



Topic 1: Introduction to Unix



Operating System

- Software that sits between user(s) + hardware
- Objective
 - Allow users to use computing systems without much hassle
 - Efficient & secure use of the hardware
- Continual Resource Management
 - Central Processing Units (CPUs)
 - Short-term, volatile, main memory
 - Secondary or long term memory including disks, flash memories.
 - I/O devices including Network Interface Cards - NICs
- File System
 - Long-term storage and management of files



Unix OS

- First version by Ken Thompson -- late 60's at Bell Laboratories (mostly assembly).
- Early 70's the system was re-written in C by Brian Kernighan and Dennis Ritchie.
- Many changes since that time:
 - 1. Major revision by UC-Berkeley (in the 80's)
 - 2. Redevelopment for SPARC procs-SUN Microsystems (90's)
 - 3. Release of Linux and ongoing development (90's & 00's)
- Unix/Linux (in various versions/ releases) are now used not only in servers but also in workstations, PCs, tablets, smartphones.
- If you do not have it on your own machine, get it



How to Obtain Linux/Unix

- *Virtualization*: from the OS of your choice, install Linux using a virtualization tool (e.g., *Virtual Box* or *VM Player*)
- *Dual Boot*: chop a portion of your disk (possibly with *gparted*) and install Linux on the second partition (while maintaining your initial installation).
- *The h@ck3r Way*: erase everything you have on your machine and install Linux. Then, proceed to re-install any other OS you may want with Linux using *Virtual Box* or *VM Player*.

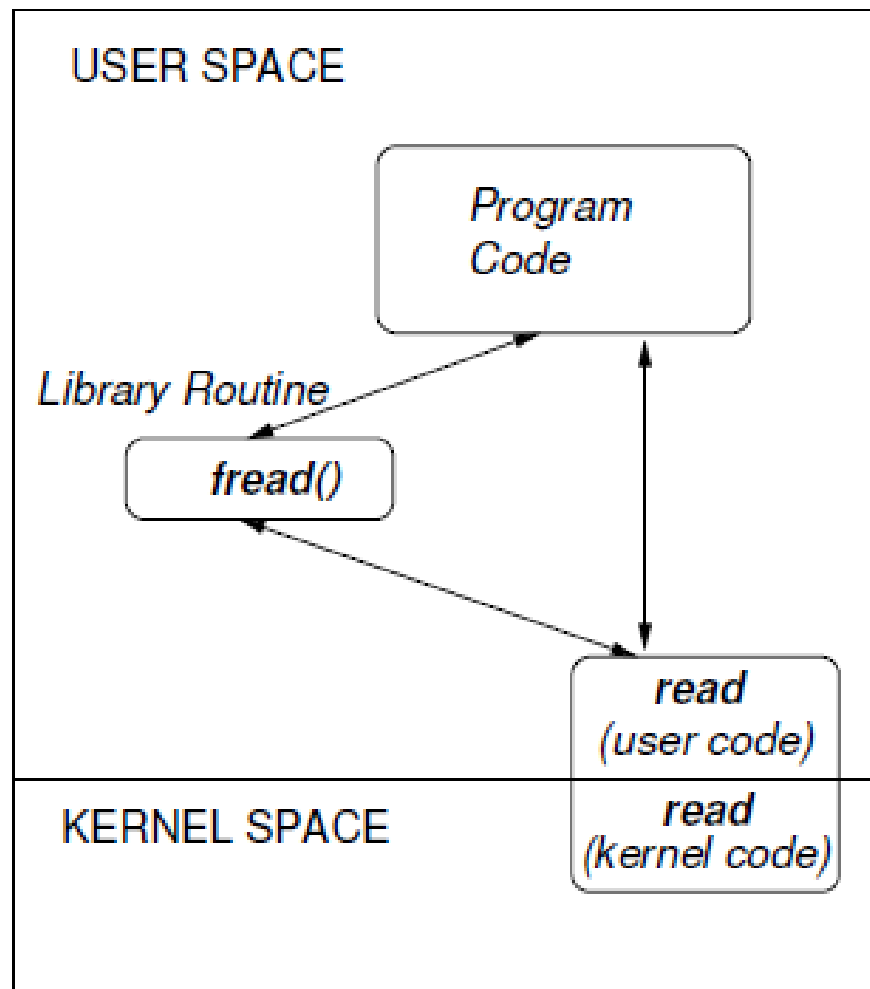


Fundamental Characteristics of Unix

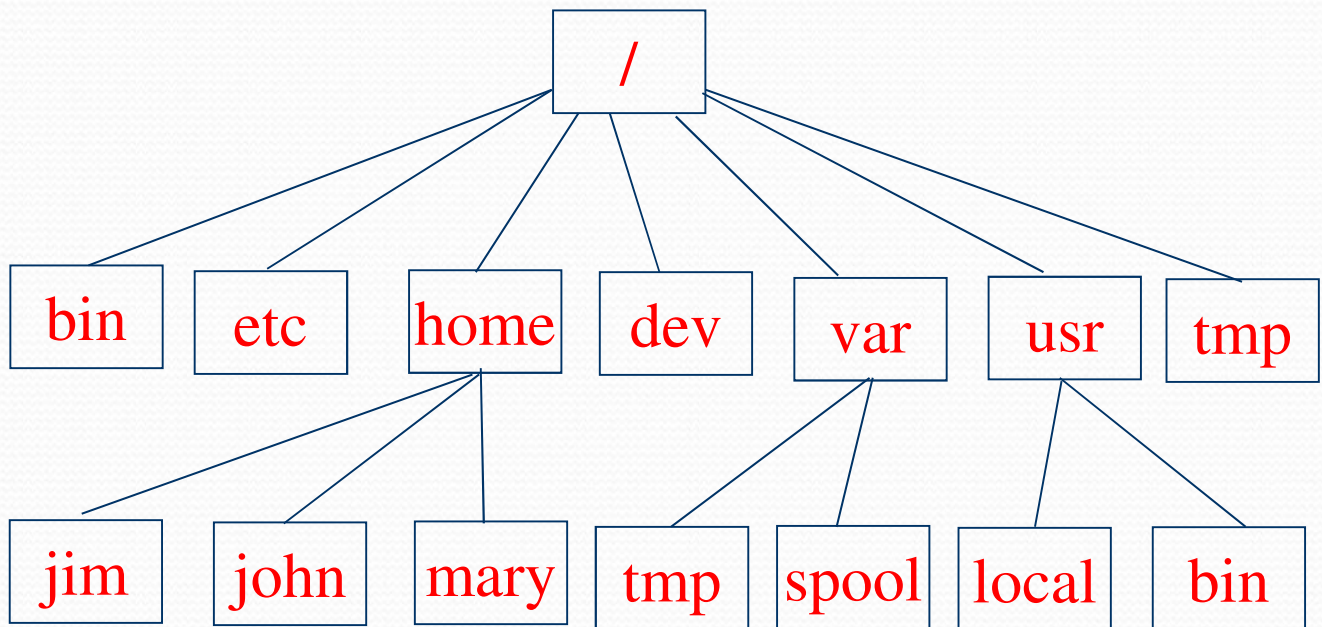
- Supports multiple users simultaneously
- Multi-tasking, concurrent processes, time-sharing
- Security mechanisms
- Networking support
- Monolithic structure:
 - 1. Kernel
 - 2. Shells (c, bash, bourne, korn, t, etc.)
 - 3. System Programs (compilers, editors, command-line tools, etc.)
- System Calls allow interaction with kernel services in terms of C functions.
- Scripting capabilities: one can “program” the shell.

System Calls and Library Routines

- System Calls allow interaction with kernel services in terms of C functions.
- library call:
`nread = fread(inputbuf, OBJSIZE, numberobjs, fileptr);`
- low-level system call:
`nread = read(filedes, inputbuf, BUFSIZE);`



Unix File System - Hierarchical Logical Structure



- ◆ Root directory (/)
- ◆ Current directory (.)
- ◆ Parent directory (..)
- ◆ Absolute and relative paths
- ◆ Files and Links
- ◆ Devices

User Account

- ◆ Username (userid) and password
- ◆ A shell is available as soon as the user is authenticated
- ◆ User groups
- ◆ (Initial) Working directory (aka home directory)

mema@bowser's password:

```
Linux bowser 2.6.24-23-generic #1 SMP Mon Jan 26  
00:13:11 UTC 2009 i686
```

The programs included with the Ubuntu system are free software; the exact distribution terms for each program are described in the individual files in `/usr/share/doc/*/copyright`.

Ubuntu comes with **ABSOLUTELY NO WARRANTY**, to the extent permitted by applicable law.

To access official Ubuntu documentation, please visit:

<http://help.ubuntu.com/>

```
Last login: Mon Jan 7 01:25:54 2008
```

```
mema@bowser:~$ pwd
```

```
/home/mema
```

Basic Unix Commands

man	Display manual page
ls	Listing of the current directory
pwd	Print working directory
cd	Change current directory to
mkdir	Create a new directory
rmdir	Remove an existing directory
cp	Copy a file/directory
mv	Move a file/directory
rm	Remove a file/directory
cat	Display the contents of a file
lpr	Print a file
vi	tty text editor

The “man” command

PWD(1)

User Commands

PWD(1)

NAME

pwd - print name of current/working directory

SYNOPSIS

pwd [OPTION]

DESCRIPTION

Print the full filename of the current working directory.

--help display this help and exit

--version

output version information and exit

NOTE: your shell may have its own version of pwd, which usually supersedes the version described here. Please refer to your shell's documentation for details about the options it supports.

man (-k)

**All lines that contain
the word “manual”**

```
linux02:/home/users/mema>man -k manual
```

```
apropos (1) - search the manual page names and descriptions
catman (8) - create or update the pre-formatted manual pages
cddb-slave2-properties (1) - manual page for Gnome cddb-slave2-j
ecj (1) - manual page for eclipse JDT Compiler
ecj-gcj (1) - manual page for eclipse JDT Compiler
esdcompat (1) - manual page for pulseaudio esd wrapper 0.9.5
gnome-volume-control (1) - manual page for Gnome gnome-volun
grub-reboot (8) - manual page for grub-reboot 0.01
gststreamer-properties (1) - manual page for Gnome gststreamer-prop
man (1) - an interface to the on-line reference manuals
manconv (1) - convert manual page from one encoding to another
mandb (8) - create or update the manual page index caches
manpath (1) - determine search path for manual pages
missing (7) - missing manual pages
pulseaudio (1) - manual page for pulseaudio 0.9.5
readahead-list (8) - manual page for readahead-list: 0.20050517.0
readahead-watch (8) - manual page for readahead-watch: 0.20050517.0
vumeter (1) - manual page for Gnome Volume Meter 2.1.0
w3mman (1) - an interface to the on-line reference manuals 1
whatis (1) - display manual page descriptions
whereis (1) - locate the binary, source, and manual page files
xman (1) - Manual page display program for the X Window
```

Command "ls" (-a, -l, -r)

```
mema@bowser>ls
```

```
direct1 direct2 direct3 quotes
```

```
mema@bowser>ls -a
```

```
. .. .bashrc direct1 direct2 direct3 quotes
```

```
mema@bowser>ls -al
```

```
total 32
```

```
drwxr-xr-x 5 mema dep 4096 Feb 27 16:37 .
```

```
drwxr-xr-x 5 mema dep 4096 Feb 27 14:36 ..
```

```
-rw-r--r-- 1 mema dep 11 Feb 27 16:37 .bashrc
```

```
dr----- 2 mema dep 4096 Feb 27 16:08 direct1
```

```
dr-x----- 2 mema dep 4096 Feb 27 16:08 direct2
```

```
d-wx----- 2 mema dep 4096 Feb 27 16:08 direct3
```

```
-rw-r--r-- 1 mema dep 23 Feb 27 16:09 quotes
```

```
mema@bowser>ls -alr
```

```
total 32
```

```
-rw-r--r-- 1 mema dep 23 Feb 27 16:09 quotes
```

```
d-wx----- 2 mema dep 4096 Feb 27 16:08 direct3
```

```
dr-x----- 2 mema dep 4096 Feb 27 16:08 direct2
```

```
dr----- 2 mema dep 4096 Feb 27 16:08 direct1
```

```
-rw-r--r-- 1 mema dep 11 Feb 27 16:37 .bashrc
```

```
drwxr-xr-x 5 mema dep 4096 Feb 27 14:36 ..
```

```
drwxr-xr-x 5 mema dep 4096 Feb 27 16:37 .
```

```
mema@bowser>
```



When the shell starts

- File that is interpreted when the shell starts:
 - bash: `.bashrc`
 - csh: `.cshrc`
 - tcsh: `.tcshrc`

Access rights

- ◆ Every file/catalog belongs to the user that has created it
- ◆ Every user belongs to at least one group
 - users, postgrads, dep, undergrads...
- ◆ Every file is described with 10 bits
- ◆ - rwx r-x r-x
 user group others
- ◆ 1st bit is either d (directory) or '-' (file)
- ◆ Three groups of bits (read/write/execute)

Permission rules

- The permissions of a file are the first line of defense in the security of a Unix system.
- **File Access Modes**
 - Read: Grants the capability to read, i.e., view the contents of the file.
 - Write: Grants the capability to modify, or remove the contents of the file.
 - Execute: User with execute permissions can run a file as a program.
- **Directory Access Modes**
 - Read: Access to a directory means that the user can read the contents. The user can look at the **filenames** inside the directory.
 - Write: Access means that the user can add or delete files from the directory.
 - Execute: Executing a directory doesn't really make sense, so think of this as a **traverse permission**.**
- A user must have **execute** access to the **bin** directory in order to execute the **ls** or the **cd** command.

Permission rules

	File	Directory
r	Read or Copy a file	Read the contents of the directory
w	Change or delete a file	Add or delete entries (file) in directory
x	Run executable file	Reference or move to (e.g., with “cd”) the directory (without seeing names of other files)

Command “ls” (-d, -R, -t)

```
$ ls -al
total 40
drwx----- 4 spro users 4096 Jan 27 13:20 .
drwxr-xr-x 7 root root 4096 Jan 27 12:10 ..
-rw----- 1 spro users 2232 Jan 27 17:40 .bash_history
-rw-r--r-- 1 spro users 24 Jan 27 12:10 .bash_logout
-rw-r--r-- 1 spro users 191 Jan 27 12:10 .bash_profile
-rw-r--r-- 1 spro users 134 Jan 27 12:24 .bashrc
drwxr-xr-x 2 spro users 4096 Jan 27 12:56 bin
-rw-r--r-- 1 spro users 820 Jan 27 12:10 .emacs
-rw-r--r-- 1 spro users 3511 Jan 27 12:10 .screenrc
$ ls -al bin
total 8
drwxr-xr-x 2 spro users 4096 Jan 27 12:56 .
drwx----- 4 spro users 4096 Jan 27 13:20 ..
$ ls -ald bin ← Μόνο για τον κατάλογο (-d)
drwxr-xr-x 2 spro users 4096 Jan 27 12:56 bin
$ ls -aR . ← Αναδρομική εκτύπωση υποκαταλόγων (-R)
.:
. .bash_history .bash_profile bin .screenrc
.. .bash_logout .bashrc .emacs

./bin:
. ..
$ ls -alt ← Ταξινόμηση με βάση τελευταία μεταβολή (-t)
total 40
-rw----- 1 spro users 2232 Jan 27 17:40 .bash_history
drwx----- 4 spro users 4096 Jan 27 13:20 .
drwxr-xr-x 2 spro users 4096 Jan 27 12:56 bin
-rw-r--r-- 1 spro users 134 Jan 27 12:24 .bashrc
drwxr-xr-x 7 root root 4096 Jan 27 12:10 ..
-rw-r--r-- 1 spro users 24 Jan 27 12:10 .bash_logout
-rw-r--r-- 1 spro users 191 Jan 27 12:10 .bash_profile
-rw-r--r-- 1 spro users 820 Jan 27 12:10 .emacs
-rw-r--r-- 1 spro users 3511 Jan 27 12:10 .screenrc
$
```

Commands: pwd, cd, mkdir, rmdir

```
$ pwd
/home/spro
$ ls -a
.    .bash_history  .bash_profile  bin    .screenrc
..   .bash_logout   .bashrc        .emacs

$ mkdir subdir
$ ls -lF ← Έξτρα σύμβολα στο τέλος ονόματος (-F)
total 8
drwxr-xr-x  2 spro  users  4096 Jan 27 12:56 bin/
drwxr-xr-x  2 spro  users  4096 Jan 27 22:14 subdir/
$ cd subdir
$ ls -al
total 8
drwxr-xr-x  2 spro  users  4096 Jan 27 22:14 .
drwx----- 5 spro  users  4096 Jan 27 22:14 ..
$ pwd ← Μονοπάτι τρέχοντος καταλόγου
/home/spro/subdir
$ cd ..
$ ls -aR
.:
.    .bash_history  .bash_profile  bin    .screenrc
..   .bash_logout   .bashrc        .emacs  subdir

./bin:
.  ..

./subdir:
.  ..
$ rmdir subdir
$ ls subdir
ls: subdir: No such file or directory
$
```

rmdir -r (όχι για καταλόγους)

Εντολές cp (-i, -r), mv (-i), rm (-i, -r, -f)


```
$ ls -a
.  .bash_history  .bash_profile  bin      .screenrc
.. .bash_logout  .bashrc        .emacs
$ cp .bash_profile newfile
$ ls -a
.  .bash_history  .bash_profile  bin      newfile
.. .bash_logout  .bashrc        .emacs  .screenrc
$ mv newfile anotherfile
$ ls -a
.  anotherfile  .bash_logout  .bashrc  .emacs
.. .bash_history  .bash_profile  bin      .screenrc
$ cp -i .bashrc anotherfile ← Αν υπάρχει, επιβεβαίωση (-i)
cp: overwrite 'anotherfile'? n
$ cp /bin/true .
$ mkdir direct
$ cp /bin/false direct
$ mv -i true direct/false ← Αν υπάρχει, επιβεβαίωση (-i)
mv: overwrite 'direct/false'? y
$ cp -r direct directory ← Αναδρομική αντιγραφή (-r)
$ mv anotherfile directory
$ ls -aR
.:
.  .bash_history  .bash_profile  bin      directory  .screenrc
.. .bash_logout  .bashrc        direct   .emacs

./bin:
.  ..

./direct:
.  ..  false

./directory:
.  ..  anotherfile  false
$ rm directory/anotherfile
$ rm -r direct ← Αναδρομική διαγραφή (-r)
$ rm -ri directory
rm: descend into directory 'directory'? y
rm: remove 'directory/false'? y
rm: remove directory 'directory'? y
$ ls -a
.  .bash_history  .bash_profile  bin      .screenrc
.. .bash_logout  .bashrc        .emacs
$
```

rm -f (διαγραφή χωρίς ερώτηση)



Commands: `cp (-i, -r)`, `mv(-i)`, `rm (-i -r, -f)`

- “-i”: interactive mode (asks before an action can be taken)
- “-r” option for directories in `cp` and `rm` commands
- “-f”: force option (in removing files/directories)
- “`rm -f`” is less powerful than “`rm -i`”

Text editors for creating ASCII files

- ◆ `vi/vim filename`: edit the content of the file with name `<filename>`
 - ◆ Two modes of operation: input and edit
 - ◆ Move from input to edit with ESC
 - ◆ Move from edit to input mode by pressing “i”
- ◆ www.yolinux.com/TUTORIALS/LinuxTutorialAdvanced_vi.html
- ◆ emacs, pico, ed, gedit, xemacs, etc
- ◆ Usage instructions in book (Kernighan & Pike) and on Web

Εντολές <code>cat (-n)</code> , <code>lpr (-P)</code>
--

```
$ cat .bashrc  
# .bashrc
```

```
# User specific aliases and functions
```

```
# Source global definitions  
if [ -f /etc/bashrc ]; then  
    . /etc/bashrc  
fi
```

```
PS1='$ '
```

```
$ cat -n .bashrc ← Εμφάνισε και αριθμούς γραμμών (-n)
```

```
 1 # .bashrc  
 2  
 3 # User specific aliases and functions  
 4  
 5 # Source global definitions  
 6 if [ -f /etc/bashrc ]; then  
 7     . /etc/bashrc  
 8 fi  
 9  
10 PS1='$ '
```

```
$ lpr /etc/passwd
```

```
$ lpr -h -Plexmark ../../etc/group
```

```
$
```

Commands for processing ASCII files

<code>chmod</code>	Change access rights for file/directory
<code>more</code>	Page-by-page display of file contents
<code>grep</code>	Search for a specific symbol pattern in a file
<code>wc</code>	Lexical statistics for files
<code>sort</code>	Sort the lines of a text file
<code>touch</code>	Change the timestamp or create a file
<code>ln</code>	Create link
<code>cut</code>	Display portions of lines from a files
<code>cmp</code>	Compare two files
<code>diff</code>	Display differences between two files
<code>head</code>	Display top-k lines of a file
<code>tail</code>	Display bottom-k lines of a file.

Εντολή chmod (-R)

```
mema@browser>cp .cshrc tmpfile
```

```
mema@browser>ls -l
```

```
total 4
```

```
-rwx----- 1 mema dep 751 Feb 17 11:09 tmpfile*
```

```
mema@browser>chmod 0744 tmpfile
```

```
mema@browser>ls -l
```

```
total 4
```

```
-rwxr--r-- 1 mema dep 1751 Feb 17 11:09 tmpfile*
```

```
mema@browser>chmod o-r tmpfile
```

```
mema@browser>chmod g+wx tmpfile
```

```
mema@browser>ls -lt
```

```
total 4
```

```
-rwxrwx--- 1 mema dep 1751 Feb 17 11:09 tmpfile*
```

```
mema@browser>chmod u-wx,g+rx,o+r tmpfile
```

```
mema@browser>ls -l
```

```
total 4
```

```
-r--rwxr-- 1 mema dep 1751 Feb 17 11:09 tmpfile*
```

```
mema@browser> rm tmpfile
```

```
rm: remove write-protected `tmpfile' (yes/no)? Y
```

```
mema@browser> rm tmpfile
```

Αφαίρεσε ανάγνωση από υπόλοιπους

Πρόσθεσε εγγραφή, εκτέλεση σε ομάδα

Εντολή chmod (-R)

```
mema@browser>ls dir1/
```

```
dir1:
```

```
file1.txt file2.txt
```

```
mema@browser>cp -r dir1 dir2
```

```
mema@browser>ls -l dir2
```

```
total 4
```

```
-rw-r--r--  1 mema    dep   16 Feb 17 11:38 file1.txt
```

```
-rw-r--r--  1 mema    dep   16 Feb 17 11:38 file2.txt
```

```
mema@browser>chmod -R 750 dir2
```

Αναδρομικά στον
κατάλογο (-R)

```
mema@browser>ls -lRF dir2
```

```
dir2:
```

```
total 4
```

```
-rwxr-x---  1 mema    dep   16 Feb 17 11:38 file1.txt*
```

```
-rwxr-x---  1 mema    dep   16 Feb 17 11:38 file2.txt*
```

```
mema@browser>
```



Δικαιώματα

Τι δικαιώματα (permissions) απαιτούνται για να μπορεί να διαφραφεί ένα αρχείο;

Τι δικαιώματα απαιτούνται για να μπορεί να εκτελεσθεί επιτυχώς η εντολή “ls dir”;

Χρειάζονται περισσότερα δικαιώματα αν η έντολη είναι η “ls -l dir”;

```
mema@browser>ls -l
total 16
drw----- 2 mema dep 4096 Feb 27 16:08 direct1
dr-x----- 2 mema dep 4096 Feb 27 16:08 direct2
d-wx----- 2 mema dep 4096 Feb 27 16:08 direct3
-rw-r--r-- 1 mema dep  23 Feb 27 16:09 quotes
mema@browser>cp quotes quotes.noRead
mema@browser>chmod 0300 quotes.noRead
mema@browser>cp quotes quotes.noWrite
mema@browser>chmod 0500 quotes.noWrite
mema@browser>cp quotes quotes.noExecute
mema@browser>chmod 0600 quotes.noExecute
mema@browser>ls -lt
total 28
-rw----- 1 mema dep 23 Feb 27 16:10 quotes.noExecute
-r-x----- 1 mema dep  23 Feb 27 16:10 quotes.noWrite
--wx----- 1 mema dep 23 Feb 27 16:10 quotes.noRead
-rw-r--r-- 1 mema dep  23 Feb 27 16:09 quotes
d-wx----- 2 mema dep 4096 Feb 27 16:08 direct3
dr-x----- 2 mema dep 4096 Feb 27 16:08 direct2
drw----- 2 mema dep 4096 Feb 27 16:08 direct1
mema@browser>rm quotes.noExecute
mema@browser>rm quotes.noRead
```

Όταν έχω wx permission
στον κατάλογο, μπορώ.
Αλλιώς όχι.

```
mema@browser>rm quotes.noWrite
rm: remove write-protected regular file 'quotes.noWrite'? n
mema@browser>ls -l
total 20
drwx----- 2 mema dep 4096 Feb 27 16:08 direct1
dr-x----- 2 mema dep 4096 Feb 27 16:08 direct2
d-wx----- 2 mema dep 4096 Feb 27 16:08 direct3
-rw-r--r-- 1 mema dep  23 Feb 27 16:09 quotes
-r-x----- 1 mema dep  23 Feb 27 16:10 quotes.noWrite
mema@browser>ls direct1
mema@browser>ls -l direct1
total 0
mema@browser>ls direct2
mema@browser>ls -l direct2
total 0
mema@browser>cd direct3
mema@browser>ls -l direct3
ls: cannot open directory 'direct3': Permission denied
mema@browser>
```

Δεν έχει δικαίωμα ανάγνωσης

Command “more”

```
$ more /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
news:x:9:13:news:/var/spool/news:
uucp:x:10:14:uucp:/var/spool/uucp:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
gopher:x:13:30:gopher:/var/gopher:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
mailnull:x:47:47:/:/var/spool/mqueue:/dev/null
rpm:x:37:37:/:/var/lib/rpm:/bin/bash
xfs:x:43:43:X Font Server:/etc/X11/fs:/bin/false
ntp:x:38:38:/:etc/ntp:/sbin/nologin
rpc:x:32:32:Portmapper RPC user:/:/bin/false
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
--More--(74%)
```

- Δυνατότητες

[Return] : Εμφάνιση μίας επιπλέον γραμμής

[Space] : Εμφάνιση μίας επιπλέον σελίδας

b : Εμφάνιση προηγούμενης σελίδας

/*<s>* [Return] : Αναζήτηση προς τα εμπρός της
συμβολοσειράς *<s>*

!*<c>* [Return] : Εκτέλεση της εντολής *<c>*

v : Κλήση του κειμενογράφου vi

. : Επανάληψη της προηγούμενης εντολής

h : Περιγραφή των δυνατοτήτων της more

q : Έξοδος από τη more

Command “less”

- Similar to more:
- [return] – skip a line
- [space] – get next page
- b – display previous page
- /<s> -- search for pattern s
- !<c>[return] -- execute command c
- h – help
- q – exit from less
- Used to offer more than more, hence “less is more”... 😊 now, on most systems, is the same binary

Εντολές grep (-n, -i, -v), wc (-l, -w, -c)

```
$ grep 6000 /etc/login.defs ← Γραμμές που περιέχουν 6000
UID_MAX                60000
GID_MAX                60000
$ grep -n rc .bashrc ← Τύπωσε # γραμμής
1:# .bashrc
5:# Source global definitions
6:if [ -f /etc/bashrc ]; then
7:    . /etc/bashrc
$ grep -i RoOt /etc/passwd ← Αγνόησε κεφαλαία-μικρά
root:x:0:0:root:/root:/bin/bash
operator:x:11:0:operator:/root:/sbin/nologin
$ grep -v '#' /etc/hosts.allow ←
ALL: 195.134.65.
ALL: 195.134.66.
ALL: 195.134.67.
ALL: 195.134.68.
ALL: 195.134.69.
# Όλες τις γραμμές ΕΚΤΟΣ
# από όσες ταιριάζουν
$ wc .bash_profile ← # Γραμμές, λέξεις, χαρακτήρες
  13      29      191 .bash_profile
$ wc -l .bash_profile ← Μόνο # γραμμών
  13 .bash_profile
$ wc -w .bash_profile ← Μόνο # λέξεων
  29 .bash_profile
$ wc -c .bash_profile ← Μόνο # χαρακτήρων
 191 .bash_profile
$ wc -cl .bash_profile
  13      191 .bash_profile
$ ls -al .bash_profile
-rw-r--r--  1 spro  users          191 Jan 27 12:10 .bash_pr
$
```

Εντολή sort

```
$ cat example.txt
```

```
john      32      london
jean       7       paris
marco     21       rome
anna      18       athens
antonio   58       madrid
peter     5       chicago
ahmet     27       ankara
luis      1       geneva
ji        41       peking
filip     18       brussels
```

```
$ sort example.txt
```

```
ahmet     27       ankara
anna      18       athens
antonio   58       madrid
filip     18       brussels
jean       7       paris
ji        41       peking
john      32       london
luis      1       geneva
marco     21       rome
peter     5       chicago
```

← Ταξινόμηση (ανά γραμμή, σαν κείμενο)

```
$ sort +1 example.txt
```

```
anna      18       athens
filip     18       brussels
luis      1       geneva
marco     21       rome
ahmet     27       ankara
john      32       london
ji        41       peking
antonio   58       madrid
peter     5       chicago
jean       7       paris
```

← Με βάση 2η στήλη, σαν κείμενο

>sort -k 2 example.txt

```
$ sort -r -n +1 example.txt
```

```
antonio   58       madrid
ji        41       peking
john      32       london
ahmet     27       ankara
marco     21       rome
filip     18       brussels
anna      18       athens
jean       7       paris
peter     5       chicago
```

← Με βάση 2η στήλη, Αριθμητική (-n), Αντίστροφη (-r)

>sort -r -n -k 2 example.txt

Εντολές touch, ln (-s)

```
mema@browser>touch empty.txt
```

```
mema@browser>ls -la  
total 4
```

Αν υπάρχει ήδη, άλλαξε χρόνο τροποποίησης. Αλλιώς δημιούργησε

```
drwxr-xr-x 2 mema dep 512 Feb 19 11:22 ./  
drwxr-xr-x 4 mema dep 512 Feb 17 11:38 ../  
-rw-r--r-- 1 mema dep  0 Feb 19 11:22 empty.txt  
-rw-r--r-- 1 mema dep 16 Feb 17 11:30 file1.txt  
-rw-r--r-- 1 mema dep 18 Feb 17 11:30 file2.txt
```

```
mema@browser>ln file1.txt hard-link
```

Σκληρός σύνδεσμος

```
mema@browser>ls -la  
total 5
```

```
drwxr-xr-x 2 mema dep 512 Feb 19 11:25 ./  
drwxr-xr-x 4 mema dep 512 Feb 17 11:38 ../  
-rw-r--r-- 1 mema dep  0 Feb 19 11:22 empty.txt  
-rw-r--r-- 2 mema dep 16 Feb 17 11:30 file1.txt  
-rw-r--r-- 1 mema dep 18 Feb 17 11:30 file2.txt  
-rw-r--r-- 2 mema dep 16 Feb 17 11:30 hard-link
```

```
mema@browser>ln -s /home/mema/rev.txt symbolic-link
```

Συμβολικός Σύνδεσμος (-s)



```
mema@browser>ls -la
```

```
total 6
```

```
drwxr-xr-x 2 mema dep 512 Feb 19 11:26 ./
```

```
drwxr-xr-x 4 mema dep 512 Feb 17 11:38 ../
```

```
-rw-r--r-- 1 mema dep  0 Feb 19 11:22 empty.txt
```

```
-rw-r--r-- 2 mema dep 16 Feb 17 11:30 file1.txt
```

```
-rw-r--r-- 1 mema dep 18 Feb 17 11:30 file2.txt
```

```
-rw-r--r-- 2 mema dep 16 Feb 17 11:30 hard-link
```

```
lrwxrwxrwx 1 mema dep 24 Feb 19 11:26 symbolic-link ->  
/home/mema/rev.txt
```

```
mema@browser>ls -F
```

```
empty.txt file1.txt file2.txt hard-link symbolic-link@
```

```
mema@browser>rm hard-link symbolic-link
```

↖
Δε σβήνει ούτε το file1.txt
ούτε το /home/mema/rev.txt

Εντολή ln (συνέχεια)

```
mema@browser>ls -l
total 2
-rw-r--r-- 1 mema  dep   16 Feb 17 11:30 file2.txt
mema@browser>ln file2.txt hard-link
mema@browser>ls -l
total 4
-rw-r--r-- 2 mema  dep   16 Feb 17 11:30 file2.txt
-rw-r--r-- 2 mema  dep   16 Feb 17 11:30 hard-link
mema@browser>ls -li
1266325 file2.txt 1266325 hard-link ← Ίδιο inode
mema@browser>rm file2.txt
mema@browser>ls -l
total 2
-rw-r--r-- 1 mema  dep   16 Feb 17 11:30 hard-link
```

Σκληροί και Συμβολικοί Σύνδεσμοι

Σκληροί	Συμβολικοί
Δείκτης σε κοινό αρχείο (newNam, sameInode)	Αντιγράφει μονοπάτι αρχείου MONO
Δεν εφαρμόζεται σε καταλόγους	Εφαρμόζεται σε καταλόγους
Μετονομασία αρχικού αρχείου δε δημιουργεί πρόβλημα	Μετονομασία αρχικού αρχείου “σπάει” το σύνδεσμο
Αλλαγές σε σύνδεσμο ή αρχικό αρχείο επηρεάζουν και τα 2	Αλλαγές σε σύνδεσμο ή αρχικό αρχείο επηρεάζουν και τα 2
Αρχείο σβήνεται όταν ΟΛΟΙ οι δείκτες σε αυτό σβηστούν	Διαγραφή ΔΕΝ επηρεάζει αρχικό αρχείο



Σκληροί και Συμβολικοί Σύνδεσμοι

Πότε μπορούμε να δημιουργήσουμε ένα αρχείο/σκληρό σύνδεσμο σε ένα αρχείο που ανήκει σε άλλον ιδιοκτήτη;

Το αρχείο/σκληρός σύνδεσμος που φτιάξαμε, ποιον έχει ιδιοκτήτη;

Ποτε μπορεί ο ιδιοκτήτης να σβήσει το σκληρό σύνδεσμο που δημιουργήσαμε;

Αν ο ιδιοκτήτης σβήσει το αρχικό αρχείο τότε σβήνει και ο σκληρός σύνδεσμος;

Παράδειγμα

1.

```
kronos:/home/users/spro/popo>ls -l
total 8
-rwx----- 1 spro    other    10 Oct 15 18:04 file700*
-rwx-wx-wx  1 spro    other    10 Oct 15 18:04 file733*
-rwxr--r--  1 spro    other    10 Oct 15 18:04 file744*
-rwxr-xr-x  1 spro    other    10 Oct 15 18:04 file755*
```
4.

```
kronos:/home/users/spro/popo>cd ~adeli/popo
kronos:/home/users/adeli/popo>rm my700
rm: my700 not removed: Permission denied
kronos:/home/users/adeli/popo>rm my733
rm: my733 not removed: Permission denied
kronos:/home/users/adeli/popo>rm my744
rm: my744 not removed: Permission denied
kronos:/home/users/adeli/popo>rm my755
rm: my755 not removed: Permission denied
```
7.

```
kronos:/home/users/adeli/popo>rm my700
kronos:/home/users/adeli/popo>rm my733
kronos:/home/users/adeli/popo>rm my744
kronos:/home/users/adeli/popo>rm my755
kronos:/home/users/adeli/popo>ls -l
total 0
kronos:/home/users/adeli/popo>cd ~spro/popo
kronos:/home/users/spro/popo>ls -l
total 8
-rwx----- 1 spro    other    10 Oct 15 18:04 file700*
-rwx-wx-wx  1 spro    other    10 Oct 15 18:04 file733*
-rwxr--r--  1 spro    other    10 Oct 15 18:04 file744*
-rwxr-xr-x  1 spro    other    10 Oct 15 18:04 file755*
kronos:/home/users/spro/popo>
```

Ιδιοκτήτης,
αλλά όχι 'w'
στο κατάλογο

Ο adeli (επόμενη σελ.) φτιάχνει συνδέσμους σε αρχεία του spro (παραπάνω). Πρέπει να σβήσουν όλοι οι δείκτες σε 1 αρχείο για να διαγραφεί αυτό. Όλα καλά αν ο adeli έχει 'x' στον ~spro/popo και 'w' στον δικό του. Ο spro διαγράφει με 'wx' στο ~adeli/popo.

Παράδειγμα

2. kronos:/home/users/adeli/popo>ls -l Όλα OK
- ```
total 0
kronos:/home/users/adeli/popo>ln ~spro/popo/file700 my700
kronos:/home/users/adeli/popo>ln ~spro/popo/file733 my733
kronos:/home/users/adeli/popo>ln ~spro/popo/file744 my744
kronos:/home/users/adeli/popo>ln ~spro/popo/file755 my755
```
3. kronos:/home/users/adeli/popo>ls -l
- ```
total 8
-rwx----- 2 spro other 10 Oct 15 18:04 my700*
-rwx-wx-wx 2 spro other 10 Oct 15 18:04 my733*
-rwxr--r-- 2 spro other 10 Oct 15 18:04 my744*
-rwxr-xr-x 2 spro other 10 Oct 15 18:04 my755*
```
5. kronos:/home/users/adeli/popo>ls -al
- ```
total 12
drwxr-xr-x 2 adeli other 512 Oct 15 2007 ./
drwxr-xr-x 17 adeli other 1024 Oct 15 18:05 ../
-rwx----- 2 spro other 10 Oct 15 18:04 my700*
-rwx-wx-wx 2 spro other 10 Oct 15 18:04 my733*
-rwxr--r-- 2 spro other 10 Oct 15 18:04 my744*
-rwxr-xr-x 2 spro other 10 Oct 15 18:04 my755*
```
6. kronos:/home/users/adeli/popo>chmod 0777 .  
kronos:/home/users/adeli/popo>

Αν ένας χρήστης ΔΕΝ έχει δικαίωμα εγγραφής σε έναν κατάλογο, δεν μπορεί να σβήσει ένα αρχείο μέσα στον κατάλογο ακόμα και αν είναι ο ιδιοκτήτης.

# Έντολή cut (-f, -d)

```
[hermes 40] cat koko2
12:john:2107271111:maria
13:jim:2107221400:panagiwta
14:antonio:2107277777:32
[hermes 41] cut -f1,3 -d':' koko2
12:2107271111
13:2107221400
14:2107277777
[hermes 42] cut -f2,4 -d':' koko2
john:maria
jim:panagiwta
antonio:32
[hermes 43] █
```

-f (επιλογή στηλών που θέλουμε)

-d (διαχωριστικό στηλών – αν δεν οριστεί είναι το tab ‘\t’)

Αν δυνατότητες της cut δεν είναι επαρκείς, τότε γλώσσες awk, perl, python, ruby...

## Εντολές cmp, diff, head (-n), tail (-n)

```
$ cp .bashrc .bashrc.dupl
$ cmp .bashrc .bashrc.dupl
$ rm .bashrc.dupl
$ cmp example1.txt example2.txt
example1.txt example2.txt differ: char 140, line 5
$ diff example1.txt example2.txt
```

← Αν διέφεραν, θα τύπωνε  
1ο σημείο διαφοράς

```
5c5
< antonio 58 madrid

```

```
> antonio 68 madrid
```

← Χωρίς όρισμα, πρώτες 10 γραμμές

```
$ head /etc/group
root:x:0:root
bin:x:1:root,bin,daemon
daemon:x:2:root,bin,daemon
sys:x:3:root,bin,adm
adm:x:4:root,adm,daemon
tty:x:5:
disk:x:6:root
lp:x:7:daemon,lp
mem:x:8:
kmem:x:9:
```

← Πρώτες 2 γραμμές (-n)

```
$ head -2 /etc/group
root:x:0:root
bin:x:1:root,bin,daemon
```

```
$ tail /etc/group
mailnull:x:47:
rpm:x:37:
xfs:x:43:
ntp:x:38:
rpc:x:32:
rpcuser:x:29:
nfsnobody:x:65534:
nscd:x:28:
ident:x:98:
radvd:x:75:
$ tail -1 /etc/group
radvd:x:75:
$
```

# Λοιπές Εντολές: echo (-n), date, passwd, hostname, whoami, lpq (-P), lprm (-P)

```
$ echo This is a message produced by echo ← Με αλλαγή γραμμής
This is a message produced by echo
$ echo -n This is a message produced by echo ← Χωρίς αλλαγή
This is a message produced by echo$ γραμμής
$ date
Sat Feb 2 21:58:13 EET 2002
$ passwd
Changing password for spro
(current) UNIX password:
New password:
Retype new password:
passwd: all authentication tokens updated successfully
$ hostname ← Όνομα μηχανήματος
galini
$ whoami
spro
$ lpr /etc/sendmail.cf
$ lpq ← Κατάσταση ουράς εκτυπωτή
lp is ready and printing
Rank Owner Job Files Total Size
active spro 143 /etc/sendmail.cf 46365 bytes
$ lprm 143 ← Διαγραφή εργασίας εκτύπωσης
dfA143galini dequeued
cfA143galini dequeued
$ lpr -Plp2 /etc/termcap
$ lpr -Plp2 /usr/share/magic
$ lpq -Plp2
lp2 is ready and printing
Rank Owner Job Files Total Size
active spro 144 /etc/termcap 737535 bytes
1st spro 145 /usr/share/magic 226045 bytes
$ lprm -Plp2 spro
dfA144galini dequeued
cfA144galini dequeued
dfA145galini dequeued
cfA145galini dequeued
$
```



# Κέλυφος (Shell)

- Program just like “date” or “who” that does some extra special stuff
- Sits between user and kernel
- Benefits:
  - Ανακατεύθυνση εισόδου-εξόδου
  - Συντομογραφικά ονόματα αρχείων με μεταχαρακτήρες (pattern matching of filenames)
  - Δυνατότητα προσωπικής προσαρμογής του περιβάλλοντος

# Κέλυφος (csh, tcsh, bash, kornsh, etc.)

- Το κέλυφος αρχικής σύνδεσης στο Linux είναι το bash
- Η εντολή logout αποσυνδέει το χρήστη
- Με csh (tcsh) δημιουργείται νεό κέλυφος C
- Η εντολή exit ή control-d τερματίζει ένα κέλυφος C
- Το ~ συμβολίζει τον κατάλογο αφετηρίας
- Κατά την ενεργοποίηση ενός κελύφους C εκτελείται το ~/.cshrc
- Κατά την αρχική σύνδεση εκτελείται το ~/.login
- Κατά την διακοπή της σύνδεσης εκτελείται το ~/.logout

# Κέλυφος (συνέχεια)

- Προκαθορισμένη είσοδος stdin (πληκτρολόγιο)
- Προκαθορισμένη έξοδος stdout (οθόνη)
- Προκαθορισμένη έξοδος διαγνωστικών σφαλμάτων stderr (οθόνη)
- Περιγραφείς αρχείων για αυτά είναι 0,1,2
  
- Για ορισμένες εντολές (cat, lpr, grep, wc, sort, head, tail κτλ) τα ορίσματα των αρχείων είναι προαιρετικά
  - Αν παραλειφθούν χρησιμοποιείται το stdin

```
% wc
```

```
This is an example where the command "wc" is used without filename argument. So, it counts the lines, words and characters of the standard input.
```

```
^D ← Τέλος εισόδου για εντολή
4 25 146
```

```
%
```

# Ανακατευθύνσεις (csh)

- Ανακατεύθυνση του stdin (<)

```
% grep reg < .screenrc ← Είσοδος από το .screenrc
Prepend/append register [/] to the paste if ^a^] is pressed.
register ["\033:se noai\015a"
register] "\033:se ai\015a"
%
```

- Ανακατεύθυνση των stdout και stderr (>, >!, >&, >&!)

```
% date > a_file ← Ημερομηνία στο a_file
% cat < a_file
Sun Feb 3 13:54:48 EET 2002
% ls -a > a_file ← Το > δεν γράφει σε υπάρχον αρχείο σε
a_file: File exists.
% echo A test line >! a_file ← κάποια UNIX συστήματα
% cat < a_file
A test line
% cp
cp: missing file arguments
Try 'cp --help' for more information.
% cp >& b_file ← Ανακατεύθυνση stdout και stderr
% cat < b_file
cp: missing file arguments
Try 'cp --help' for more information.
% cat c_file >& b_file
b_file: File exists.
% cat c_file >! b_file ← Γράψε πάνω στο αρχείο
% cat b_file
cat: c_file: No such file or directory
%
```

**ΠΡΟΣΟΧΗ:** Για να προστατέψετε τα υπάρχοντα αρχεία σε συστήματα όπου η συμπεριφορά είναι διαφορετική, γράψτε: `set noclobber` για να μην γράφει σε υπάρχοντα αρχεία.

# Ανακατευθύνσεις με προσάρτηση

```
% cat a_file
A test line
% cat b_file
cat: c_file: No such file or directory
% date >> c_file
c_file: No such file or directory.
% date >> a_file
% cat a_file
A test line
Sun Feb 3 13:59:33 EET 2002
% wc < .bashrc >>! c_file
% cat c_file
 8 21 124
% lls >>& b_file
% cat b_file
cat: c_file: No such file or directory
lls: Command not found.
% cat b_file c_file e_file >>&! d_file
% cat d_file
cat: c_file: No such file or directory
lls: Command not found.
 8 21 124
cat: e_file: No such file or directory
% rm a_file b_file c_file d_file
%
```

Γράψε έξοδο του date στο ΤΕΛΟΣ του c\_file

Γράψε με έμφαση στο c\_file τα στατιστικά του .bashrc

Προσάρτηση stdout και stderr

- Φίλτρα και σωληνώσεις (|)

```
% ls -al | grep 6
total 36
drwx----- 3 sprocsh users 4096 Feb 3 14:01 .
drwxr-xr-x 8 root root 4096 Feb 2 09:15 ..
drwxr-xr-x 3 sprocsh users 4096 Feb 2 09:15 .kde
% cat .bashrc | tail -5 | sort | lpr -h -Plp2
%
```

Τύπωσε ταξινομημένες τις τελευταίες 5 γραμμές του .bashrc

# Ανακατευθύνσεις

- 1) Για ανακατεύθυνση stdout and stderr σε διαφορετικά αρχεία:

```
(command > stdout_file) >& stderr_file
```

- 2) Για ανακατεύθυνση μόνο stderr

```
sh -c 'command 2> stderr_file'
```

```
mema@browser> more foo
foo: No such file or directory
mema@browser> more bar
hey
mema@browser> (cat foo bar > outFile) >& errFile
mema@browser> more outFile
hey
mema@browser> more errFile
cat: foo: No such file or directory
mema@browser> sh -c 'cat foo bar 2> errFile2'
hey
mema@browser> more errFile2
cat: foo: No such file or directory
```

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

```
#include <unistd.h>
```

```
int main(){
```

```
 int i=0;
```

```
 while(1) {
```

```
 for(i=0; i<3; i++) {
```

```
 fprintf(stdout,"Guillermo ");
```

```
 write(1,"Guillermo ",10);
```

```
 fflush(stdout);
```

```
 }
```

```
 fprintf(stdout,"!\n");
```

```
 // write to stderr U r a smart fella
```

```
 write(2,"U r a smart fella\n",20);
```

```
 sleep(1);
```

```
 }
```

```
}
```

```
mema@browser>a.out
```

```
Guillermo Guillermo Guillermo Guillermo Guillermo C
```

```
U r a smart fella
```

```
Guillermo Guillermo Guillermo Guillermo Guillermo C
```

```
U r a smart fella
```

```
^C
```

```
mema@browser>
```

## Send stdout to file out.log

---

```
mema@browser> a.out > out.log
```

```
U r a smart fella
```

```
U r a smart fella
```

```
U r a smart fella
```

```
^C
```

```
mema@browser> cat out.log
```

```
Guillermo Guillermo Guillermo Guillermo Guillermo C
```

```
Guillermo Guillermo Guillermo Guillermo Guillermo C
```

```
Guillermo Guillermo Guillermo Guillermo Guillermo C
```

```
Guillermo Guillermo Guillermo Guillermo Guillermo
```

---

```
mema@browser> a.out 1> out.log
```

```
U r a smart fella
```

```
U r a smart fella
```

```
U r a smart fella
```

```
^C
```

```
mema@browser> cat out.log
```

```
Guillermo Guillermo Guillermo Guillermo Guillermo C
```

```
Guillermo Guillermo Guillermo Guillermo Guillermo C
```

```
Guillermo Guillermo Guillermo Guillermo Guillermo C
```

```
Guillermo Guillermo Guillermo Guillermo Guillermo
```

# Send stdout to file out.log and stderr to err.log

---

```
mema@browser> ./a.out 1> out.log 2> err.log
```

```
^C
```

```
mema@browser>cat out.log
```

```
Guillermo Guillermo Guillermo Guillermo Guillermo C
```

```
Guillermo Guillermo Guillermo Guillermo Guillermo C
```

```
Guillermo Guillermo Guillermo Guillermo Guillermo
```

```
mema@browser>cat err.log
```

```
U r a smart fella
```

```
U r a smart fella
```

```
U r a smart fella
```

```
mema@browser>
```

---

# Send stdout and stderr to file allout (3 different ways)

---

```
mema@browser> ./a.out >& allout
```

```
^C
```

```
mema@browser> cat allout
```

```
Guillermo Guillermo Guillermo Guillermo Guillermo Guillermo U r a sm:
Guillermo !
```

```
Guillermo Guillermo Guillermo Guillermo Guillermo U r a smart f
```

```
mema@browser> ./a.out &> allout
```

```
^C
```

```
mema@browser>cat allout
```

```
Guillermo Guillermo Guillermo Guillermo Guillermo Guillermo U r a sm:
Guillermo !
```

```
Guillermo Guillermo Guillermo Guillermo Guillermo U r a smart fella
```

```
mema@browser> ./a.out > allout 2>&1
```

```
^C
```

```
mema@browser>cat allout
```

```
Guillermo Guillermo Guillermo Guillermo Guillermo Guillermo U r a sm:
Guillermo !
```

```
Guillermo Guillermo Guillermo Guillermo Guillermo U r a smart fella
Guillermo !
```

```
Guillermo Guillermo Guillermo Guillermo Guillermo U r a smart fella
```

```
mema@browser>
```

---



# The Beauty of Unix

- With pipes, we avoid the use of temporary files
- By following the “contract” of stdin and stdout, programs are simplified

# Ακολουθίες και Ομάδες Εντολών (;)

```
% pwd
/home/sprocsh
% mkdir mydir ; cd mydir ; ls -al
total 8
drwxr-xr-x 2 sprocsh users 4096 Feb 3 14:04 .
drwx----- 4 sprocsh users 4096 Feb 3 14:04 ..
% pwd
/home/sprocsh/mydir
% cd ..
% rmdir mydir
% (mkdir mydir ; cd mydir ; ls -al) ← Με παρένθεση
total 8
drwxr-xr-x 2 sprocsh users 4096 Feb 3 14:05 .
drwx----- 4 sprocsh users 4096 Feb 3 14:05 ..
% pwd ← Με παρένθεση παραμένουμε στον ίδιο
/home/sprocsh
% rmdir mydir κατάλογο που δώσαμε τις εντολές
%
```

## Εντολή umask

- ΠΡΟΣΟΧΗ: Ο αριθμός της umask δείχνει ποια δικαιώματα ΔΕΝ θέλουμε να δίνουμε
- Συγκρίνετε με chmod

```
% umask
22 ← 022: Όχι εγγραφή σε ομάδα, υπόλοιπους
% touch file1
% mkdir dir1
% ls -ld file1 dir1
drwxr-xr-x 2 sprocsh users 4096 Feb 3 14:06 dir1
-rw-r--r-- 1 sprocsh users 0 Feb 3 14:06 file1
% umask 077 ← 077: Όλα κομμένα σε ομάδα, υπόλοιπους
% touch file2
% mkdir dir2
% ls -ld file2 dir2
drwx----- 2 sprocsh users 4096 Feb 3 14:07 dir2
-rw----- 1 sprocsh users 0 Feb 3 14:07 file2
% umask 022
% rm file1 file2
% rmdir dir1 dir2
%
```

# Μεταχαρακτήρες

- Μεταχαρακτήρες για ονόματα αρχείων
  - \* : Ταιριάζει με κανένα ή περισσότερους χαρακτήρες
  - ? : Ταιριάζει με έναν ακριβώς χαρακτήρα
  - [<c><sub>1</sub><c><sub>2</sub>...<c><sub>n</sub>] : Ταιριάζει με έναν ακριβώς χαρακτήρα από τους <c><sub>1</sub>, <c><sub>2</sub>, ..., <c><sub>n</sub>
  - [<c><sub>1</sub> - <c><sub>2</sub>] : Ταιριάζει με έναν ακριβώς χαρακτήρα μεταξύ των <c><sub>1</sub> και <c><sub>2</sub> συμπεριλαμβανομένων

```
.....
% ls -dF ???
gtk/ kde/ nmh/ ntp/ opt/ ppp/ rmt@ rpc rpm/ ssh/ X11/
% cp *tab* ~
% ls -dF [bmpwz][acjp]*
bashrc mail.rc pam.d/ passwd warnquota.conf
mail/ makedev.d/ pam_smb.conf passwd-
mailcap l_____config paper.config ppp/
% ls -ldF *[a-e][f-i][k-q][r-z]*
drwxr-xr-x 2 root root 4096 Dec 27 14:35 cron.daily/
% cd ~
% ls
anacrontab crontab fstab fstab.REVOKE inittab mtab
% rm -i *tab*
rm: remove 'anacrontab'? y
rm: remove 'crontab'? y
rm: remove 'fstab'? y
rm: remove 'fstab.REVOKE'? y
rm: remove 'inittab'? y
rm: remove 'mtab'? y
%
```

# Μεταβλητές Περιβάλλοντος και Μεταβλητές κελύφους

- `setenv` displays/sets all environment (i.e., global) variables usable by any program.
  - Αν θέλουμε όλα τα προγράμματα που τρέχουμε από μέσα από το κέλυφος να έχουν πρόσβαση σε κάποιες μεταβλητές τότε τις καθορίζουμε με `setenv`.
- `set` displays/sets properties specific to the shell
  - Αν θέλουμε να ορίσουμε ιδιότητες που θέλουμε να έχει το κέλυφος καθώς εκτελεί τις διαταγές που του δίνουμε
  - e.g., `prompt`, `path`, etc.

# Μεταβλητές Περιβάλλοντος στο csh (setenv, unsetenv)

```
% setenv | tail -5 ← Τελευταίες 5 ορισμένες μεταβλητές
SSH_ASKPASS=/usr/libexec/openssh/gnome-ssh-askpass
KDEDIR=/usr
LANG=en_US
SUPPORTED=en_US:en:el_GR:el
LESSOPEN=|/usr/bin/lesspipe.sh %s
% setenv MYVAR value_of_myvar
% setenv | tail -6
SSH_ASKPASS=/usr/libexec/openssh/gnome-ssh-askpass
KDEDIR=/usr
LANG=en_US
SUPPORTED=en_US:en:el_GR:el
LESSOPEN=|/usr/bin/lesspipe.sh %s
MYVAR=value_of_myvar
% echo $MYVAR
value_of_myvar
% unsetenv MYVAR ← Ακύρωση εντολής
% setenv | tail -5
SSH_ASKPASS=/usr/libexec/openssh/gnome-ssh-askpass
KDEDIR=/usr
LANG=en_US
SUPPORTED=en_US:en:el_GR:el
LESSOPEN=|/usr/bin/lesspipe.sh %s
% echo $MYVAR
MYVAR: Undefined variable.
%
```

# Μεταβλητές κελύφους csh (set, unset)

- Τοπικές μεταβλητές κελύφους C (set, unset)

term : Ορίζει τον τύπο του τερματικού

path : Ορίζει τους καταλόγους στους οποίους το σύστημα ψάχνει για εκτελέσιμα αρχεία

prompt : Ορίζει τη συμβολοσειρά που εκτυπώνει το κέλυφος C όταν περιμένει είσοδο από το χρήστη

noclobber : Έλεγχος ανακατευθύνσεων σε υπάρχοντα αρχεία

ignoreeof : Έλεγχος χρήσης ^D για τερματισμό κελύφους

```
% set | grep '^[i-t]'
```

loginsh ← Μεταβλητές που ξεκινάνε από [i-t]

```
noclobber
owd /etc
path (/usr/local/bin /bin /usr/bin /usr/X11R6/bin)
prompt %
prompt2 %R?
prompt3 CORRECT>%R (y|n|e|a)?
shell /bin/tcsh
shlvl 1
sourced 1
status 0
tcsh 6.10.00
term xterm
tty pts/0
% set term=vt100
% set path=($path /etc)
% set prompt="hostname '/whoami' --> "
galini/sprocsh --> set prompt="% "
% touch afile
% ls -a > afile
afile: File exists.
% unset noclobber
% ls -a > afile
% rm afile
% set ignoreeof ← Αγνόησε το ^D ως logout
% ^D
Use "logout" to logout.
%
```

# Ιστορία εντολών

- Μηχανισμός ιστορίας
  - Τοπική μεταβλητή `history` κελύφους `C`
  - Εντολή `history`

```
% set history=5
% ls -a .*bash*
.bash_logout .bash_profile .bashrc
% date
Sun Feb 3 23:00:00 EET 2002
% wc .bashrc
 8 21 124 .bashrc
% echo testing...
testing...
% history
 2 22:59 ls -a .*bash*
 3 23:00 date
 4 23:00 wc .bashrc
 5 23:00 echo testing...
 6 23:00 history

% !5
echo testing...
testing...
% !da
date
Sun Feb 3 23:00:35 EET 2002
% hostname
galini
% !!
hostname
galini
```

## Ψευδώνυμα Εντολών (alias, unalias)

```
% alias dir ls ← Για όταν ξεχνιέμαι (windows)
% alias rm 'rm -i' ← Πάντα ερώτηση σε διαγραφή
% cp .bash_logout logout_file
% dir
logout_file
% rm logout_file
rm: remove 'logout_file'? y
% alias llf 'ls -alF'
% llf
total 36
drwx----- 3 sprocsh users 4096 Feb 3 23:03 ./
drwxr-xr-x 8 root root 4096 Feb 2 09:15 ../
-rw-r--r-- 1 sprocsh users 24 Feb 2 09:15 .bash_logout
-rw-r--r-- 1 sprocsh users 191 Feb 2 09:15 .bash_profile
-rw-r--r-- 1 sprocsh users 124 Feb 2 09:15 .bashrc
-rw-r--r-- 1 sprocsh users 30 Feb 3 13:54 .cshrc
-rw-r--r-- 1 sprocsh users 820 Feb 2 09:15 .emacs
drwxr-xr-x 3 sprocsh users 4096 Feb 2 09:15 .kde/
-rw-r--r-- 1 sprocsh users 3511 Feb 2 09:15 .screenrc
% alias pp 'lpr -h -Plp2'
% pp .bash_profile
% alias cd 'cd \!* ; echo $cwd'
% cd /usr/lib
/usr/lib
% cd
/home/sprocsh
% alias
cd cd \!* ; echo $cwd
dir ls
l. ls -d .[a-zA-Z]* --color=tty
ll ls -l --color=tty
llf ls -alF
ls ls --color=tty
pp lpr -h -Plp2
rm rm -i
% unalias rm ← Ακύρωση ψευδώνυμου
% unalias cd
%
```

## Setting/unsetting variables in bash

```
mema@browser> set | head -4
BASH =/bin/bash
BASH_ALIASES =()
BASH_ARGC =()
BASH_ARGV =()
mema@browser> MYVAR =" foobar "
mema@browser> echo MYVAR
MYVAR
mema@browser> echo $MYVAR
foobar
mema@browser> set | grep MYVAR
MYVAR = foobar
mema@browser> unset MYVAR
mema@browser> echo $MYVAR
mema@browser>
mema@browser> set | grep MYVAR
mema@browser>
```

# Process Management

|                   |                                                              |
|-------------------|--------------------------------------------------------------|
| <code>ps</code>   | Display status of processes on system                        |
| <code>jobs</code> | Display interrupted processes                                |
| <code>kill</code> | Kill a specific process                                      |
| <code>bg</code>   | Make a process run in the “background”                       |
| <code>fg</code>   | Make a process run in the “foreground”                       |
| <code>top</code>  | Show statistics on processor, memory, running programs, etc. |

# Εντολές ps (-u), jobs, kill (-9)

```
$ ps
 PID TTY TIME CMD
 1052 pts/0 00:00:00 bash
 1093 pts/0 00:00:00 ps
$ ps -u
USER PID %CPU %MEM VSZ RSS TTY STAT START TIME COMMAND
sprog 1052 0.7 2.1 2488 1344 pts/0 S 21:08 0:00 -bash
sprog 1094 0.0 1.1 2600 716 pts/0 R 21:09 0:00 ps -u
$ cp -r /usr/share/icons .
^Z
[1]+ Stopped cp -r /usr/share/icons .
$ ps -u
USER PID %CPU %MEM VSZ RSS TTY STAT START TIME COMMAND
sprog 1052 0.4 2.1 2488 1348 pts/0 S 21:08 0:00 -bash
sprog 1095 1.8 0.9 1696 604 pts/0 T 21:09 0:00 cp -r
 /usr/share/
sprog 1096 0.0 1.1 2600 716 pts/0 R 21:09 0:00 ps -u
$ jobs
[1]+ Stopped cp -r /usr/share/icons .
$ kill %1

[1]+ Stopped cp -r /usr/share/icons .
$
[1]+ Terminated cp -r /usr/share/icons .
$ cp -r /usr/lib . &
[1] 1097
$ ps
 PID TTY TIME CMD
 1052 pts/0 00:00:00 bash
 1097 pts/0 00:00:00 cp
 1098 pts/0 00:00:00 ps
$ kill -9 1097
$
[1]+ Killed cp -r /usr/lib .
$ ps
 PID TTY TIME CMD
 1052 pts/0 00:00:00 bash
 1099 pts/0 00:00:00 ps
$ rm -r icons lib
$
```

## Εντολές bg, fg

Ξεκινά σταματημένη διεργασία  
στο παρασκήνιο

```
$ cp -r /usr/share .
```

```
^Z
```

```
[1]+ Stopped
```

```
cp -r /usr/share .
```

```
$ bg
```

```
[1]+ cp -r /usr/share . &
```

```
$ jobs
```

```
[1]+ Running
```

```
cp -r /usr/share . &
```

```
$ ps
```

| PID  | TTY   | TIME     | CMD  |
|------|-------|----------|------|
| 1052 | pts/0 | 00:00:00 | bash |
| 1102 | pts/0 | 00:00:01 | cp   |
| 1103 | pts/0 | 00:00:00 | ps   |

Ξεκινά σταματημένη διεργασία  
στο προσκήνιο ή φέρνει μια  
ενεργή από το παρασκήνιο  
στο προσκήνιο

```
$ fg
```

```
cp -r /usr/share .
```

```
^Z
```

```
[1]+ Stopped
```

```
cp -r /usr/share .
```

```
$ bg
```

```
[1]+ cp -r /usr/share . &
```

```
$ ps
```

| PID  | TTY   | TIME     | CMD  |
|------|-------|----------|------|
| 1052 | pts/0 | 00:00:00 | bash |
| 1102 | pts/0 | 00:00:01 | cp   |
| 1104 | pts/0 | 00:00:00 | ps   |

```
$ kill 1102
```

```
$
```

```
[1]+ Terminated
```

```
cp -r /usr/share .
```

```
$ ps
```

| PID  | TTY   | TIME     | CMD  |
|------|-------|----------|------|
| 1052 | pts/0 | 00:00:00 | bash |
| 1105 | pts/0 | 00:00:00 | ps   |

```
$ chmod -R 755 share
```

```
^C
```

# Dealing with multiple jobs in the background

```
mema@browser>sleep 100 & sleep 200 & sleep 300 & sleep 400 &
```

```
[1] 7238
```

```
[2] 7239
```

```
[3] 7240
```

```
[4] 7241
```

```
mema@browser> jobs
```

```
[1] Running sleep 100 &
```

```
[2] Running sleep 200 &
```

```
[3] - Running sleep 300 &
```

```
[4]+ Running sleep 400 &
```

```
mema@browser> fg
```

```
sleep 400
```

```
^C
```

```
mema@browser> jobs
```

```
[1] Running sleep 100 &
```

```
[2] - Running sleep 200 &
```

```
[3]+ Running sleep 300 &
```

```
mema@browser> fg
```

```
sleep 300
```

```
^C
```

```
mema@browser> jobs
```

```
[1] - Running sleep 100 &
```

```
[2]+ Running sleep 200 &
```

```
mema@browser> fg
```

```
sleep 200
```

```
^C
```

```
mema@browser> jobs
```

```
[1]+ Running sleep 100 &
```

```
mema@browser> fg
```

```
sleep 100
```

```
^C
```

```
mema@browser> jobs
```

```
mema@browser>
```

Note the plus/minus signs

# top Command

load averages: 0.03, 0.04, 0.04 18:00

31 processes: 30 sleeping, 1 on cpu

CPU states: 92.1% idle, 3.8% user, 1.2% kernel, 3.0% iowait, 0.0% swap

Memory: 128M real, 46M free, 12M swap in use, 244M swap free

| PID  | USERNAME | THR | PRI | NICE | SIZE  | RES   | STATE | TIME   | CPU   | COMMAND       |
|------|----------|-----|-----|------|-------|-------|-------|--------|-------|---------------|
| 1153 | root     | 1   | -3  | 0    | 2832K | 1776K | sleep | 0:10   | 0.84% | sendmail893   |
| 6376 | root     | 1   | 33  | 0    | 17M   | 1568K | sleep | 40.8H  | 0.36% | screensaver   |
| 1212 | adeli    | 1   | 33  | 0    | 1800K | 1104K | cpu   | 0:00   | 0.31% | top           |
| 166  | root     | 1   | 33  | 0    | 2352K | 1368K | sleep | 23:21  | 0.00% | amd           |
| 1180 | root     | 1   | -15 | 0    | 2984K | 1936K | sleep | 0:18   | 0.00% | sendmail893   |
| 358  | root     | 1   | 33  | 0    | 5448K | 744K  | sleep | 425:47 | 0.00% | sshd-1.2.26   |
| 374  | root     | 1   | 33  | 0    | 2416K | 944K  | sleep | 16:19  | 0.00% | sendmail893   |
| 112  | root     | 1   | 33  | 0    | 2080K | 912K  | sleep | 8:58   | 0.00% | rpcbind       |
| 1    | root     | 1   | 33  | 0    | 688K  | 200K  | sleep | 8:22   | 0.00% | init          |
| 213  | root     | 1   | 23  | 0    | 1584K | 672K  | sleep | 5:29   | 0.00% | cron          |
| 197  | root     | 1   | 33  | 0    | 2000K | 1176K | sleep | 2:56   | 0.00% | inetd         |
| 1080 | root     | 15  | 7   | 0    | 4832K | 1952K | sleep | 0:30   | 0.00% | syslogd       |
| 206  | root     | 1   | 34  | -12  | 2280K | 688K  | sleep | 0:19   | 0.00% | xntpd         |
| 154  | root     | 1   | 33  | 0    | 5592K | 1384K | sleep | 0:01   | 0.00% | sshd-1.2.26   |
| 168  | adeli    | 1   | 4   | 10   | 1424K | 352K  | sleep | 0:00   | 0.00% | spinner       |
| 157  | adeli    | 1   | 23  | 0    | 2120K | 1608K | sleep | 0:00   | 0.00% | tcsh          |
| 6375 | root     | 1   | 23  | 0    | 920K  | 328K  | sleep | 0:00   | 0.00% | screensave.sh |

# Συμπίεση, αποσυμπίεση αρχείων

- ◆ Πολλά προγράμματα (tar, zip, gzip, gunzip, zcat, uuencode, uudecode...)

gzip compr.tar

gunzip compr.tar.gz

- ◆ Συμπίεση των αρχείων Makefile \*.c \*.h και του καταλόγου experiments στο αρχείο compr.tar:

```
tar cvf compr.tar Makefile *.c *.h experiments
```

- ◆ Αποσυμπίεση με:

```
tar xvf compr.tar
```

- ◆ Σε .gz χρησιμοποιήστε gunzip
- ◆ Σε .tgz χρησιμοποιήστε gunzip και μετά tar xvf
- ◆ Σε .uue χρησιμοποιήστε uudecode

# Other commands

- tr** Translate or delete characters in file
- colrm** Delete columns from file
- find** Locate files/directories in the file system
- last** Last appearances of user
- script** Record an interaction with the shell
- sleep** Wait for a number of seconds
- clear** Clear tty

# Εντολές tr (-d), colrm

```
$ cat /etc/resolv.conf
nameserver 195.134.65.119
search di.uoa.gr
domain di.uoa.gr
$ tr 56789abc 012345xyz < /etc/resolv.conf
n5meserver 140.134.10.114
se5ryh di.uo5.gr
dom5in di.uo5.gr
```

Τύπωσε στην οθόνη το `/etc/resolv.conf` αντικαθιστώντας τους χαρακτήρες του `56789abc` με τους αντίστοιχους από το `012345xyz`

```
$ tr -d '[0-9]uoa' < /etc/resolv.conf
```

```
nmeserver ...
serch di..gr
dmin di..gr
```

Όλα εκτός από τους χαρακτήρες...

```
$ colrm 8 14 < /etc/resolv.conf
```

```
nameser.134.65.119
search gr
domain gr
```

Όλους τους χαρακτήρες σε στήλες 8-14

```
$ colrm 15 < /etc/resolv.conf
```

```
nameserver 195
search di.uoa.
domain di.uoa.
```

Όλους τους χαρακτήρες σε στήλες  $\geq 15$

```
$ ls -ald bin
```

```
drwxr-xr-x 2 spro users 4096 Jan 27 12:56 bin
```

```
$ ls -ald bin | colrm 11 55
```

```
drwxr-xr-x bin
```

```
$
```

**tr:** *translate*, squeeze, and/or delete characters from standard input, writing to standard output

**colrm:** *remove* columns from standard input, writing to standard output

# Εντολή find (-name, -exec)

```
kronos:/home/users/spro>find . -name '*.og*' -exec cat {} . \;
@(#)login 1.7 89/09/05 SMI
setenv TERM `tset -Q -`
cat: input error on .: Is a directory
kronos:/home/users/spro>find . -name '*ht*' -print
./public_html
./public_html/index.html
./public_html/index.htm
./public_html/spethics.htm
kronos:/home/users/spro>find . -name '*.og*' -exec cat {} . \;
@(#)login 1.7 89/09/05 SMI
setenv TERM `tset -Q -`
cat: input error on .: Is a directory
kronos:/home/users/spro>find . -name '*.og*' -print -exec cat {} . \;
./login
@(#)login 1.7 89/09/05 SMI
setenv TERM `tset -Q -`
cat: input error on .: Is a directory
kronos:/home/users/spro>
```

↑ Τύπωσε περιεχόμενα από τα αρχεία (-exec την cat εντολή)

↑ Τύπωσε πλήρες όνομα αρχείων (-print) και τα περιεχόμενά τους

```

$ last -n 8
spro pts/O ppp-116.dialup.u Wed Feb 6 21:28 still logged in
reboot system boot 2.4.7-10 Wed Feb 6 16:19 (05:33)
spro pts/O knossos.di.uoa.g Wed Feb 6 12:24-12:27 (00:03)
spro pts/O ppp-117.dialup.u Wed Feb 6 10:48-11:03 (00:14)
spro pts/O ppp-100.dialup.u Wed Feb 6 09:21-10:20 (00:59)
spro pts/O ppp-50.dialup.uo Tue Feb 5 21:08-22:00 (00:52)
reboot system boot 2.4.7-10 Tue Feb 5 16:29 (22:19)
sprocsh pts/1 ppp-146.dialup.u Mon Feb 4 22:33-22:49 (00:16)

wtmp begins Fri Feb 1 12:27:24 2002
$ script my_session ←
Script started, file is my_session
$ date
Wed Feb 6 21:56:29 EET 2002
$ whoami
spro
$ exit
exit
Script done, file is my_session
$ cat my_session
Script started on Wed Feb 6 21:56:25 2002
$ date
Wed Feb 6 21:56:29 EET 2002
$ whoami
spro
$ exit ←
exit

Script done on Wed Feb 6 21:56:37 2002
$ rm my_session
$ date ; sleep 20 ; date
Wed Feb 6 21:57:24 EET 2002
Wed Feb 6 21:57:44 EET 2002
$ clear

```

Κατέγραψε τα πάντα από  
εδώ και πέρα στο  
my\_session

Επίσης με ^D, τέλος καταγραφής

## Άλλα Χρήσιμα...

- ◆ Ηλεκτρονικό ταχυδρομείο
  - Προγράμματα mail, mutt, pine, xmail
- ◆ Μεταγλωτιστές της C/C++: gcc/g++

```
mema@browser> ls
InputFile LinkedList.c main.c MyHeader.h OrderedLinkedList.h
InputFile.out LinkedList.h Makefile OrderedLinkedList.c
mema@browser>
mema@browser> gcc -c main.c
mema@browser> gcc -c LinkedList.c
mema@browser> gcc -c OrderedLinkedList.c
mema@browser> gcc -o myexe main.o LinkedList.o OrderedLinkedList.o
mema@browser> ./myexe
expected syntax: myexe inputfile
```



# C/C++ Compilation

- To compile programs in C/C++ use `gcc/g++`
  - for separate compilation: `-c` flag
  - every reference needs to be to a *declared* variable/function, but it may be *defined* in a different “compilation unit”
  - compilation unit: `.c/.cpp` file with all others it includes
  - for functions: declaration has signature, no body
  - for variables: declaration uses “extern”
  - C header files (`.h`) should contain only function declarations, get included wherever these functions are referenced

# Makefiles

- ◆ Οργάνωση μεταγλώττισης αρχείων
- ◆ Στη μορφή:

Στόχος: Απαιτούμενα

\t Εντολή

- ◆ Προσοχή στο tab. ΠΡΕΠΕΙ ΝΑ ΤΟ ΒΑΖΕΤΕ.
- ◆ Τα Απαιτούμενα πρέπει να υλοποιηθούν πριν το Στόχο.
- ◆ Αν τα Απαιτούμενα δεν έχουν αλλάξει, δεν εκτελείται η εντολή
- ◆ Συχνά ο Στόχος του εκτελέσιμου προγράμματος έχει το όνομα all
- ◆ Συχνά ο Στόχος clean διαγράφει τα παραγόμενα εκτελέσιμα και \*.o

## Παράδειγμα Makefile

```

In order to execute this "Makefile" just type "make"
A. Delis (ad@di.uoa.gr)
#
```

```
OBJS = main.o LinkedList.o OrderedLinkedList.o
SOURCE = main.c LinkedList.c OrderedLinkedList.c
HEADER = MyHeader.h LinkedList.h OrderedLinkedList.h
OUT = myexe
CC = gcc
FLAGS = -g -c
-g option enables debugging mode
-c flag generates object code for separate files
```

```
all: $(OBJS)
 $(CC) -g $(OBJS) -o $(OUT)
```

```
create/compile the individual files >>separately<<
```

```
main.o: main.c
 $(CC) $(FLAGS) main.c
```

```
LinkedList.o: LinkedList.c
 $(CC) $(FLAGS) LinkedList.c
```

```
OrderedLinkedList.o: OrderedLinkedList.c
 $(CC) $(FLAGS) OrderedLinkedList.c
```

```
clean house
```

```
clean:
 rm -f $(OBJS) $(OUT)
```

```
do a bit of accounting
```

```
count:
 wc $(SOURCE) $(HEADER)
```

# Παράδειγμα Makefile

**OBJS = main.o iodat.o dorun.o**

**LIB = /usr/lib/crtn.o**

**CC = /usr/bin/gcc**

**program: \${OBJ} \${LIB} mylibops.a**  
**\${CC} -o \$@ \${OBJS} \${LIB} mylibops.a**

**main.o: main.c**  
**\${CC} -c \$?**

**iodat.o: iodat.c**  
**\${CC} -c \$?**

**dorun.o: dorun.c**  
**\${CC} -c \$?**

**mylibops.a: interact.o sched.o gen.o**  
**ar r \$@ \$?**

**interact.o: interact.c**  
**\${CC} -c \$?**

**gen.o: gen.c**  
**\${CC} -c \$?**

**sched.o: sched.c**  
**\${CC} -c \$?**

**\$? : current list of prerequisites**  
**\$@: current target**