

# Introduzione al corso di laboratorio

Fondamenti di Informatica  
Anno Accademico 2010/2011

Antonio Miele   Ettore Speziale   Michele Tartara

Politecnico di Milano

29/30 settembre 2010

# Sommario

- 1 Organizzazione del laboratorio
- 2 Stile di lavoro
- 3 Strumenti informatici
- 4 Esempi
- 5 Bibliografia

# Indice

- 1 Organizzazione del laboratorio
- 2 Stile di lavoro
- 3 Strumenti informatici
- 4 Esempi
- 5 Bibliografia

# Obiettivi

Questo è un corso introduttivo; tratteremo argomenti base:

- 1 Comprendere ed analizzare i problemi proposti
- 2 Progettare autonomamente algoritmi per risolverli
- 3 Implementarli tramite linguaggio C
- 4 Trasformarli in eseguibili usando gli strumenti GNU
- 5 Eseguirli, correggerli e validarne i risultati

Il laboratorio permette di fare pratica sugli argomenti presentati a lezione.

# Regole

Alcune convenzioni da rispettare:

- Frequenza non obbligatoria
- Obbligatorio firmare il foglio presenze una volta entrati
- Si inizia alle 8:30 (alle 10:30 **SOLO** per la squadra 2)
- Solo in casi eccezionali e all'inizio del corso è possibile cambiare squadra di laboratorio, **previo accordo** con il responsabile
- I responsabili vi aiutano, non vi **risolvono** gli esercizi

# Didattica I

Il laboratorio assegna 5 punti che sono cumulati con il punteggio dello scritto. Acquisibili tramite:

- ① Freqwentazione laboratorio e risoluzione esercizi proposti (2 punti)
  - I responsabili controlleranno occasionalmente i programmi implementati chiedendo di commentare le soluzioni proposte
- ② Sostenimento prova finale
  - Si terrà durante l'ultima sessione di laboratorio

# Didattica II

Attenzione:

- È richiesto almeno 1 punto per passare il corso
- Con 0 punti si ripete l'intero corso l'anno prossimo
- I ripetenti possono
  - Frequentare di nuovo il corso, o
  - Sostenere solo la prova finale di laboratorio, o
  - Mantenere il voto dell'anno scorso

# Esercizi

Ad ogni lezione verranno proposti 4/5 esercizi:

- Non è indispensabile risolverli tutti in laboratorio
- Un minimo di impegno è richiesto (almeno 2 esercizi per sessione)

Risolvere i restanti esercizi è utile per prepararsi all'esame.

Le soluzioni saranno pubblicate con un lieve ritardo sul sito del corso [2].

Inoltre è disponibile un forum [4] in cui scambiarsi idee.



# Indice

- 1 Organizzazione del laboratorio
- 2 Stile di lavoro**
- 3 Strumenti informatici
- 4 Esempi
- 5 Bibliografia

# Carta e penna

Risolvere problemi implica diverse fasi. Le più importanti **non richiedono** l'utilizzo del computer:

- Capire il problema
- Definire un algoritmo per risolverlo

Le prime volte è utile ragionare su “**carta**”, e poi passare all'implementazione della soluzione su PC.

Ricordate che non esiste una sola soluzione giusta. Spesso ne esistono molte.

# Codifica

Il vostro scopo deve essere quello di produrre un programma corretto e “buono”:

- 1 Risolve il problema
- 2 Usa le strutture giuste
- 3 E' scritto “bene”  $\Rightarrow$  Facile capire cosa fa e modificarlo

Ottenere tali risultati in un solo passaggio è difficile. Operare per fasi:

- Risolvo una parte del problema per volta
- Prima di procedere verifico **sempre** 1, 2, 3

# Testing

Per verificare che il programma funzioni va eseguito e il suo output controllato. Cosa fare se i risultati ottenuti e quelli attesi non combaciano?

- L'algoritmo è corretto?
- Sono stati commessi errori durante il processo di codifica?

Se non bastasse bisogna ispezionare il comportamento del programma durante l'esecuzione tramite un debugger.

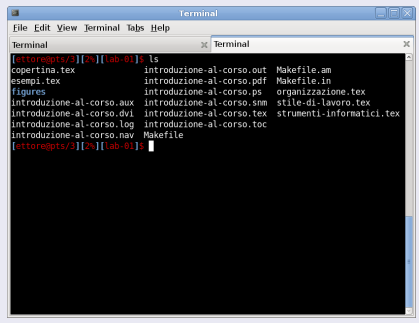
# Indice

- 1 Organizzazione del laboratorio
- 2 Stile di lavoro
- 3 Strumenti informatici**
- 4 Esempi
- 5 Bibliografia

# GNU/Linux

L'interfaccia base è il terminale.

## Screenshot



```
Terminal
File Edit View Terminal Tabs Help
Terminal
[ettore@pts/2][2%][lab-01]$ ls
copertina.tex      introduzione-al-corso.out  Makefile.am
esempi.tex         introduzione-al-corso.pdf  Makefile.in
figures            introduzione-al-corso.ps   organizzazione.tex
introduzione-al-corso.aux  introduzione-al-corso.snm  stile-di-lavoro.tex
introduzione-al-corso.dvi  introduzione-al-corso.tex  strumenti-informatici.tex
introduzione-al-corso.log  introduzione-al-corso.toc
introduzione-al-corso.nav  Makefile
[ettore@pts/2][2%][lab-01]$
```

## Comandi utili

Cartella corrente `pwd`

Lista file `ls`

Cambia cartella `cd`

Manuale `man COMMAND`

Eeguire programmi `./program`

Editor `gedit`

# Il compilatore utilizzato: GCC

Permette di tradurre i programmi da C a linguaggio macchina.  
Il tutto è mascherato dal comando `gcc input.c -o output`.

## Errore

```
c : bash
File Modifica Visualizza Scorrimento Segnalibri Impostazioni Aiuto
miky@peprice:~/programmi/c$ gcc err1.c
err1.c: In function 'main':
err1.c:4: error: expected ';' before 'return'
miky@peprice:~/programmi/c$
```

## Warning

```
c : bash
File Modifica Visualizza Scorrimento Segnalibri Impostazioni Aiuto
miky@peprice:~/programmi/c$ gcc -Wall err2.c
err2.c: In function 'main':
err2.c:4: warning: control reaches end of non-void function
miky@peprice:~/programmi/c$
```

# Note finali I

Il consiglio è di installare GNU/Linux sui PC di casa:

- Vasta disponibilità di software per sviluppatori
- Documentazione immensa
- Facile programmare come in laboratorio

Una delle distribuzioni più semplici da usare ed intuitive è Ubuntu [3].



# Note finali II

Per Windows:

- Codice conforme agli standard  $\Rightarrow$  Compila anche su Windows
- Disponibile ambiente che simula GNU/Linux  $\Rightarrow$  Cygwin
- Compilatore C nativo  $\Rightarrow$  Dev-C++ [1]

Sul sito del corso [2] sono disponibili delle tabelle riassuntive con le istruzioni C, i comandi GNU/Linux e di GDB.

# Indice

- 1 Organizzazione del laboratorio
- 2 Stile di lavoro
- 3 Strumenti informatici
- 4 Esempi**
- 5 Bibliografia

# Esempi

Due semplici esempi, per prendere confidenza con gli strumenti di sviluppo:

`helloworld` stampa un saluto

`hellonumbers` stampa alcuni numeri

# Indice

- 1 Organizzazione del laboratorio
- 2 Stile di lavoro
- 3 Strumenti informatici
- 4 Esempi
- 5 Bibliografia**

# Bibliografia



Bloodshed.

Dev-C++.

<http://www.bloodshed.net/devcpp.html>, 2009.



Cristiana Bolchini.

Fondamenti di Informatica.

<http://home.dei.polimi.it/bolchini/didattica.htm>, 2009.



Canonical.

Ubuntu Linux.

<http://www.ubuntu.com>, 2010.



Poseidon.

Forums at Poseidon.

<http://poseidon.elet.polimi.it/smf/>, 2009.