

NOZIONI BASE PER ESERCITAZIONI

DIPARTIMENTO DI FISICA



SAPIENZA
UNIVERSITÀ DI ROMA

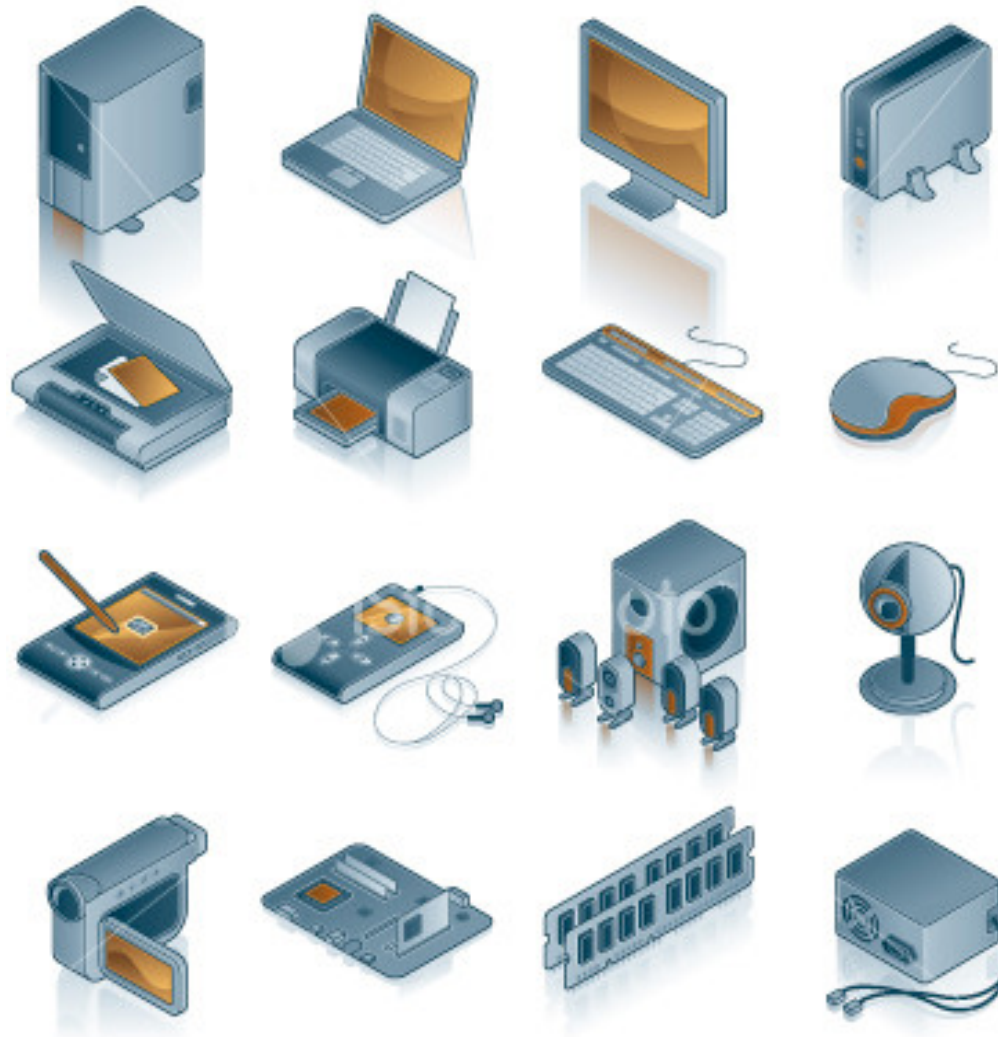
Shahram Rahatlou

Laboratorio di Calcolo, Anno Accademico 2025-2026

<http://www.roma1.infn.it/people/rahatlou/labcalc/>

Sistema Operativo

Hardware

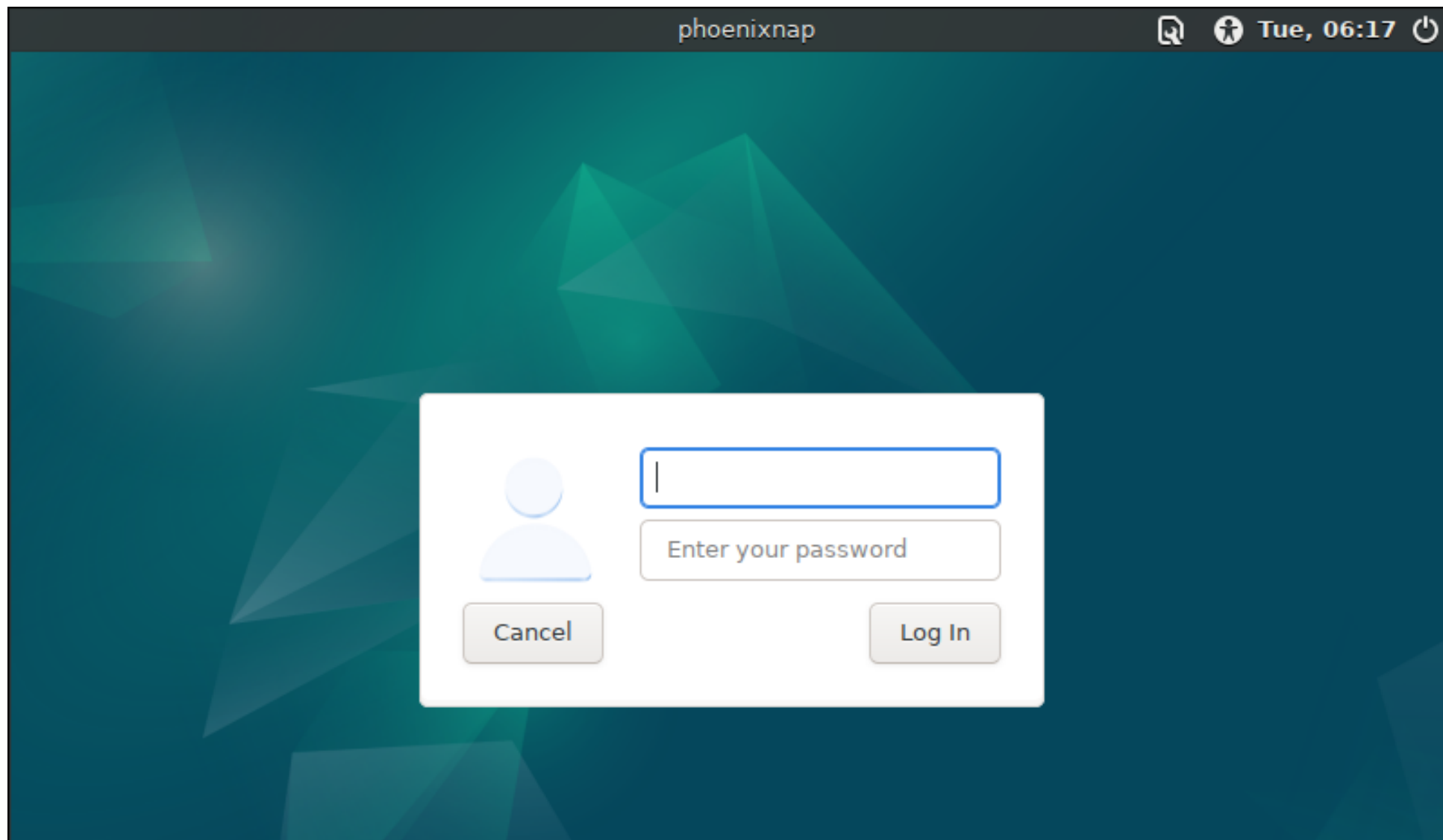


Software



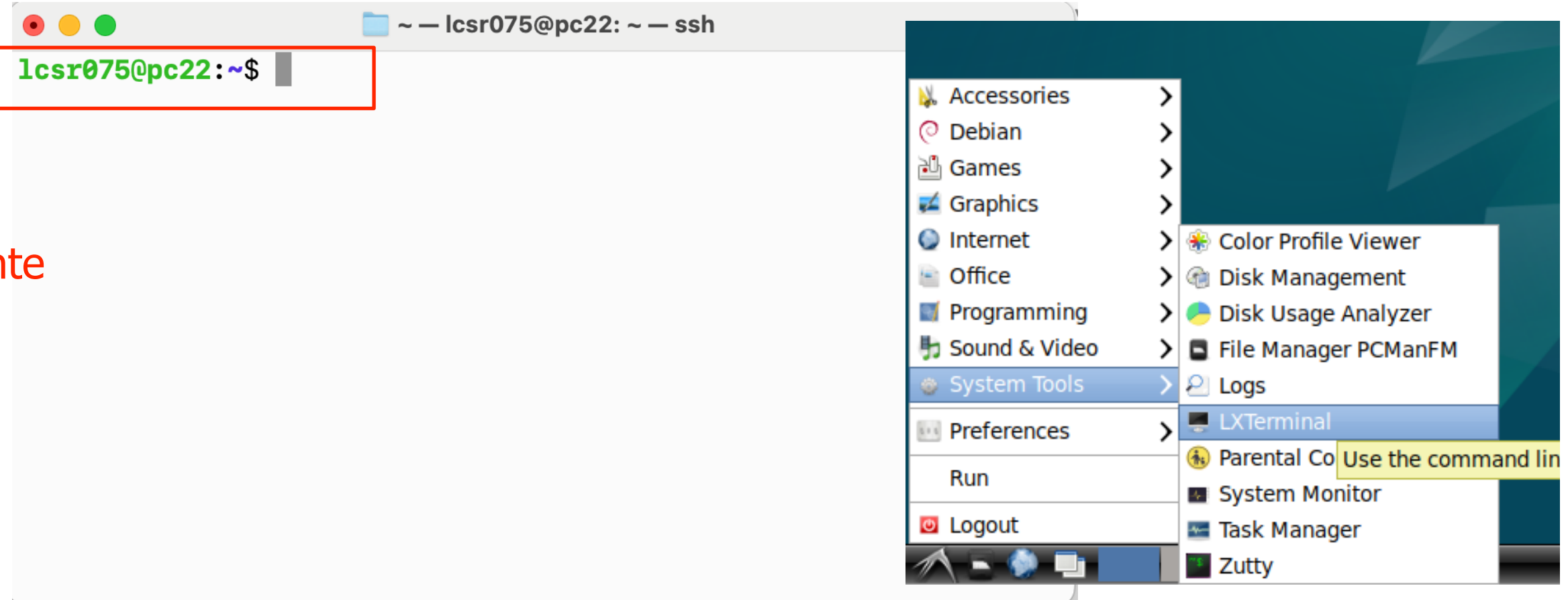
- Routine e programmi di base per interagire con le componenti (hardware) di un calcolatore
 - Accedere alle risorse come il disco fisso
 - Utilizzare le periferiche per Input/Output (I/O)
 - Mouse, tastiera, schermo
 - Compilare programmi, creare eseguibili e girarli

Accesso al sistema (Login)



- Per l'accesso servono uno **username** ed una **password**
- A ciascun gruppo e' assegnato uno username del tipo ***lcsrXXX*** dove ***XXX=001, 002, ..., 099***

Terminale e Shell



- L'utente può impartire comandi al sistema utilizzando la **shell**
- La **shell** è accessibile all'interno di una finestra di riga di comando chiamata comunemente il terminale

Home Directory

A terminal window with a title bar that reads '~ — lcsr075@pc22: ~ — ssh'. The terminal shows a prompt 'lcsr075@pc22:~\$' followed by the command 'pwd'. The output is '/home/lcsr075'. The prompt is then shown again as 'lcsr075@pc22:~\$' with a cursor.

```
~ — lcsr075@pc22: ~ — ssh
lcsr075@pc22:~$ pwd
/home/lcsr075
lcsr075@pc22:~$
```

- Aperta una shell l'utente si trova nella sua area 'home directory
- Si tratta di uno spazio assegnato a ciascun utente dove immagazzinare i suoi file
- Gli altri utenti (se autorizzati) possono vedere i file ma non modificarli
- E' possibile vedere il percorso(path) della directory con il comando **pwd**

Lista dei file e cartelle con `ls`

```
labcalc:/home/rahatlou
File Edit View Terminal Help
[ppp30@labcalc ~]$ ls
a.out      Desktop   prova.c   teach.ico test.o
data2.txt  main     prova.c~  test.c   test.txt~
data.txt   mario    prova.o   test.c~  WeightedAverage.pdf
[ppp30@labcalc ~]$
```

- Potete elencare la lista dei file e cartelle contenuti nella cartella in cui vi trovate con il comando `ls`
- Alcuni tipi di file potrebbero essere visualizzati con un colore
 - Non è proprietà del file ma dipende dalla configurazione della shell

Navigazione nelle cartelle (directory)

```
labcalc.fisica.uniroma1.it:/home/docenti/rahatlou — ssh — 85x26
$ pwd
/home/docenti/rahatlou
$ ls
animated_favicon1.gif  Desktop/  labcalc@  public_html/  tiburtina.tgz
C/                    favicon.ico  private/  root@
C++/                  index.html@  programmazione++@  temp/
$ cd C
$ pwd
/home/docenti/rahatlou/C
$ ls
mularray.c  multarr2.c  nav18*  ptr18*  ptr21*  ptr.exe*  testreturn.c
mularray.o  multarr2.o  nave1.18.c  ptr18.c  ptr.c  testreturn*  uso_++.c
$ cd ..
$ pwd
/home/docenti/rahatlou
```

comando **ls** per vedere la lista dei file in una cartella

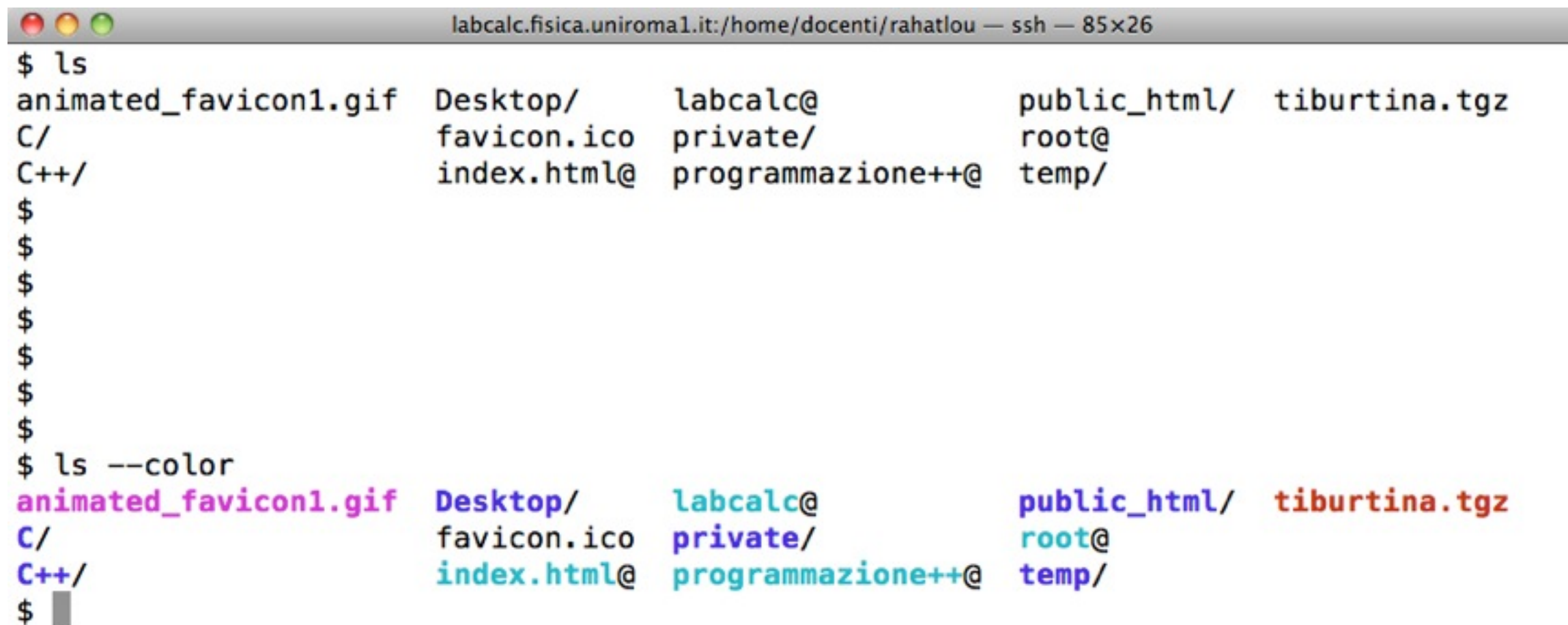
comando **cd** per spostarsi in una nuova cartella

(punto punto) **..** : un simbolo per tornare rapidamente alla cartella contenente quella attuale

comando **pwd** per vedere il percorso della cartella dove ci troviamo

Colori non sono importanti!

- Molti terminali in diversi sistemi operativi mostrano file di tipi diversi con colori diversi
- Solo un artificio grafico per aiutare l'utente
- Sistema operativo non sa nulla e non usa i colori!



```
labcalc.fisica.uniroma1.it:/home/docenti/rahatlou — ssh — 85x26
$ ls
animated_favicon1.gif  Desktop/      labcalc@      public_html/  tiburtina.tgz
C/                    favicon.ico  private/      root@
C++/                  index.html@  programmazione++@ temp/
$
$
$
$
$
$
$
$
$
$
$
$ ls --color
animated_favicon1.gif  Desktop/      labcalc@      public_html/  tiburtina.tgz
C/                    favicon.ico  private/      root@
C++/                  index.html@  programmazione++@ temp/
$
```


Opzioni utili di ls

```
labcalc:/home/rahatlou
File Edit View Terminal Help
[ppp30@labcalc ~]$ ls -ltr
total 124
-rwxr-xr-x  1 ppp30  studenti    453 Apr 29 00:46 data.txt
-rwxr-xr-x  1 ppp30  studenti     63 May  9 18:33 data2.txt
-rwxr-xr-x  1 ppp30  studenti 43049 May 12 15:24 main
-rw-r--r--  1 ppp30  studenti 13066 May 12 15:29 WeightedAverage.pdf
-rw-r--r--  1 ppp30  studenti  5430 Jun  9 09:27 teach.ico
-rw-r--r--  1 ppp30  studenti   54 Oct  1 12:05 test.txt~
-rw-r--r--  1 ppp30  studenti  116 Oct  1 12:26 test.c~
-rw-r--r--  1 ppp30  studenti  118 Oct  1 12:27 test.c
-rw-r--r--  1 ppp30  studenti  912 Oct  1 16:08 test.o
-rwxr-xr-x  1 ppp30  studenti 4770 Oct  1 16:10 mario
-rw-r--r--  1 ppp30  studenti  706 Oct  1 16:24 prova.o
-rw-r--r--  1 ppp30  studenti  156 Oct  1 16:21 prova.c~
-rw-r--r--  1 ppp30  studenti  147 Oct  1 16:33 prova.c
-rwxr-xr-x  1 ppp30  studenti 4803 Oct  1 16:33 a.out
drwxr-xr-x  3 ppp30  studenti 4096 Oct  1 16:46 Desktop
[ppp30@labcalc ~]$
```

Permessi del file

Data e l'ora in cui il file e' stato modificato l'ultima volta

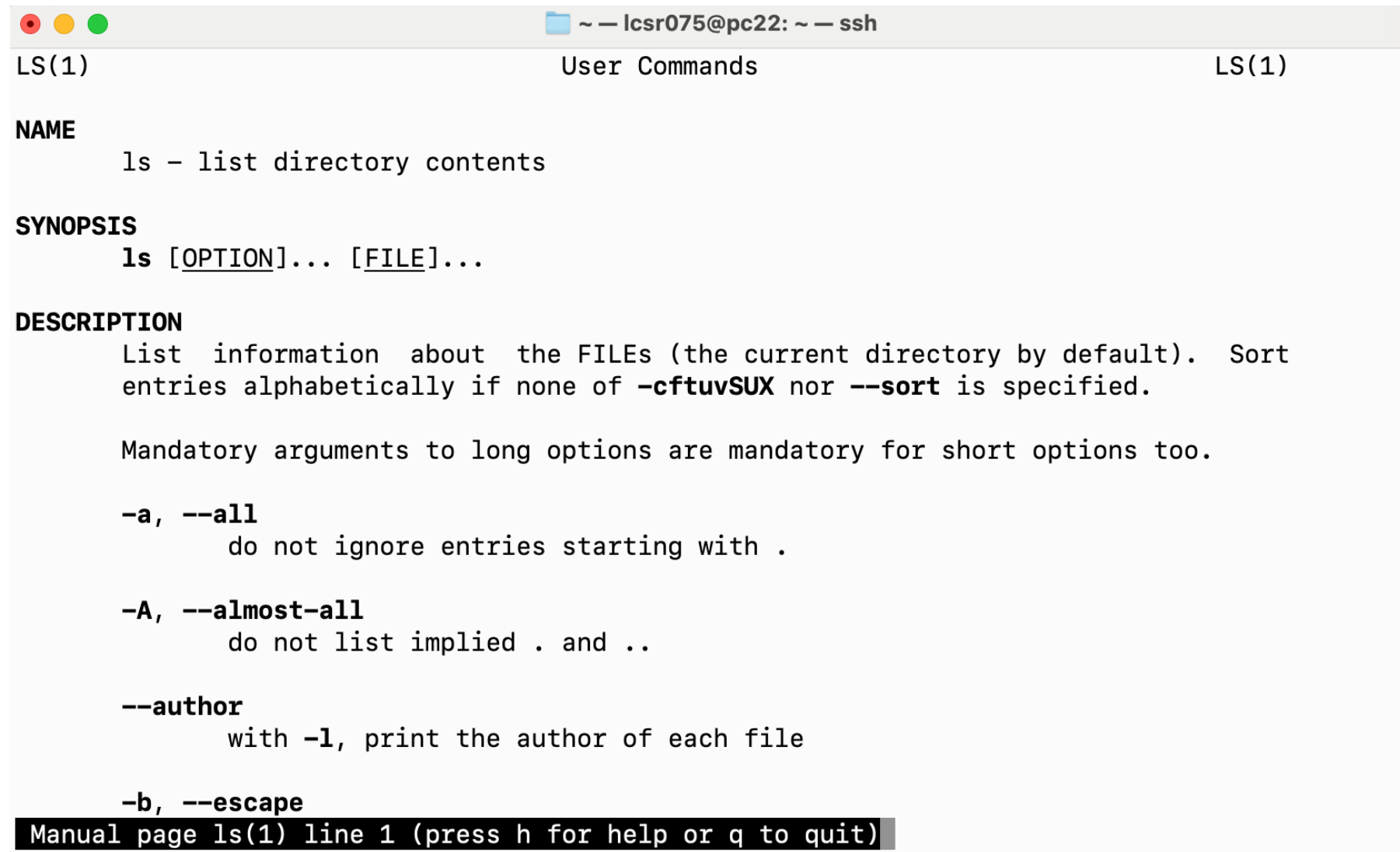
Utente proprietario del file

Gruppo dell'utente

■ Ci sono diverse opzioni di ls per aumentare il livello di informazione o ordinare l'output

- **-l**: informazioni dettagliate come l'ora e la data della modifica ed i permessi
- **-t**: ordina i file secondo data della modifica. Primi file più recenti
- **-r**: inverte l'ordine di **-t**; primi i file più vecchi

Trovare altre opzioni con **man**



```
LS(1) User Commands LS(1)

NAME
  ls - list directory contents

SYNOPSIS
  ls [OPTION]... [FILE]...

DESCRIPTION
  List information about the FILES (the current directory by default). Sort
  entries alphabetically if none of -cftuvSUX nor --sort is specified.

  Mandatory arguments to long options are mandatory for short options too.

  -a, --all
    do not ignore entries starting with .

  -A, --almost-all
    do not list implied . and ..

  --author
    with -l, print the author of each file

  -b, --escape
    Manual page ls(1) line 1 (press h for help or q to quit)
```

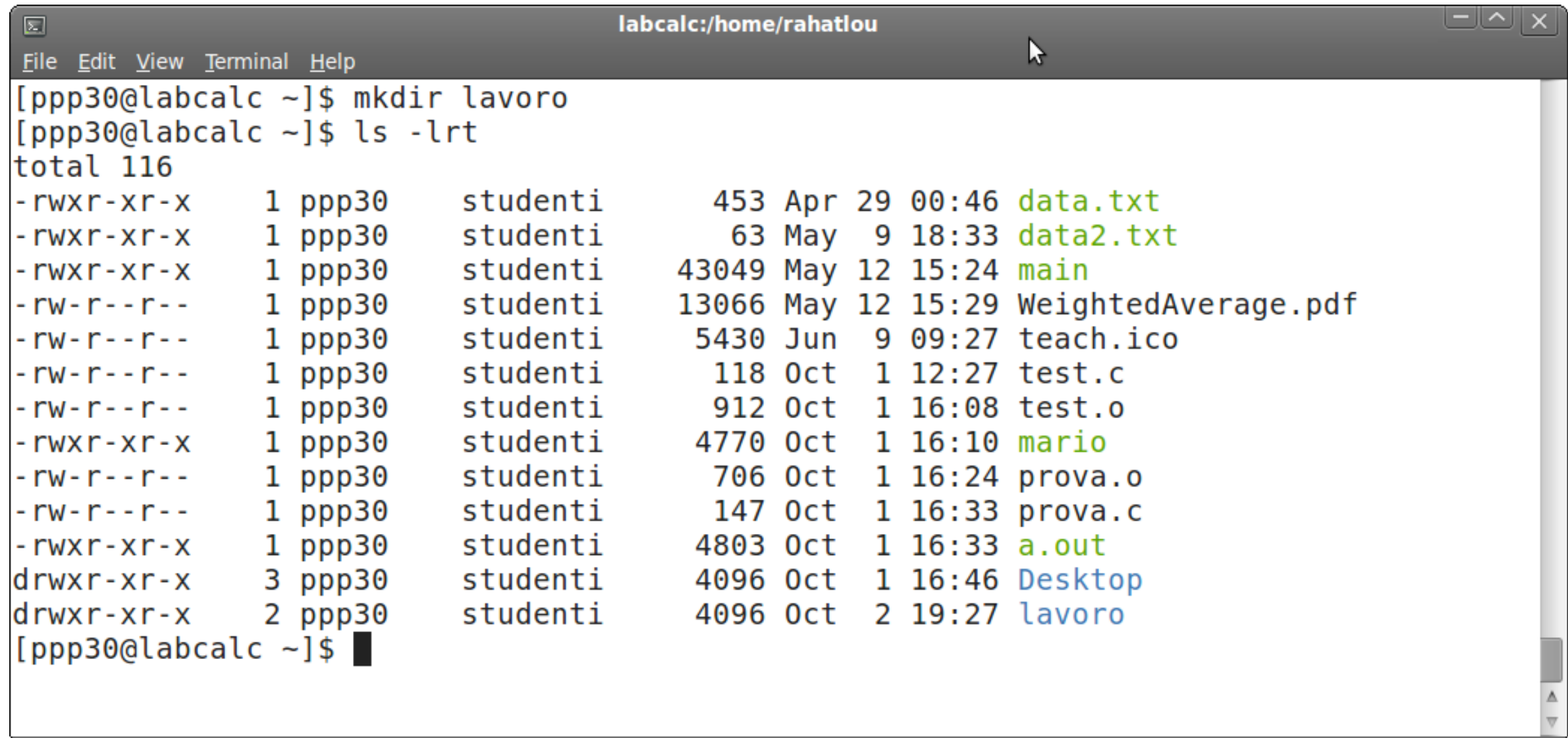
- Per tutti i comandi di Linux potete accedere al manuale con il comando **man** ad esempio

man ls

risultata nella schermata di sopra

- Per uscire dal manuale premere il tasto **q**

Creare cartelle con `mkdir`



A terminal window titled 'labcalc:/home/rahatlou' showing the execution of the `mkdir` and `ls` commands. The user `ppp30` is in the home directory `~`. The `mkdir lavoro` command is executed, and then `ls -lrt` is used to list the directory contents. The output shows a list of files and directories with their permissions, owner, group, size, and timestamps. The newly created directory `lavoro` is visible at the bottom of the list.

```
labcalc:/home/rahatlou
File Edit View Terminal Help
[ppp30@labcalc ~]$ mkdir lavoro
[ppp30@labcalc ~]$ ls -lrt
total 116
-rwxr-xr-x  1 ppp30  studenti    453 Apr 29 00:46 data.txt
-rwxr-xr-x  1 ppp30  studenti     63 May  9 18:33 data2.txt
-rwxr-xr-x  1 ppp30  studenti 43049 May 12 15:24 main
-rw-r--r--  1 ppp30  studenti 13066 May 12 15:29 WeightedAverage.pdf
-rw-r--r--  1 ppp30  studenti  5430 Jun  9 09:27 teach.ico
-rw-r--r--  1 ppp30  studenti   118 Oct  1 12:27 test.c
-rw-r--r--  1 ppp30  studenti   912 Oct  1 16:08 test.o
-rwxr-xr-x  1 ppp30  studenti  4770 Oct  1 16:10 mario
-rw-r--r--  1 ppp30  studenti   706 Oct  1 16:24 prova.o
-rw-r--r--  1 ppp30  studenti   147 Oct  1 16:33 prova.c
-rwxr-xr-x  1 ppp30  studenti  4803 Oct  1 16:33 a.out
drwxr-xr-x  3 ppp30  studenti  4096 Oct  1 16:46 Desktop
drwxr-xr-x  2 ppp30  studenti  4096 Oct  2 19:27 lavoro
[ppp30@labcalc ~]$
```

- Potete creare cartelle (o sotto-cartelle) con un nome

Copiare i file con cp

- Potete creare una copia di un file esistente nella cartella in cui vi trovate oppure in una nuova cartella

Crea un file vuoto

```
~ — rahatlou@labcalc2: ~/esempio — ssh
[rahatlou@labcalc2:~/esempio$ ls
[rahatlou@labcalc2:~/esempio$
[rahatlou@labcalc2:~/esempio$ touch prova.txt
[rahatlou@labcalc2:~/esempio$ ls
prova.txt
[rahatlou@labcalc2:~/esempio$ cp prova.txt esempio.txt
[rahatlou@labcalc2:~/esempio$
[rahatlou@labcalc2:~/esempio$ ls
esempio.txt  prova.txt
[rahatlou@labcalc2:~/esempio$ mkdir lavoro
[rahatlou@labcalc2:~/esempio$ ls
esempio.txt  lavoro  prova.txt
[rahatlou@labcalc2:~/esempio$
[rahatlou@labcalc2:~/esempio$ cp prova.txt lavoro/
[rahatlou@labcalc2:~/esempio$ ls
esempio.txt  lavoro  prova.txt
[rahatlou@labcalc2:~/esempio$
[rahatlou@labcalc2:~/esempio$
[rahatlou@labcalc2:~/esempio$ ls lavoro/
prova.txt
[rahatlou@labcalc2:~/esempio$
```

Spostare o rinominare i file con `mv`

- Si usa un unico comando per
 - Rinominare un file
 - Spostare un file da una directory in un'altra

```
~ — rahatlou@labcalc2: ~/esempio — ssh
[rahatlou@labcalc2:~/esempio$ touch prova.txt
[rahatlou@labcalc2:~/esempio$
[rahatlou@labcalc2:~/esempio$ ls
prova.txt
[rahatlou@labcalc2:~/esempio$
[rahatlou@labcalc2:~/esempio$ mv prova.txt elenco.txt
[rahatlou@labcalc2:~/esempio$ ls
elenco.txt
[rahatlou@labcalc2:~/esempio$ mkdir classe
[rahatlou@labcalc2:~/esempio$
[rahatlou@labcalc2:~/esempio$ mv elenco.txt classe/
[rahatlou@labcalc2:~/esempio$ ls
classe
[rahatlou@labcalc2:~/esempio$ cd class
-bash: cd: class: No such file or directory
[rahatlou@labcalc2:~/esempio$ cd classe
[rahatlou@labcalc2:~/esempio/classe$
[rahatlou@labcalc2:~/esempio/classe$ ls
elenco.txt
[rahatlou@labcalc2:~/esempio/classe$ cd ..
[rahatlou@labcalc2:~/esempio$ ls
classe
[rahatlou@labcalc2:~/esempio$ ls classe/
elenco.txt
[rahatlou@labcalc2:~/esempio$
```


Cancellare (senza poterci ripensare) con `rm`

- Ricordatevi che una volta rimosso un file o una cartella non c'è modo di recuperarli
- Rimuovere una cartella rimuove anche tutti i file che contiene

```
~ — rahatlou@labcalc2: ~/esempio — ssh

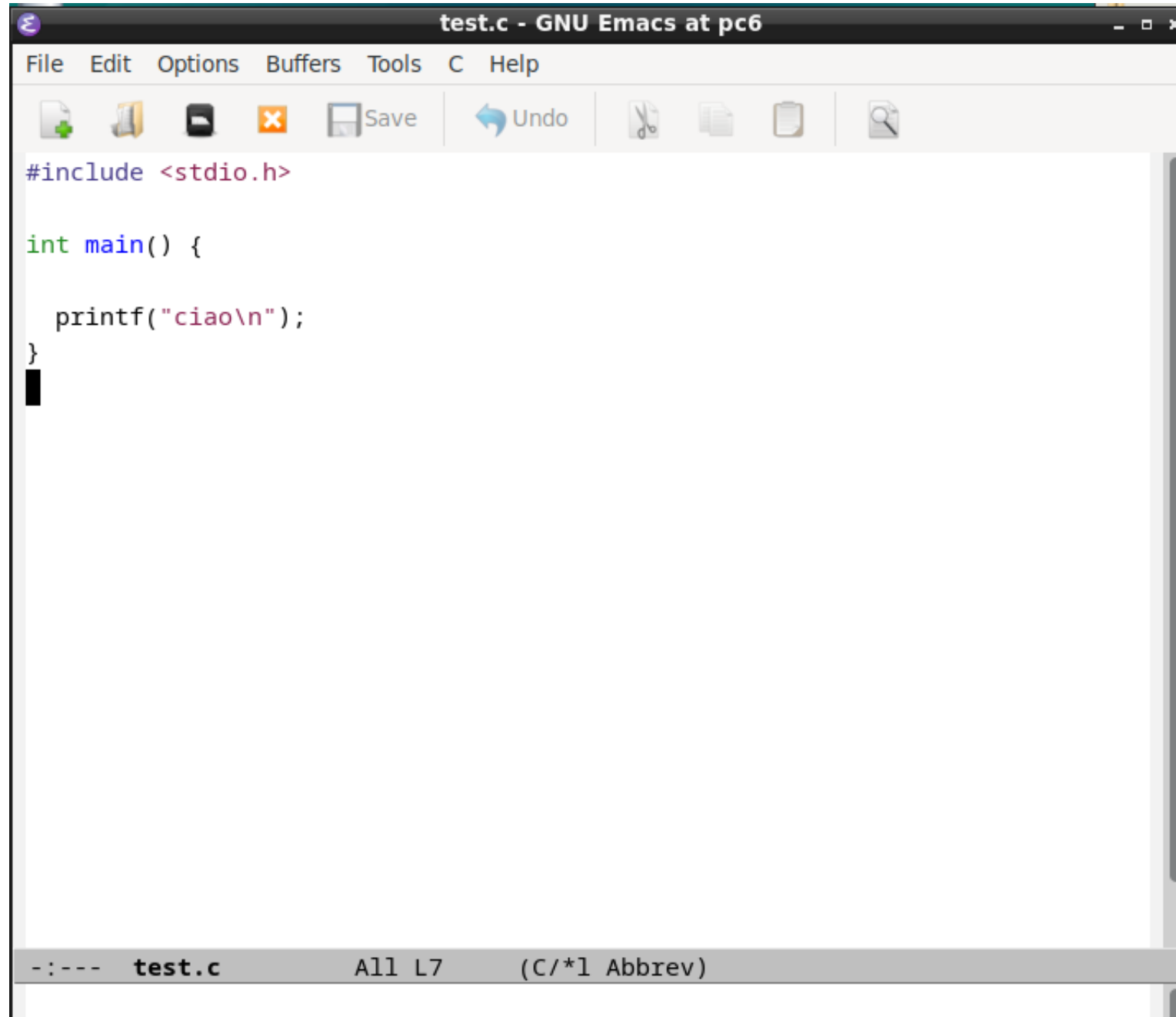
[rahatlou@labcalc2:~/esempio$ ls
classe  prova.txt
[rahatlou@labcalc2:~/esempio$ rm prova.txt
[rahatlou@labcalc2:~/esempio$ 
[rahatlou@labcalc2:~/esempio$ ls
classe
[rahatlou@labcalc2:~/esempio$ rm classe
rm: cannot remove 'classe': Is a directory
[rahatlou@labcalc2:~/esempio$ 
[rahatlou@labcalc2:~/esempio$ ls classe/
elenco.txt
[rahatlou@labcalc2:~/esempio$ 
[rahatlou@labcalc2:~/esempio$ rm -R classe
[rahatlou@labcalc2:~/esempio$ 
[rahatlou@labcalc2:~/esempio$ ls
[rahatlou@labcalc2:~/esempio$
```

Editor di Testo

- Qualsiasi applicazione per scrivere testo senza formattazione
- Da non confondere con “Word Processor” come Microsoft Word o OpenOffice
 - Oltre al testo queste applicazioni aggiungono informazioni di formattazione quali: font, dimensione, indentatura, impaginazione
- Useremo l'applicazione gratuita **emacs** nelle nostre sessioni di laboratorio

Editor di testo emacs

- Editor che useremo per questo corso



The screenshot shows the GNU Emacs editor window titled "test.c - GNU Emacs at pc6". The menu bar includes File, Edit, Options, Buffers, Tools, C, and Help. The toolbar contains icons for file operations (new, open, save, close), editing (undo, redo, cut, copy, paste), and search. The main text area contains the following C code:

```
#include <stdio.h>

int main() {

    printf("ciao\n");
}
```

The status bar at the bottom displays "test.c", "All L7", and "(C/*1 Abbrev)".

- Per invocarlo dalla riga di comando

```
prompt> emacs nomefile.c
```

Colori nell'editor di testo

- Quasi tutti gli editor usano i colori per evidenziare la sintassi dei linguaggi di programmazione
 - Solo di utilità visuale per l'utente e per riconoscere più facilmente errori comuni come
 - parentesi aperte ma non chiuse: viene evidenziata la parentesi aperta
 - testo commentato: di colore diverso rispetto al testo che verrà compilato
 - colore diverso per le parole chiave del linguaggio
- **main, open, printf, for, exit**
- I compilatori non capiscono e non usano i colori!
 - abituatevi a non dipendere unicamente dai colori per capire il codice

```
#include <stdio.h>
#include <math.h>

#define NBIT 20

int main() {

    int in, cifra=0, resto;
    int binary[NBIT] = {0};

    do{
        printf("Inserisci numero positivo < %d in base 10: ", (int)pow(2,NBIT));
        scanf("%d", &in);
    } while( in <= 0 || in >= pow(2, NBIT) );

    do {

        resto = in % 2;
        printf("%3d-esima cifra: %d \t moltiplica 2 ^%3d (%8.0f)\n",cifra+1, resto, cifra,pow(2,cifra));

        binary[cifra] = resto;
        in /= 2;
        cifra++;

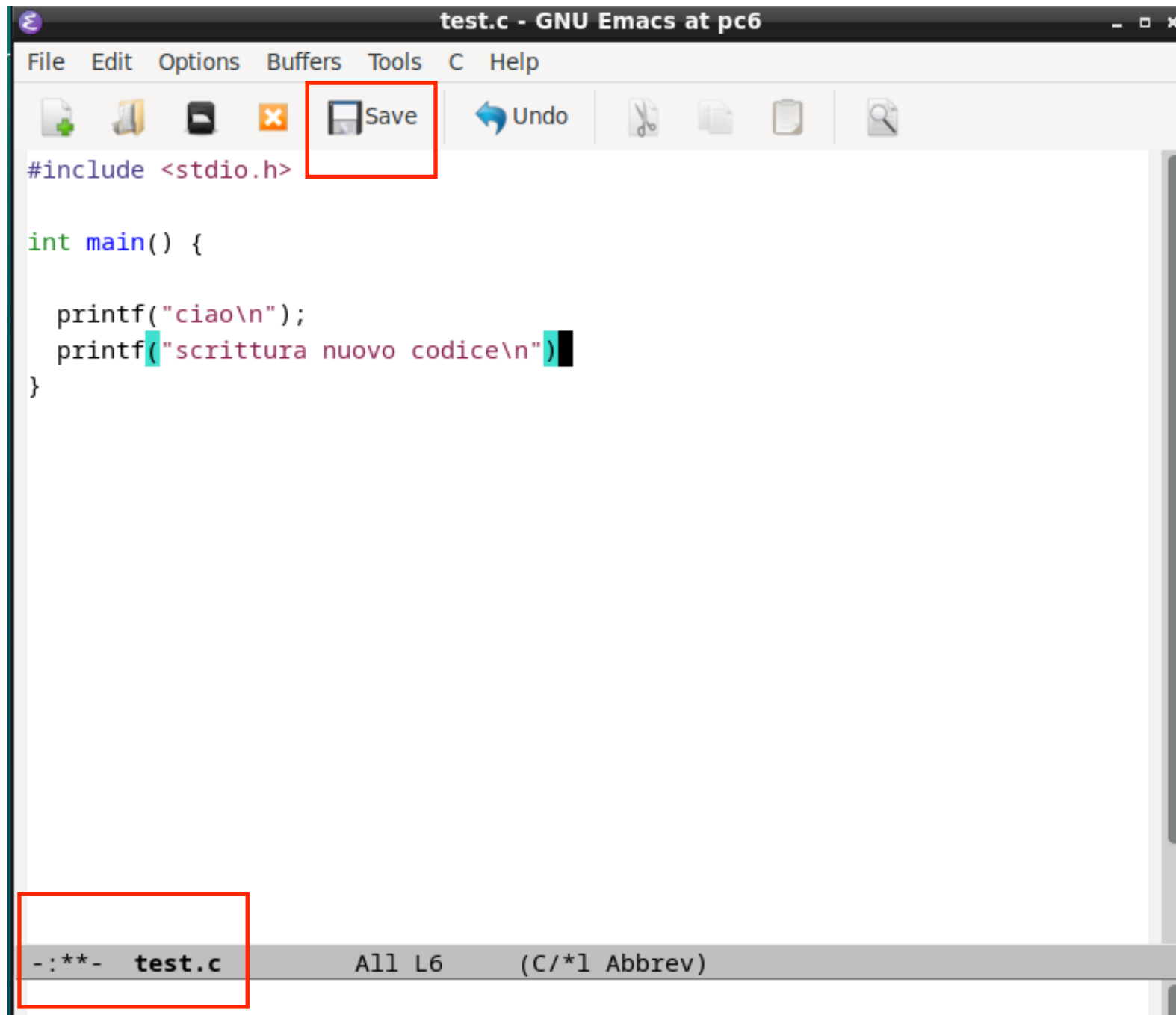
    } while (in != 0);

    printf("base 10: %d \t base 2: ", in);
    for(int i = NBIT-1; i>= 0; i--) {
        printf("%1d", binary[i]);

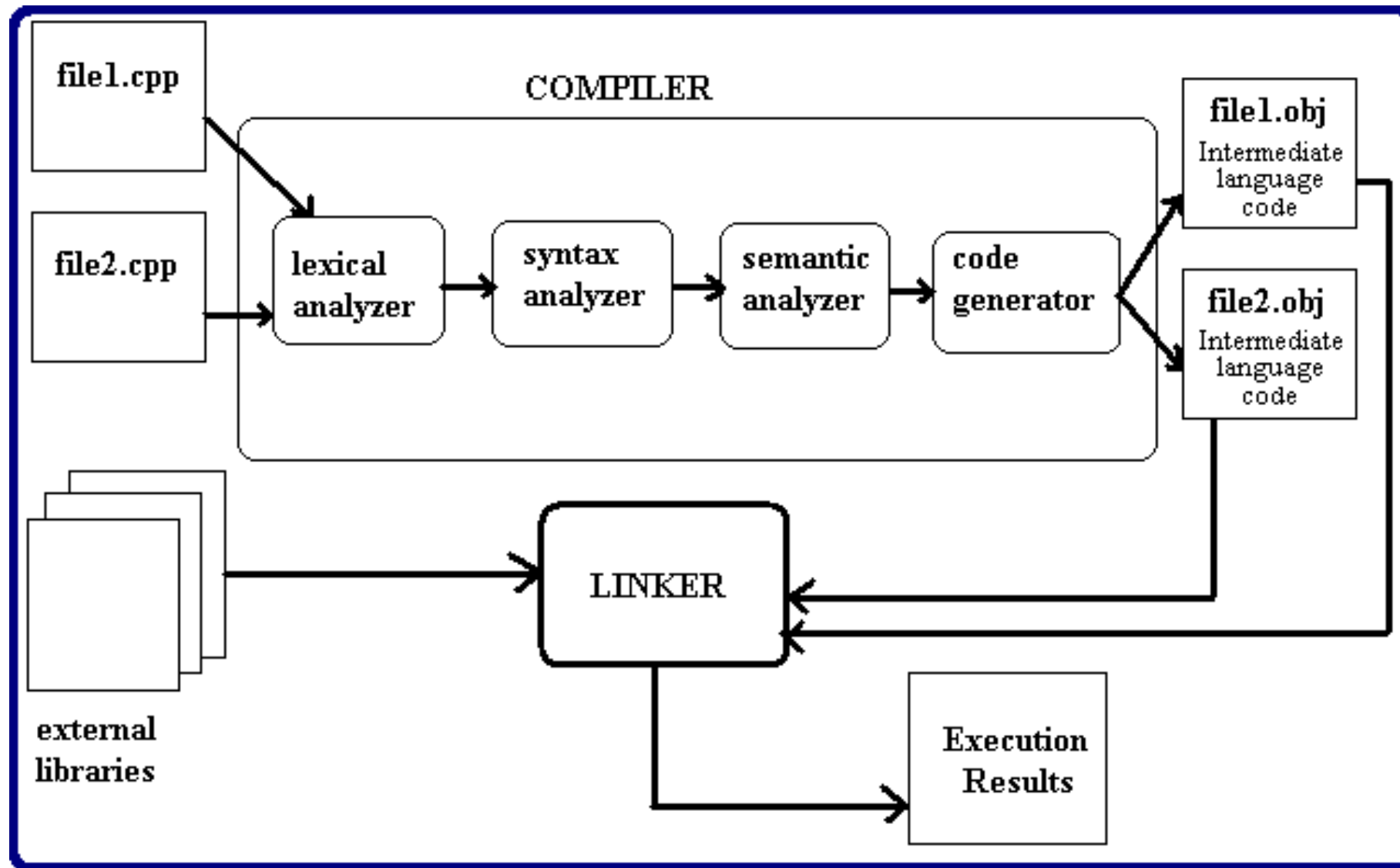
    }
    printf("\n");
}
```

Salvare un nuovo file

- Per creare un nuovo file: emacs test.c
 - si dice nel buffer dell'editor di testo
- Se non salvato sul disco il testo scritto verra` perso chiudendo l'editor
- Abituatevi a salvare regolarmente il file per non perdere le ultime modifiche



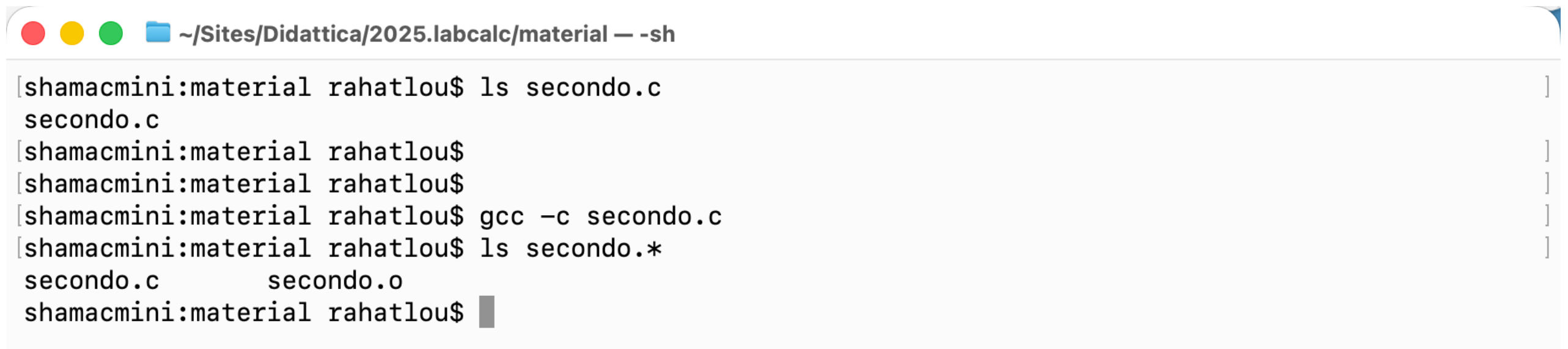
Compilare e 'linkare' eseguibili



- **Compilare:** tradurre istruzioni di linguaggi ad alto livello (C/C++/Java) in istruzioni macchina
- **Linkare:** mettere insieme o collegare diversi pezzi di istruzioni binarie per creare un unico programma eseguibile

Compilazione

- Useremo gcc durante il corso ma potete usare anche altri compilatori
- Per compilare senza linkare (ossia senza creare l'eseguibile)
 - Traduce solo le istruzioni dal linguaggio C in linguaggio macchina



```
~/Sites/Didattica/2025.labcalc/material — -sh
[shamacmini:material rahatlou$ ls secondo.c
secondo.c
[shamacmini:material rahatlou$
[shamacmini:material rahatlou$
[shamacmini:material rahatlou$ gcc -c secondo.c
[shamacmini:material rahatlou$ ls secondo.*
secondo.c      secondo.o
shamacmini:material rahatlou$
```

- Per ciascun file di sorgente (estensione .c) il compilatore genera un object file (estensione .o)

Compilare e linkare l'eseguibile allo stesso tempo

- L'opzione `-o` specifica il nome dell'eseguibile
 - Altrimenti di solito il linker lo chiamerà `a.out` oppure `a.exe`
 - Non si usa l'opzione `-c` proprio per invocare il linker dopo la compilazione
- Si può fare l'eseguibile solo per un file `.c` che contenga una funzione `main`

```
~/Sites/Didattica/2025.labcalc/material — -sh
[shamacmini:material rahatlou$ ls secondo.c
secondo.c
[shamacmini:material rahatlou$
[shamacmini:material rahatlou$
[shamacmini:material rahatlou$ gcc -c secondo.c
[shamacmini:material rahatlou$ ls secondo.*
secondo.c      secondo.o
[shamacmini:material rahatlou$ gcc -Wall -o app.exe secondo.c
[shamacmini:material rahatlou$ ls app.exe
app.exe
shamacmini:material rahatlou$
```

Sommario uso compilatore

Comando	Commento
<code>gcc -Wall -c prova.c</code>	compila il file sorgente <code>prova.c</code> e crea il file oggetto <code>prova.o</code> Non c'è nessun eseguibile
<code>gcc -Wall -o app.exe prova.c</code>	compila e linka il file sorgente <code>prova.c</code> e crea il file eseguibile <code>app.exe</code> Non c'è nessun <code>file.o</code> in questo caso
<code>gcc -Wall -lm -o app.exe prova.c</code>	compila e linka il file sorgente <code>prova.c</code> utilizzando la libreria matematica (opzione <code>-lm</code>) e crea il file eseguibile <code>app.exe</code> Non c'è nessun <code>file.o</code> in questo caso
<code>gcc -Wall -o app.exe prova.c grafica.o</code>	compila il file sorgente <code>prova.c</code> e crea il file eseguibile <code>app.exe</code> utilizzando anche il contenuto del file <code>grafica.o</code> (compilato in precedenza)
<code>gcc -Wall -o app.exe prova.c grafica.c</code>	compila i due file sorgente <code>prova.c</code> e <code>grafica.c</code> e crea il file eseguibile <code>app.exe</code> Non viene creato nessun file oggetto

GIRARE IL PROGRAMMA ESEGUIBILE

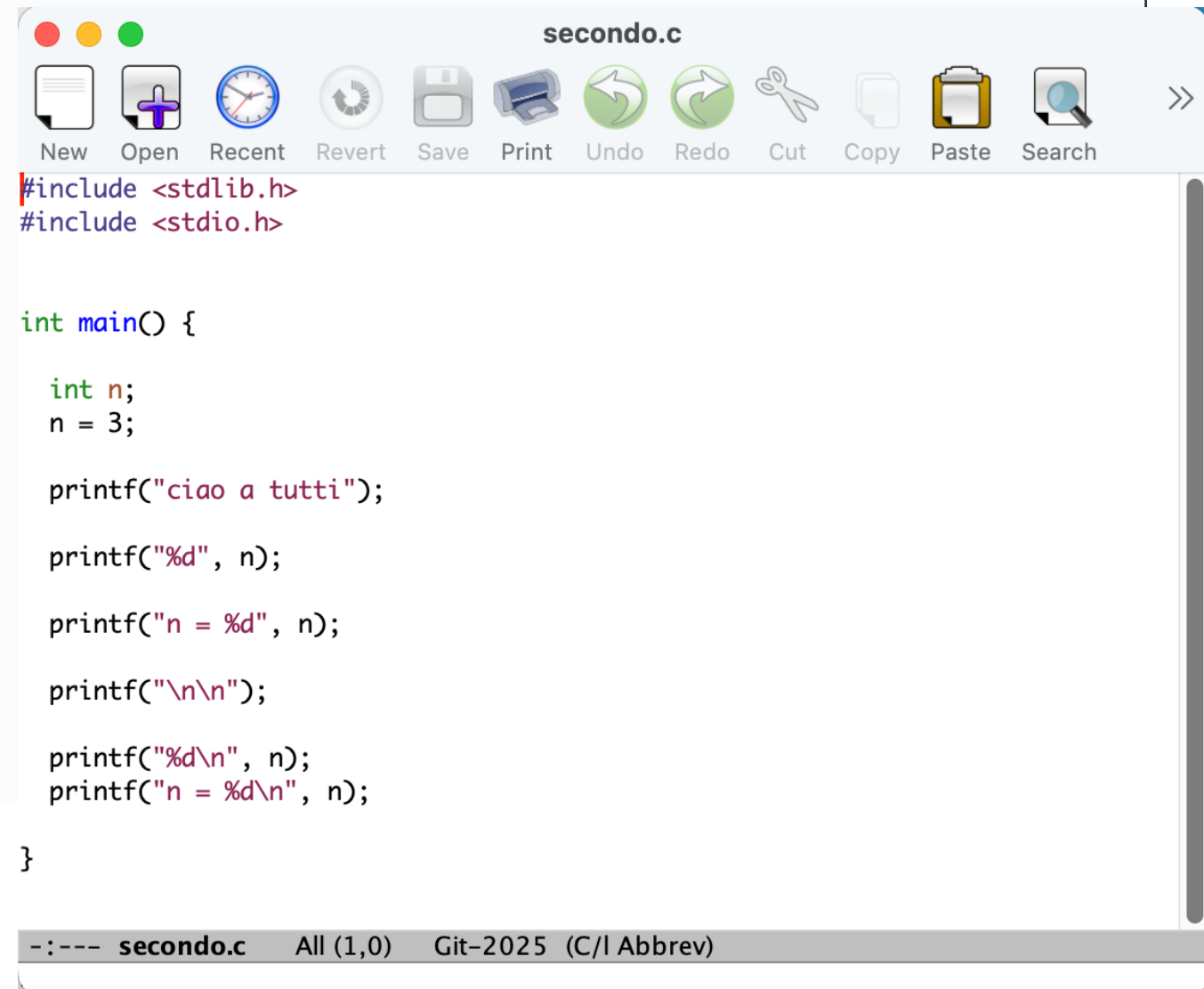
- Bisogna trovarsi nella cartella in cui si trova l'eseguibile

~/Sites/Didattica/2025.labcalc/material — -sh

```
[shamacmini:material rahatlou$ gcc -Wall -o app.exe secondo.c -lm
[shamacmini:material rahatlou$
[shamacmini:material rahatlou$
[shamacmini:material rahatlou$ ls app.exe
app.exe
[shamacmini:material rahatlou$
[shamacmini:material rahatlou$
[shamacmini:material rahatlou$
[shamacmini:material rahatlou$ ./app.exe
ciao a tutti3n = 3
```

Notate bene ./ prima del nome dell'eseguibile

```
3
n = 3
[shamacmini:material rahatlou$
[shamacmini:material rahatlou$
[shamacmini:material rahatlou$ app.exe
-sh: app.exe: command not found
shamacmini:material rahatlou$
```



The screenshot shows a code editor window titled 'secondo.c'. The editor has a menu bar with icons for New, Open, Recent, Revert, Save, Print, Undo, Redo, Cut, Copy, Paste, and Search. The code is as follows:

```
#include <stdlib.h>
#include <stdio.h>

int main() {

    int n;
    n = 3;

    printf("ciao a tutti");

    printf("%d", n);

    printf("n = %d", n);

    printf("\n\n");

    printf("%d\n", n);
    printf("n = %d\n", n);

}
```

The status bar at the bottom shows: -:--- secondo.c All (1,0) Git-2025 (C/I Abbrev)