



Intro ad analisi e programmazione

Alessandra Giordani

agiordani@disi.unitn.it

Lunedì 5 marzo 2011

<http://disi.unitn.it/~agiordani/>



Analisi e programmazione

- l'insieme delle attività preliminari atte a risolvere problemi utilizzando un elaboratore
 - dalla formulazione del problema
 - fino alla predisposizione dell'elaboratore
- Scopo dell'analisi → definire un algoritmo
- Scopo della programmazione → definire un programma



Analisi

- Algoritmo → elenco finito di istruzioni, che specificano le operazioni eseguendo le quali si risolve una classe di problemi
- Un problema della classe viene risolto con l'algoritmo sui dati che lo caratterizzano
- Un algoritmo non può essere eseguito direttamente dall'elaboratore
- Vediamo alcuni esempi...

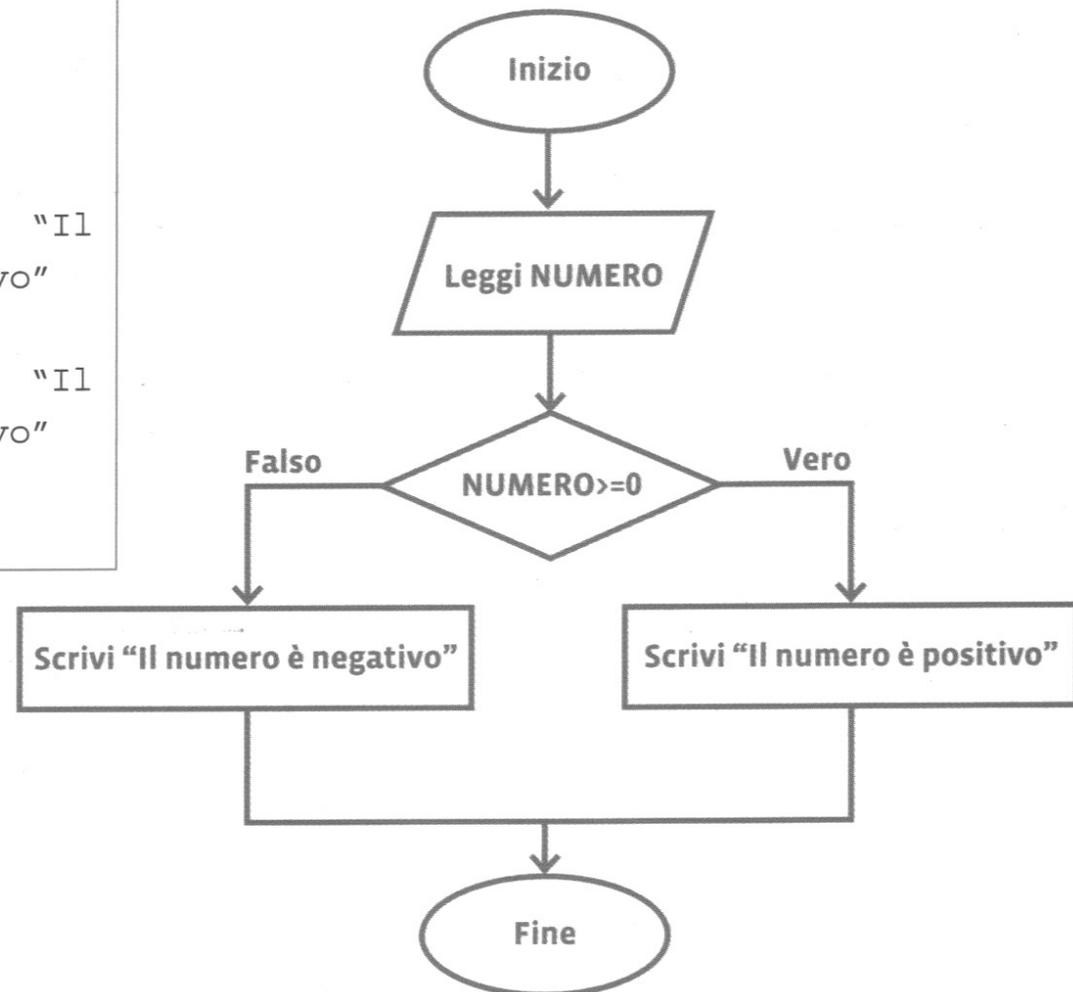


Esempi di analisi

- Analizzare e risolvere i seguenti problemi
 - Dato un numero dire se è positivo o negativo
 - Dire se un numero è positivo, negativo o nullo
 - Eseguire il prodotto tra due numeri usando solo l'operazione di somma
 - Trovare il maggiore tra tre numeri
 - Organizzare una spesa con 100€ massimo
 - Divisione tra due numeri
- Diagrammi a blocchi - di flusso
- Pseudo codice

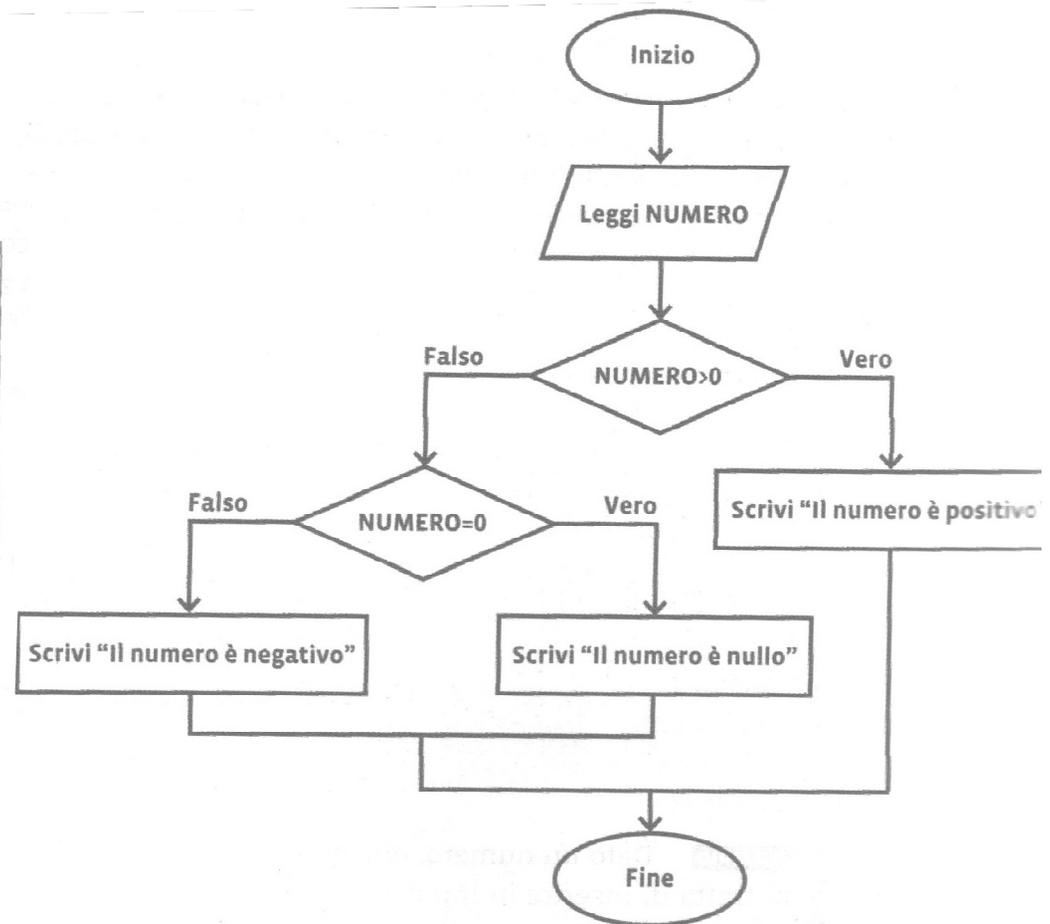
Il numero è positivo o negativo?

```
Inizio
  Leggi NUMERO
  Se NUMERO >= 0
    Allora
      Scrivi "Il
numero è positivo"
    Altrimenti
      Scrivi "Il
numero è negativo"
  FineSe
Fine
```



... e se fosse nullo?

```
Inizio
Leggi NUMERO
Se NUMERO>0
Allora
  Scrivi "Il numero è positivo"
Altrimenti
  Se NUMERO=0
  Allora
    Scrivi "Il numero è nullo"
  Altrimenti
    Scrivi "Il numero è negativo"
  FineSe
FineSe
Fine
```





Prodotto di due numeri

PROBLEMA

- Dati due numeri effettuare il loro prodotto usando solo l'operazione somma.

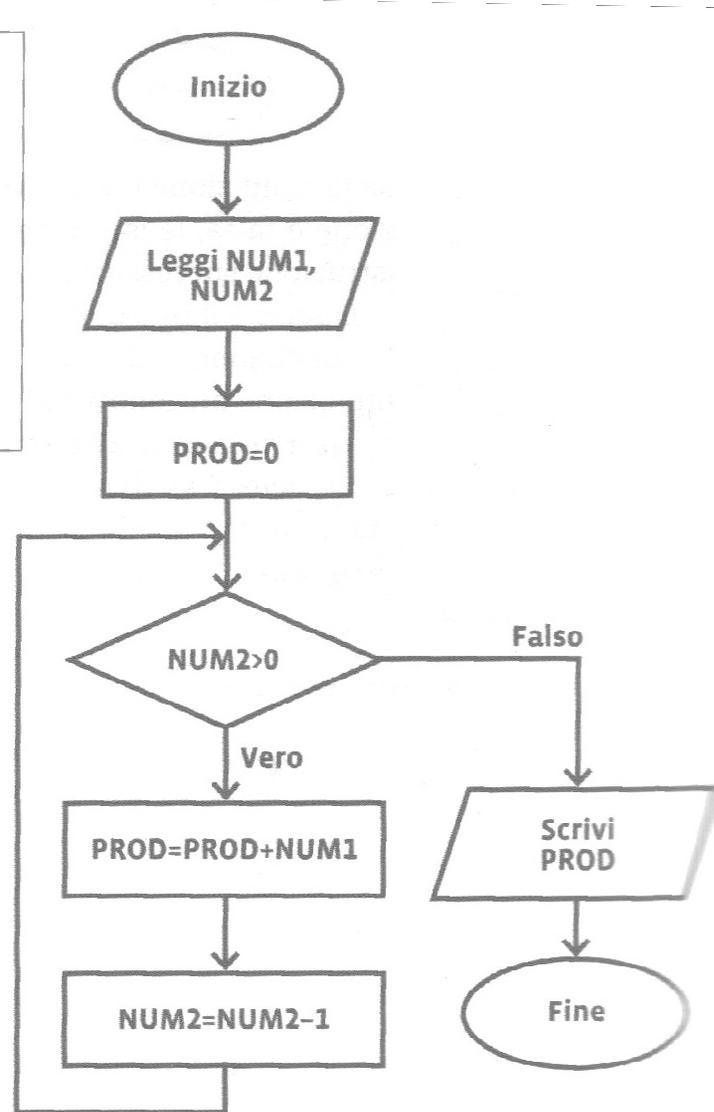
SOLUZIONE

- Provate a trovare una soluzione al problema descrivendola con diagrammi di flusso

Prodotto di due numeri

```
Inizio
  Leggi NUM1, NUM2
  PROD=0
  Mentre NUM2>0
    PROD=PROD+NUM1
    NUM2=NUM2-1
  FineMentre
  Scrivi PROD
Fine
```

- Usando solo l'operazione somma
- SOLUZIONE:





Maggiore tra 3 numeri

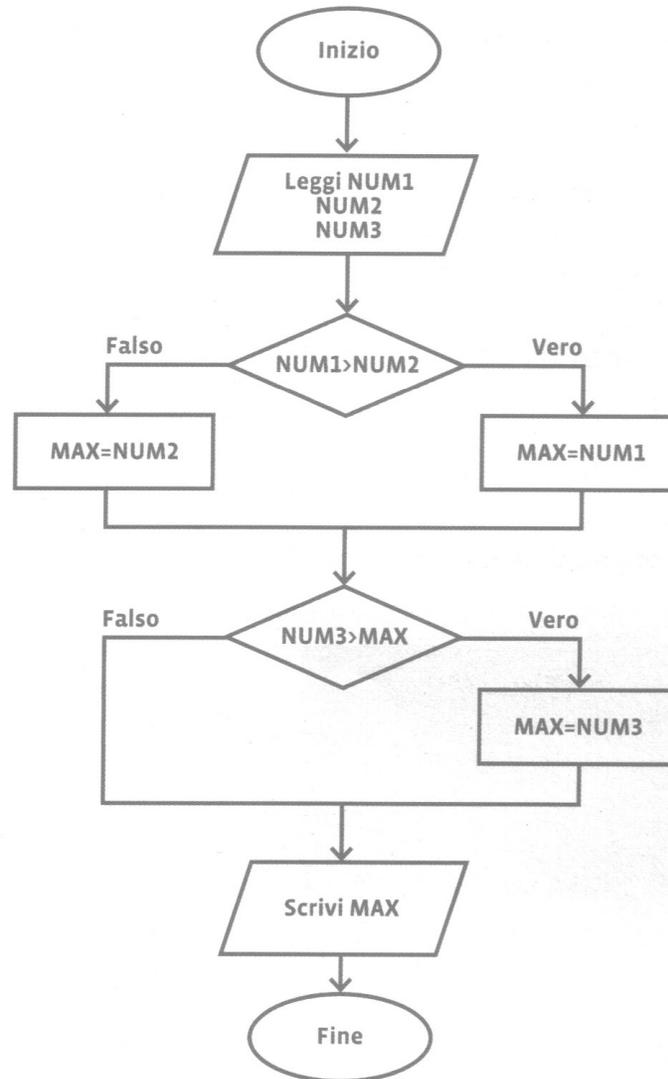
PROBLEMA

- Dati 3 numeri stabilire quale sia il valore maggiore

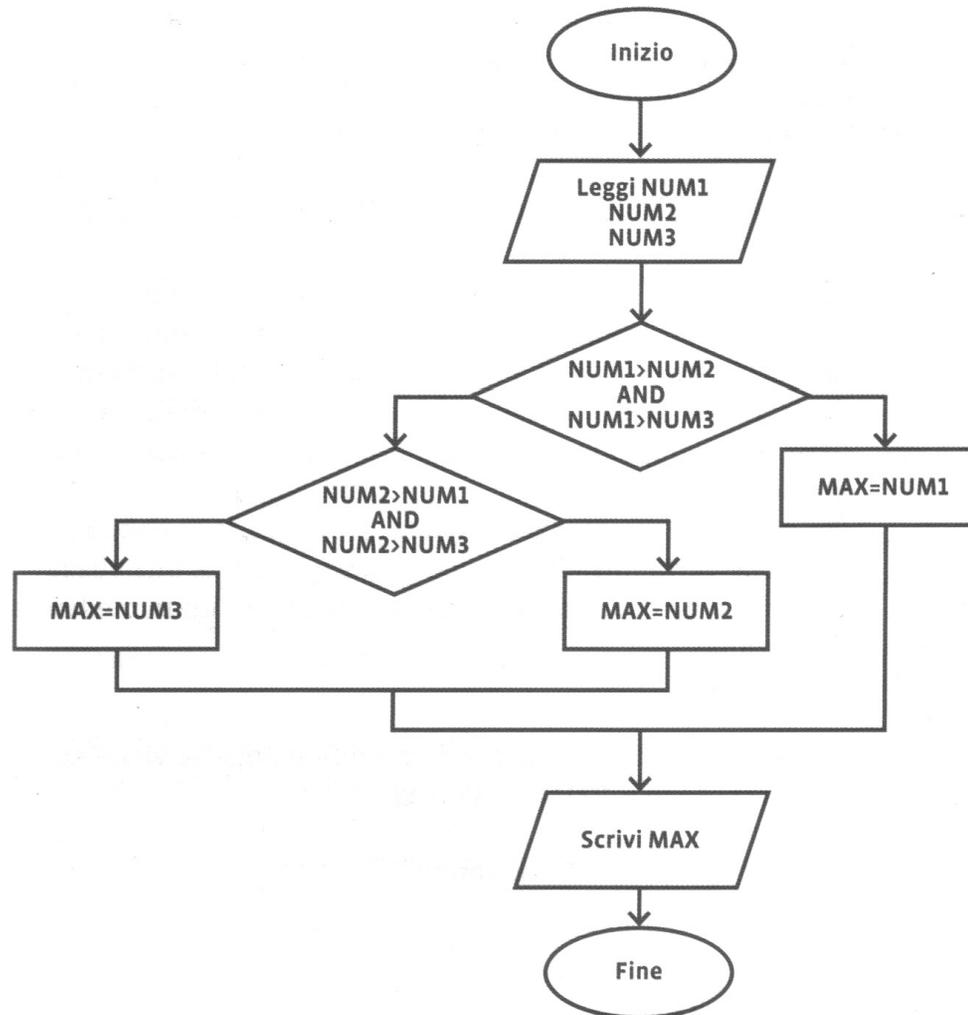
SOLUZIONE

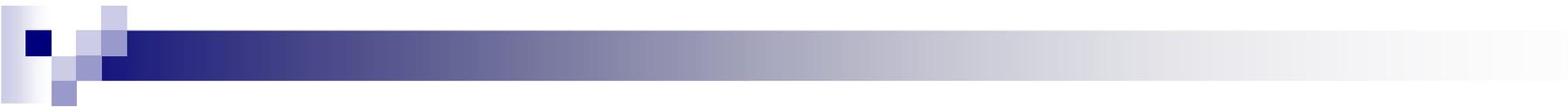
- Provate a trovare una soluzione al problema descrivendola con diagrammi di flusso

Maggiore tra 3 numeri (ver.1)



Maggiore tra 3 numeri (ver.2)





Spesa di prodotti per max 100€

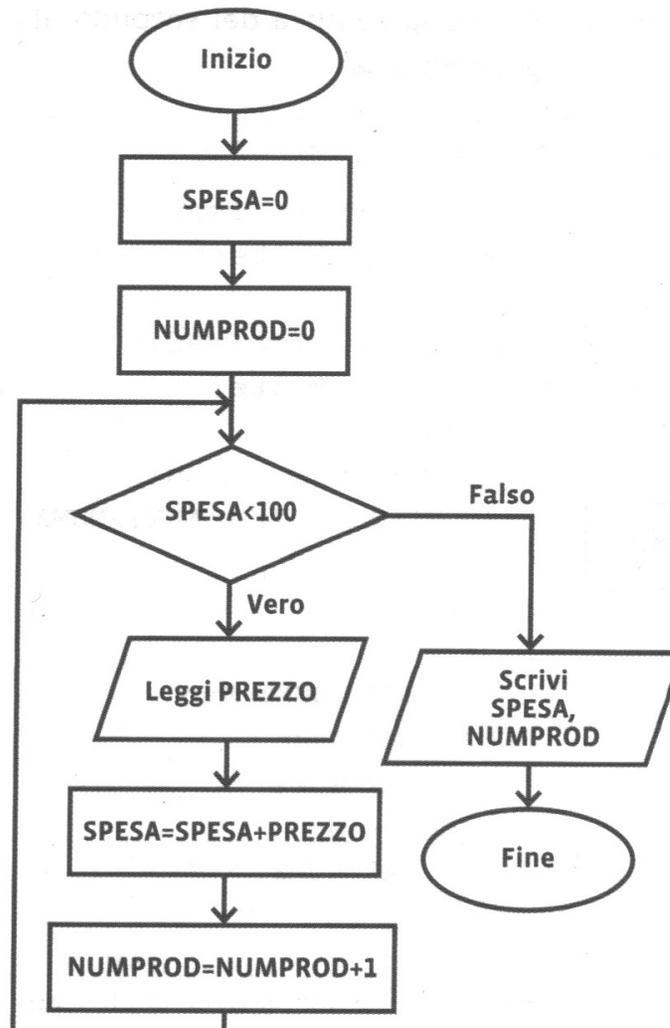
PROBLEMA

- Si progetti un terminale per automatizzare la spesa in un supermercato. Il terminale deve contare il numero di prodotti messi nel carrello per una spesa massima di 100€.

SOLUZIONE

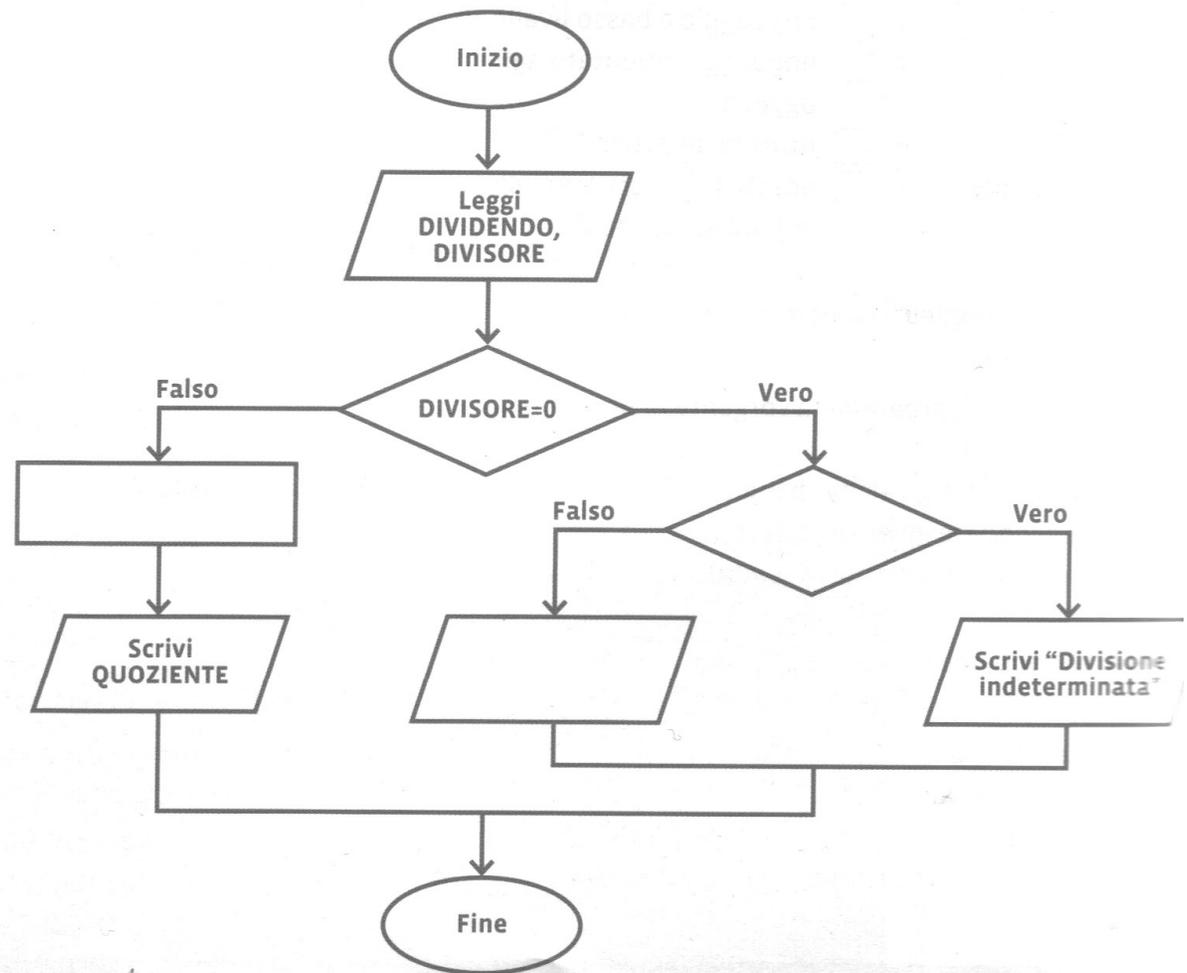
- Analizzare il problema e descrivere la soluzione con diagrammi di flusso

Spesa di prodotti per max 100€



Divisione tra due numeri (per casa)

- Completa il diagramma
- Verifica se è impossibile (divisore=0) o indeterminata (dividendo=0 e divisore=0)





Programmazione

- **Programma** → codifica dell'algoritmo in un linguaggio compilabile in linguaggio macchina
- **Linguaggio di programmazione** (es. C)
→ linguaggio rigoroso che permette la formalizzazione di un algoritmo in un programma
- Vediamo un esempio...



Riepilogo comandi Shell

- `pwd` → visualizza directory corrente;
- `cd <dirname>` → `dirname` diventa la directory corrente.
- `ls <dirname>` → visualizza il contenuto di una directory.
- `mkdir <dirname>` → crea una directory;
- `rmdir <dirname>` → elimina una directory (solo se vuota);
- `touch <filename>` → crea un file (se non esistente);
- `cp <source> <dest>` → copia il file `source` in `dest`;
- `cp -r <source> <dest>` → copia la directory `source` in `dest`;
- `mv <oldname> <newname>` rinomina/sposta file e directory;
- `rm <filename>` → elimina un file;
- `rm -r <dirname>` → elimina una directory e il suo contenuto



Organizzate la vostra home

- Per ogni lezione create una cartella
 - La scorsa volta avevamo creato lez1 e relative sottocartelle, rifare per esercizio
 - Ora create la nuova cartella lez2. Come?
 - `cd`
 - `mkdir lez2`
 - Entrate nella cartella appena creata
 - `cd lez2`

Il mio primo programma

- [Vedere slides GCC.pdf](#)
- Creare il file mainvuoto.c (`touch`)
- Scrivere un main vuoto (`gedit`)
- Farne 1 copia (`cp`)
- Rinominarla hello.c (`mv`)
- Modificarlo in modo che stampi "Hello world!" (`gedit`)
- Compilare (`gcc`) ed eseguire

```
int main()
{
}
```

```
#include <stdio.h>
int main()
{
    printf("Hello world!\n");
}
```



Nota sull'editor di testi `gedit`

- E' un semplice editor di free text
- Non c'è formattazione
- Alcune parole sono formattate secondo opzioni regolabili nell'apposito menù
 - ad esempio, indicando all'editor che stiamo scrivendo del codice C, lui ci aiuta facendoci visualizzare le parole chiave a colori, ma il codice rimane comunque non formattato



Come esercitarsi da casa

- Senza dover installare un sistema Unix sul vostro computer → UBUNTU CD Live
- <http://www.ubuntu.com/download/ubuntu/download>
- Seguire i passi 1, 2 e 3 indicati sul sito
 1. Scaricare Ubuntu
 2. Masterizzare il cd
 3. Riavviare il pc con il cd all'interno
- Per ogni dubbio agiordani@disi.unitn.it