

# **Impianti di Elaborazione (allievi Gestionali)**

Giuseppe Pozzi

Impianti di Elaborazione per Allievi Gestionali - Como  
Facoltà di Ingegneria dell'Informazione  
Politecnico di Milano

giuseppe.pozzi@polimi.it  
- versione del 7 novembre 2003 -

## **Laboratori del corso di Impianti di Elaborazione**

## Premesse

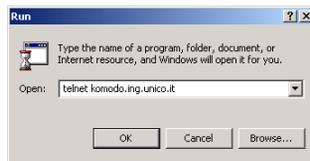
- Ogni studente dispone di un propria login e password sulla macchina komodo.ing.unico.it (131.175.57.3)
- Ogni studente edita, compila, ed esegue sulla **macchina locale** linux (quindi non su komodo) il proprio programma.
- Il nome del file sorgente e' dato dal numero di matricola dello studente seguito da "-1.c", "-2.c", "-3.c", "-... .c" per il primo, il secondo ... codice di prova di laboratorio.
- Ogni studente al termine del laboratorio invia il file sorgente del proprio programma tramite la procedura da eseguire sulla macchina linux:
  - `consegna nome_file_sorgente proprio_account_su_komodo`

## Principali comandi Unix

- Riferimenti:
  - per una guida di riferimento veloce sul sistema Linux:
    - Siever E., Linux in a Nutshell - A Desktop Quick Reference, 2<sup>nd</sup> ed. O'Reilly, Sebastopol, CA -USA-, 1999.
    - Welsh M., Kaufman L., Il manuale di Linux. Jackson libri, 1997.
  - per un sommario dei principali comandi Unix/Linux:
    - Vermeir D., Unix for Beginners:  
<http://tinf2.vub.ac.be/~dvermeir/manuals/uintro/uintro.html>
    - E' poi sempre possibile utilizzare su macchine Unix il comando `man`, specificando l'argomento relativo al quale si richiede la visualizzazione del manuale comandi (ad es. `man ls` per avere descrizione sul comando `ls` che visualizza i file di un direttorio).

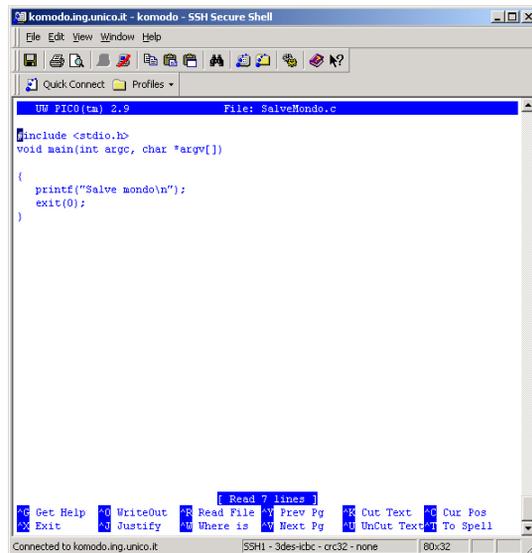
```
tera Term - morgana.elet.polimi.it VT
File Edit Setup Control Window Help
/home/pozzi> man fork
Reformatting page. Please wait... completed
fork(2) fork(2)
NAME
fork, vfork - Creates a new process
SYNOPSIS
#include <unistd.h>
pid_t fork(void);
pid_t vfork(void);
Application developers may want to specify an #include statement for
<sys/types.h> before the one for <unistd.h> if programs are being developed
for multiple platforms. The additional #include statement is not required
on Tru64 UNIX systems or by ISO or X/Open standards, but may be required on
other vendors' systems that conform to these standards.
STANDARDS
Interfaces documented on this reference page conform to industry standards
as follows:
fork(): POSIX.1, XPG4, XPG4-UNIX
vfork(): XPG4-UNIX
Refer to the standards(5) reference page for more information about indus-
try standards and associated tags.
DESCRIPTION
The fork() and vfork() functions create a new process (child process) that
is identical to the calling process (parent process).
The child process inherits the following attributes from the parent pro-
cess:
+ Environment
+ Close-on-exec flags
+ Signal handling settings
```

## Avvio del client telnet su Pc per collegarsi a komodo



# Per editare un file su komodo

- Comandi da digitare al prompt:
  - pico miofile.c  
Apri una sessione di editor creando o leggendo il file miofile.c  
per salvare il file: ctrl-o  
per uscire: ctrl-x



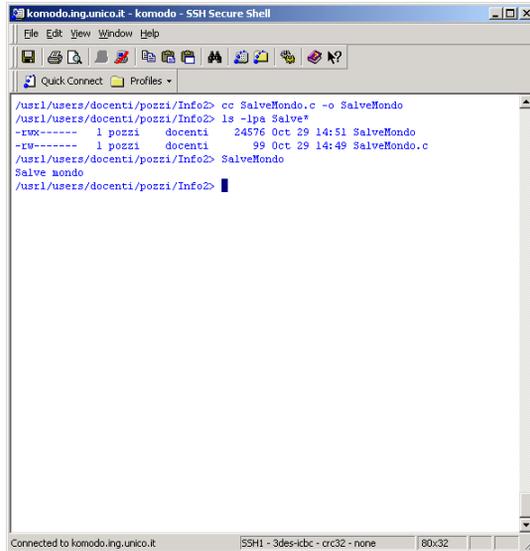
```
komodo.ing.unico.it - komodo - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
UV PICO (tm) 2.9 File: SalveMondo.c
#include <stdio.h>
void main(int argc, char *argv[])
{
    printf("Salve mondo\n");
    exit(0);
}
Read ? lines
Get Help WriteOut Read File Prev Pg Cut Text Cur Pos
Exit Justify Where is Next Pg UnOut Text To Spell
Connected to komodo.ing.unico.it SSH1 - 3des-rc4 - crc32 - none 80x32
```

## Per compilare su komodo

- Comandi da digitare al prompt:
  - `cc miofile.c -o miofile`  
L'opzione "-o" richiede che il file di uscita (e quindi il file eseguibile) si chiami "miofile". In assenza della specificazione, il file avrebbe nome "a.out".

## Per eseguire programmi

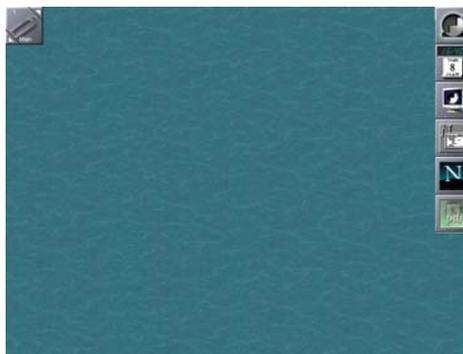
- Digitare al prompt il nome del programma da eseguire, seguito da eventuali parametri:
  - `miofile`



```
komodo.ing.unico.it - komodo - SSH Secure Shell
File Edit View Window Help
Quick Connect Profiles
/usr1/users/docenti/pozzi/Info> cc SalveMondo.c -o SalveMondo
/usr1/users/docenti/pozzi/Info> ls -lpa Salve*
-rwx----- 1 pozzi docenti 24576 Oct 29 14:51 SalveMondo
-rw----- 1 pozzi docenti 99 Oct 29 14:49 SalveMondo.c
/usr1/users/docenti/pozzi/Info> SalveMondo
Salve mondo
/usr1/users/docenti/pozzi/Info> █
```

Connected to komodo.ing.unico.it SSH1 - 3des-icbc - crc32 - none 80x32

## Interfaccia grafica in ambiente linux

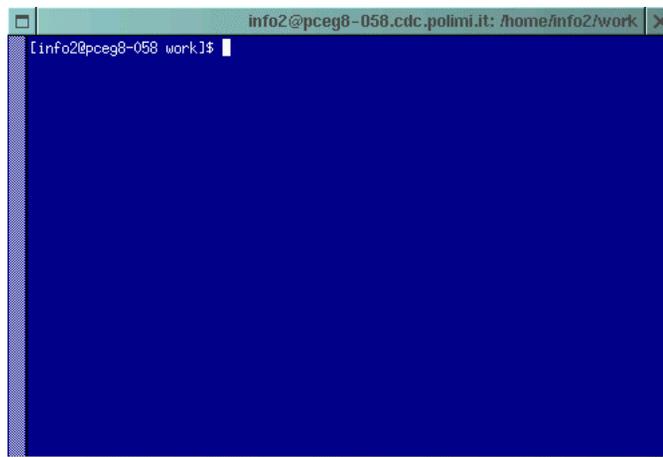


## Alcuni comandi in ambiente linux

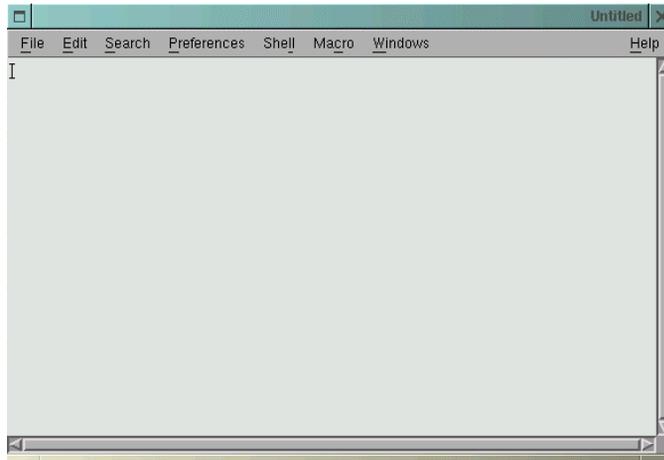
- xterm: per aprire in una nuova finestra un interprete comandi.
- nedit: per aprire un editor di testo da utilizzare per scrivere il codice richiesto.
- Netscape: per aprire un browser internet



## Finestra interprete comandi in ambiente linux



## Finestra nedit in ambiente linux



7 novembre 2003

Impianti di Elaborazione - Introduzione ai laboratori

15

## Gestione finestre in ambiente linux

- Per chiudere le singole applicazioni, utilizzare il bottone X a destra sul bordo superiore della relativa finestra.
- Per ridurre a icona sul desktop le singole applicazioni, utilizzare il bottone a sinistra sul bordo superiore della relativa finestra. Cliccando due volte sull'icona, l'applicazione viene riaperta.
- Per portare in primo piano una certa applicazione, è necessario cliccare sul bordo della relativa finestra.

7 novembre 2003

Impianti di Elaborazione - Introduzione ai laboratori

16

## Chiusura della sessione linux

- Per lasciare l'ambiente grafico, posizionarsi sullo sfondo, premere il tasto destro del mouse e scegliere la voce **exit**. Verrà riproposto il prompt di sistema.
- Per chiudere la sessione, digitare **exit** al prompt di sistema.
- Non spegnere il PC al termine della sessione di lavoro.

## Per compilare in ambiente linux

- Comandi da digitare nella finestra dell'interprete comandi:
  - `cc miofile.c -o miofile`  
L'opzione "-o" richiede che il file di uscita (e quindi il file eseguibile) si chiami "miofile". In assenza della specificazione, il file avrebbe nome "a.out".

## Per eseguire programmi in ambiente linux

- Digitare nella finestra dell'interprete comandi il nome del programma da eseguire, seguito da eventuali parametri:
  - miofile

## Altri comandi

- Creare un direttorio:  
mkdir
- Cambiare direttorio corrente:  
chdir
- Modificare la lista di accesso di un file:  
chmod
- Elenco dei file:  
ls -lpa

## Laboratorio 0

- Nessuna valutazione.
- Tema: editare, compilare ed eseguire i seguenti programmi.
  - stampa a video del messaggio "salve mondo";
  - stampa a video di tutti i parametri passati a un programma attraverso la linea di comando;
  - lettura di stringhe e relativa stampa.

## Salve mondo

```
#include <stdio.h>
void main(int argc, char *argv[])

{
    printf("Salve mondo\n");
    exit(0);
}
```

## Stampa parametri della linea di comando

```
#include <stdio.h>

void main(int argc, char * argv[])
{
    int i;

    printf("\n\n");
    printf("Nome del programma eseguito: %s\n",argv[0]);
    printf("Numero di parametri ricevuti: %d\n",argc);
    for (i=1; i< argc; i++) {
        printf("    parametro %2d, valore %s\n",i,argv[i]);}
    printf("\n\n");
    exit(0);
}
```

## Letture e stampa di stringhe

```
#include <stdio.h>
void main(int argc, char* argv[]){
    char* stringa[5] = {"", "", "", "", ""};
    int contatore = 0;
    int i = 0;

    printf("Quante parole vuoi inserire? ");
    if (scanf("%i", &contatore)!=1){
        exit(1);
    }
    for (i=0;(i<contatore)&&(i<5) ;i++){
        printf("Inserisci la parola n° %i: ",i+1);
        if (scanf("%s", stringa[i])!=1){
            exit(1);
        }
    }
    for (i=0;(i<contatore)&&(i<5) ;i++){
        printf("Hai inserito: %s\n", stringa[i]);
    }
    exit(0);
}
```