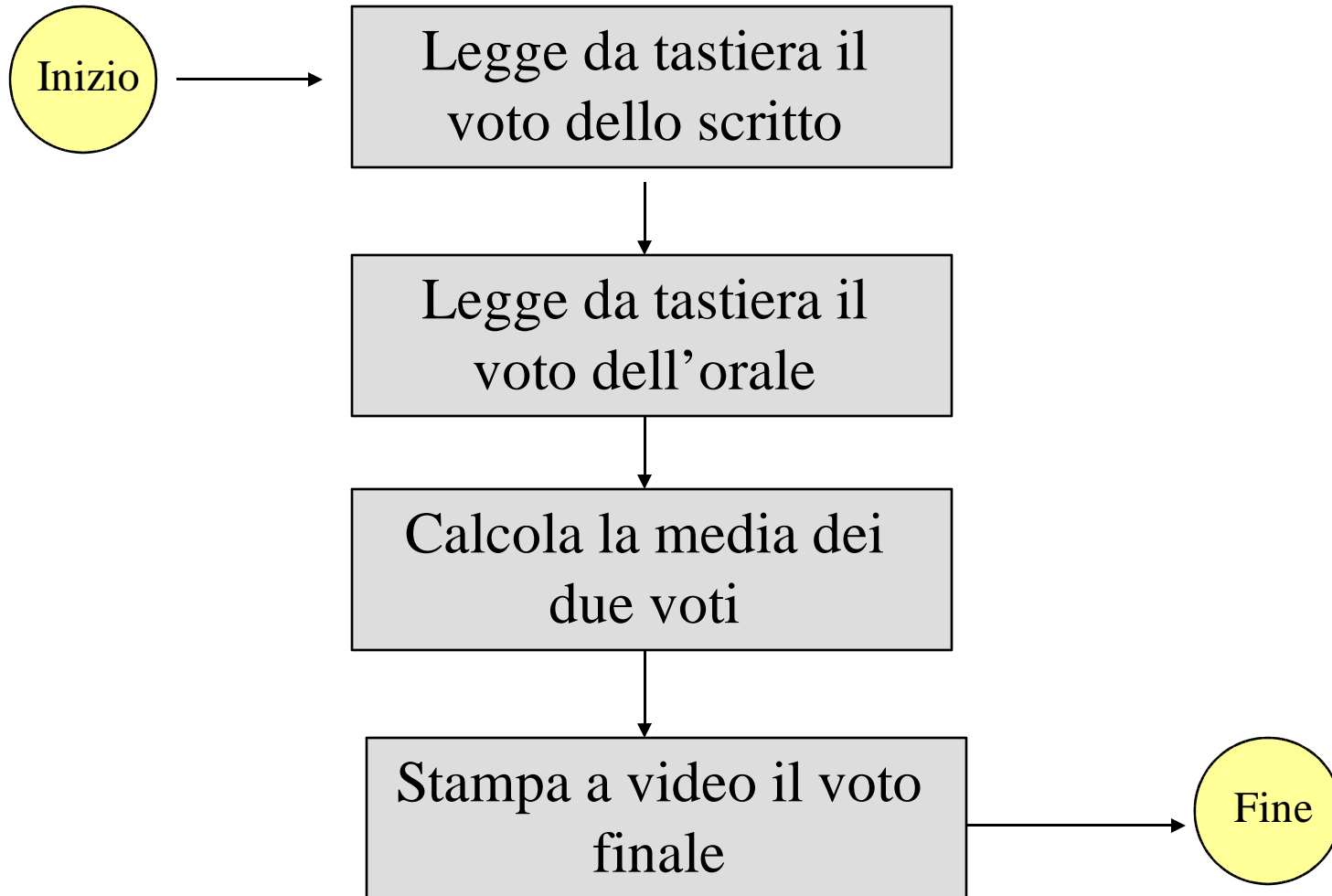
# Esercizio

---

*“Scrivere un programma che legga da tastiera due numeri interi che rappresentano il voto scritto ed orale di uno studente ad un esame e stampi a video la votazione finale ottenuta come media dei punteggi precedenti.”*

# L'algoritmo...

---



# Dichiarazione delle variabili

---

```
int scritto=0, orale=0, finale=0;
```

- *scritto* e *orale* conterranno i rispettivi i voti
- *finale* conterrà la media
- sono tutti *int* (numeri interi)



# Raccolta dell'input

---

```
printf( "\nVoto nello scritto : " );  
scanf( "%d", &scritto);  
printf( "\nVoto nell'orale : " );  
scanf( "%d", &orale);
```

- **%d** indica un intero (decimale)
- **\n** significa “vai a capo”
- **&** indica un parametro di tipo **out** (passaggio per indirizzo)

# Calcolo e stampa del risultato

---

```
finale = (scritto + orale) / 2;
```

Ora finale contiene la media dei due numeri

```
printf("\nLa votazione  
finale e`: %d\n", finale);
```

Stampa a video il risultato finale

# Soluzione completa 1/2

```
/* Programma che calcola la media tra il voto scritto e quello
   orale di un esame e stampa la votazione finale */
#include <stdio.h>
void main()
{
    /* Dichiarazione delle variabili */
    /* orale e scritto servono ad immagazzinare i rispettivi voti */
    /* finale conterra' la valutazione finale */
    int orale = 0, scritto = 0, finale = 0;

    /* Lettura dei dati */
    printf("\nVoto dell'orale = ");
    scanf("%d",&orale);
```

# Soluzione completa 2/2

```
printf("\nVoto dello scritto = ");
scanf("%d",&scritto);

/* Elaborazione delle informazioni */
finale = (orale + scritto) /2;

/* Stampa dei risultati */
printf("\nIl voto finale e' %d",finale);
}
```

# Qualche link...

---

- La pagina ufficiale del corso:  
<http://zeus.elet.polimi.it/mcinfo/>
- Il sito del compilatore  
<http://www.cs.virginia.edu/~lcc-win32/>
- Indirizzi dei responsabili di laboratorio  
*Davide Balzarotti: [davide.balzarotti@polimi.it](mailto:davide.balzarotti@polimi.it)*  
*Paolo Costa: [paolo.costa@polimi.it](mailto:paolo.costa@polimi.it)*