



Διάλεξη 2 Επανάληψη Εντολών Unix για Χρήστες

Δημήτρης Ζεϊναλιπούρ



Περιεχόμενο Διάλεξης

- **UNIX εντολές:** *passwd, man, whatis, apropos, info, help, users, who, whoami, finger, w,*
- **Ρεύματα Εισόδου/Εξόδου** (*stdin, stdout, stderr*), Ανακατεύθυνση Αρχείων (*Redirection*), Διοχέτευση/Σωλήνωση (*Piping*).
- **Εξέταση Αρχείων** (*file, head, tail, cat, more, wc, sort, uniq*), Χαρακτήρες Τερματισμού Γραμμής (*unix, win, mac*)
- **Κανονικές Εκφράσεις** (*grep, egrep*)
- **Σημείωση:** Αναζητήστε ΚΑΘΕ εντολή στο *man manual*.



Λογαριασμός Χρήστη

- Όνομα χρήστη
- Συνθηματικό
 - *passwd* για αλλαγή συνθηματικού

```
www2n.cs.ucy.ac.cy - PuTTY
1) dzeina@www2n> passwd
Changing password for user dzeina.
Enter login(LDAP) password: █
```

- Κέλυφος αρχικής σύνδεσης
 - `$ bash --version`
GNU bash, version 3.1.17(1)-release (i686-redhat-linux-gnu)
Copyright (C) 2005 Free Software Foundation, Inc.
- Κατάλογος αφετηρίας
 - π.χ., `/home/faculty/dzeina`



Πού να βρούμε βοήθεια

- Εντολή *man* (επιλογές *-k*, *-f*, *-a*)
 - Εμφάνιση **οδηγιών χρήσης εντολών**
 - Π.χ., `$man ls`
 - Παρουσιάζει επιλογές χρήσης της εντολής `ls`
- **Επισημάνσεις**
 - Η πιο σημαντική εντολή στο UNIX
 - Σε όλα τα UNIX (εκτός Debian) παράγονται `man pages` για εντολές του συστήματος.
 - Οδηγίες χρήσης διαθέσιμες και στο Διαδίκτυο (π.χ., http://en.wikipedia.org/wiki/List_of_Unix_utilities)



Πού να βρούμε βοήθεια

bash-3.1\$ **man pwd**

Τ' όνομα του προγράμματος και μια σύντομη περιγραφή

Πώς να τρέξουμε αυτή την εντολή

Λίστα όλων των επιλογών και τι κάνουν

```
cs4038.in.cs.ucy.ac.cy - PuTTY
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.
AUTHOR
  Written by Jim Meyering.
REPORTING BUGS
  Report bugs to <bug-coreutils@gnu.org>.
COPYRIGHT
  Copyright © 2006 Free Software Foundation, Inc.
  This is free software. You may redistribute copies of it under the terms of the
  GNU General Public License <http://www.gnu.org/licenses/gpl.html>. There is NO
  WARRANTY, to the extent permitted by law.
SEE ALSO
  The full documentation for pwd is maintained as a Texinfo manual. If the info
  and pwd programs are properly installed at your site, the command
  info pwd
  should give you access to the complete manual.
pwd 5.97                                August 2006                               PWD (1)
~
bash-3.1$
```



Πού να βρούμε βοήθεια

– επιλογή: **man -f** <εντολή>

- Επιστρέφει την σύντομη περιγραφή της εντολής.

Π.Χ.,

`man -f ls`

`ls`

`ls`

Μέρη του Man Manual

(1) - list directory contents

(1p) - list directory contents

Posix standard

- Στη συνέχεια “`man 1p ls`” για να δούμε την εμφάνιση
- Ίσοδύναμο: **whatis** <λέξη_κλειδί>

– επιλογή: **man -k** < λέξη_κλειδί >

- Επιστρέφει όλες τις εντολές, βιβλιοθήκες, κτλ στις οποίες εμφανίζεται η λέξη κλειδί.

- Ισοδύναμο: **apropos** <λέξη_κλειδί>



Πού να βρούμε βοήθεια

bash-3.1\$ **man -f pwd**

```
cs4038.in.cs.ucy.ac.cy - PuTTY
bash-3.1$ man -f pwd
pwd          (1)  - print name of current/working directory
pwd          (1p) - return working directory name
pwd [builtins] (1)  - bash built-in commands, see bash(1)
pwd.h [pwd]   (Op) - password structure
bash-3.1$ █
```

bash-3.1\$ **man -k pwd**

```
cs4038.in.cs.ucy.ac.cy - PuTTY
bash-3.1$ man -k pwd
pwd          (1)  - print name of current/working directory
pwd          (1p) - return working directory name
pwd [builtins] (1)  - bash built-in commands, see bash(1)
pwd.h [pwd]   (Op) - password structure
pwdx         (1)  - report current working directory of a process
unix_chkpwd  (8)  - helper binary that verifies the password of the current user
bash-3.1$ █
```



Πού να βρούμε βοήθεια

- Το *man* (εγχειρίδιο) χωρίζεται γενικά σε οκτώ αριθμημένα τμήματα (sections)
 - κάθε σελίδα-οδηγίες χρήσης έχει τ' όνομα της εντολής ακολουθούμενο από τον αριθμό τμήματος σε παρένθεση.

Τμήμα	Περιγραφή
0	C library headers
1	User Commands
2	System Calls
3	C library routines
4	Administrative files
5	Miscellaneous
6	Games
7	I/O and special files
8	System administration commands



Πού να βρούμε βοήθεια

- Υπο-τμήματα:
 - Π.χ., 1p (Portable Operating System Interface (POSIX)-based user command)

Υπο-τμήμα	Περιγραφή
p	POSIX specifications
x	X Window System documentation

- Παράδειγμα

bash-3.1\$ **man 3 printf (C Library Routine)**

bash-3.1\$ **man 1 printf (User Command)**



Πού να βρούμε βοήθεια

- Εντολή *info*

- Εναλλακτικό εγχειρίδιο χρήσης εντολών συστήματος.
- Πιο ολοκληρωμένη περιγραφή με χρήση υπερσυνδέσμων.
 - Παράγεται από το **GNU project (GNU is Not Unix)** (Λ.Σ. = χωρίς πυρήνα - που αποτελείται αποκλειστικά από ανοικτού πηγαίου κώδικα)

- Χρήση επιλογής *--help*

- χρησιμοποιείται σε συγκεκριμένες εντολές κελύφους
- παράδειγμα: `bash-3.1$ ps --help`

```
www2n.cs.ucy.ac.cy - PuTTY
3)dzeina@www2n> ps --help
ps --help
***** simple selection ***** ***** selection by list *****
-A all processes                    -C by command name
-N negate selection                  -G by real group ID (supports nam
-a all w/ tty except session leaders -U by real user ID (supports name
-d all except session leaders        -g by session OR by effective gro
-e all processes                      -p by process ID
```

Ποιοί είναι συνδεδεμένοι (logged on) και τι κάνουν;



- Εντολή *users*
 - Λίστα όλων των χρηστών που είναι συνδεδεμένοι στο σύστημα

```
cs4038.in.cs.ucy.ac.cy - PuTTY
bash-3.1$ users
cchrys cspgcc1 totis
bash-3.1$
```

- Εντολή *who*
 - Λίστα όλων των χρηστών που είναι συνδεδεμένοι στο σύστημα και από πού έχουν συνδεθεί και από ποια ώρα

```
cs4038.in.cs.ucy.ac.cy - PuTTY
bash-3.1$ who
totis pts/1 2007-01-23 08:24 (cs1746.cs.ucy.ac.cy)
cspgcc1 pts/2 2007-01-23 14:40 (cs1534.cs.ucy.ac.cy)
cchrys pts/3 2007-01-23 14:37 (cs1534.cs.ucy.ac.cy)
bash-3.1$
```

Ποιοί είναι συνδεδεμένοι (logged on) και τι κάνουν;



- Εντολή *whoami*
 - Εμφάνιση ονόματος χρήστη

```
dzeina@xyz ~  
$ whoami  
dzeina
```

- Εντολή *finger <user_name>*
 - Παρουσιάζει συνοπτικές πληροφορίες σχετικά με το χρήστη
 - Πρόκειται για υπηρεσία που τρέχει στο port 79.

```
www2n.cs.ucy.ac.cy - PuTTY  
1)dzeina@www2n> finger dzeina  
Login: dzeina Name: Demetris Zeinalipour  
Directory: /home/faculty/dzeina Shell: /bin/bash  
On since Tue Jan 19 16:03 (EET) on pts/1 from xyz.in.cs.ucy.ac.cy  
No mail.  
Project: _____ → ~/.project  
No project  
Plan: _____ → ~/.plan  
No plan  
1)dzeina@www2n>
```

Ποιοί είναι συνδεδεμένοι (logged on) και τι κάνουν;



- Εντολή **w**

- Λίστα όλων των χρηστών που είναι συνδεδεμένοι στο σύστημα και τι τρέχουν

Total time by all processes

uptime Last 1 min, 5min, 15min

```
cs4038.in.cs.ucy.ac.cy - PuTTY
bash-3.1$ w
14:57:39 up 1 day, 10:44, 3 users, load average: 1.14, 1.07, 1.01
USER      TTY      FROM             LOGIN@   IDLE   JCPU   PCPU   WHAT
totis     pts/1    cs1746.cs.ucy.ac 08:24    1:58m  1:59m  53.93s top
cspgcc1   pts/2    cs1534.cs.ucy.ac 14:40    0.00s  0.21s  0.01s w
cchrys    pts/3    cs1534.cs.ucy.ac 14:37    27.00s 0.18s  0.18s -ksh
bash-3.1$
```

time by current process

- Είναι shortcut του συνδυασμού εντολών:

- who (show who is logged on)
- uptime (Tell how long the system has been running)
- ps -a (list all processes with a terminal).

Default Editor - select-editor



- **select-editor:** provides a coherent mechanism for selecting and storing a preferred sensible-editor on a per-user basis.
 - `SELECTED_EDITOR` is overridden by the `VISUAL` and `EDITOR` environment variables.

- **Example:**

Select an editor. To change later, run 'select-editor'.

1. `/bin/nano` <---- easiest
2. `/usr/bin/vim.tiny`
3. `/bin/ed`

Other:

- `select-default-ispell`
- `select-default-wordlist`
- `select-default-iwrap`



Ρεύματα Εισόδου/Εξόδου

- Όταν εκτελέσουμε ένα πρόγραμμα ανοίγουν τρία προσωπικά αρχεία για αυτό το αρχείο τα οποία είναι ως ακολούθως:
- ***stdin***
 - Είναι το προκαθορισμένο ρεύμα για **παραλαβή εισόδου**, όταν ένα πρόγραμμα τρέχει.
 - default: είσοδος από το πληκτρολόγιο
- ***stdout***
 - Είναι το προκαθορισμένο ρεύμα για **αποστολή εξόδου**, όταν ένα πρόγραμμα τρέχει.
 - default: έξοδος στην οθόνη
- ***stderr***
 - Είναι το προκαθορισμένο ρεύμα για αποστολή εξόδου **διαγνωστικών μηνυμάτων**, όταν ένα πρόγραμμα τρέχει.
 - default έξοδος – μηνύματα λάθους στην οθόνη

Ανακατεύθυνση Αρχείων



- Χρήση ειδικών τελεστών του UNIX για διάβασμα (*read*), γράψιμο (*write*) και επισύναψη (*append*) δεδομένων από/σε αρχείο.
 - < **ανακατεύθυνση εισόδου** από αρχείο
το αρχείο πρέπει να υπάρχει.
 - > **ανακατεύθυνση εξόδου** σε αρχείο
αν δεν υπάρχει το αρχείο δημιουργείται,
αλλιώς γράφεται πάνω από το υπάρχων.
 - >> **ανακατεύθυνση εξόδου και
επισύναψη** των δεδομένων στο
υπάρχων αρχείο (*append*)
αν δεν υπάρχει το αρχείο δημιουργείται,
αλλιώς συνεχίζει στο υπάρχων.

Ανακατεύθυνση Αρχείων



```
bash-3.1$ ls test/ /bin/DoesNotExist
ls: /bin/DoesNotExist: No such file or directory
test/:
ErrorOutput HardLinkToFile1.txt
  SymbLinkToFile2.txt test1 test2
bash-3.1$ ls test/ /bin/DoesNotExist 2>
  test/ErrorOutput
test/:
ErrorOutput HardLinkToFile1.txt
  SymbLinkToFile2.txt test1 test2
bash-3.1$ cat test/ErrorOutput
ls: /bin/DoesNotExist: No such file or directory

./foo < InputFile 1>> OutputFile 2>> ErrorFile
```



Διοχέτευση/Σωλήνωση

- **Διοχέτευση (*pipe*)** επιτρέπει τη σύνδεση της εξόδου ενός προγράμματος στην είσοδο ενός άλλου.

– Επιτυγχάνεται με τη χρήση του ειδικού συμβόλου |

– *Παραδείγματα:*

```
bash-3.1$ ls test
```

```
HardLinkToFile1.txt
```

```
SymbLinkToFile2.txt
```

```
test1
```

```
test2
```

```
bash-3.1$ ls test | wc
```

```
4
```

```
4
```

```
52
```

Εκθέτει τον αριθμό των γραμμών, λέξεων και χαρακτήρων των περιεχομένων του καταλόγου *test*



Διοχέτευση/Σωλήνωση

```
bash-3.1$ cat test-uniq.test
```

Dan

Dan

**Παράδειγμα ταξινόμησης αρχείου και μετά
αφαίρεσης των διπλότυπων**

Dan

Athena

Ryu

Ryu

Athena

Nick

```
bash-3.1$ sort test-uniq.test | uniq
```

Athena

Dan

Nick

Ryu

```
bash-3.1$ sort test-uniq.test | uniq -c
```

```
  2 Athena
```

```
  3 Dan
```

```
  1 Nick
```

```
  2 Ryu
```

Ανακατεύθυνση και Διοχέτευση



- Μπορούμε να έχουμε ταυτόχρονα ανακατεύθυνση αρχείων και διοχέτευση.

```
bash-3.1$ sort < test-uniq.test 2> ErrorOutput | uniq 1> Output
```

```
bash-3.1$ cat ErrorOutput
```

```
bash-3.1$ cat Output
```

Athena

Dan

Nick

Ryu

Προσθήκη στη Αρχή Κειμένου

Prepending Text to File



- **Με Ενδιάμεσο Αρχείο**

```
$ echo 'task goes here' | cat - todo.txt >  
temp && mv temp todo.txt
```

- **Με SED (αργότερα)**

```
$ sed -i '1itask goes here' todo.txt
```

ή

```
$ sed -i '1s/^/task goes here\n/' todo.txt
```

- **Με Command Substitution (αργότερα)**

```
$ echo -e "task goes here\n$(cat todo.txt)"  
> todo.txt
```



Εξέταση Αρχείων

- Τύποι αρχείων
 - Διαφορετικά αρχεία έχουν διαφορετική χρήση
 - Πώς γνωρίζεις τι τύπου είναι το αρχείο;
- Εντολή *file*
 - αναγνώριση τύπων αρχείων
 - Βασίζεται στην **ανάλυση των πρώτων λίγων bytes του κειμένου** όπου περιέχεται μια **ακολουθία (magic number)** η οποία είναι μοναδική για κάθε τύπο εμπορικού αρχείου.



Εξέταση Αρχείων

```
bash-3.1$ file test/  
test/: directory  
bash-3.1$ file test/test1/test1.txt  
test/test1/test1.txt: ASCII text  
bash-3.1$ file test.c  
test.c: ASCII C program text  
bash-3.1$ file tes  
tes: executable (RISC System/6000  
    V3.1) or obj module not stripped  
bash-3.1$
```

Εξέταση Αρχείων

Μηχανισμός Αναγνώρισης



[http://en.wikipedia.org/wiki/Magic_number_\(programming\)](http://en.wikipedia.org/wiki/Magic_number_(programming))

- [Compiled Java](#) class files ([bytecode](#)) start with 0xCAFEBAFE on [big-endian](#) systems.
- [GIF](#) image files have the [ASCII](#) code for 'GIF89a' (0x474946383961) or 'GIF87a' (0x474946383761)
- [JPEG](#) image files begin with 0xFFD8FF, and JPEG/[JFIF](#) files contain the [ASCII](#) code for 'JFIF' (0x4A464946) or JPEG/[EXIF](#) files contain the [ASCII](#) code for 'Exif' (0x45786966) beginning at byte 6 in the file, followed by more [metadata](#) about the file.
- [PNG](#) image files begin with an 8-[byte](#) signature which identifies the file as a PNG file and allows immediate detection of some common file-transfer problems: \211 P N G \r \n \032 \n (0x89504e470d0a1a0a)
- Standard [MIDI](#) music files have the [ASCII](#) code for 'MThd' (0x4D546864) followed by more metadata about the file.
- Unix script files usually start with a [shebang](#), '#' (0x2321, or 0x2123 on [little-endian](#) processors) followed by the path to an interpreter.
- [PostScript](#) files and programs start with '%!' (0x2521).
- [PDF](#) files start with '%PDF'.
- Old [MS-DOS](#) .exe files and the newer [Microsoft Windows PE](#) (Portable Executable) .exe files start with the [ASCII](#) string 'MZ' (0x4D5A), the initials of the designer of the file format, [Mark Zbikowski](#). The definition allows 'ZM' as well but it is quite uncommon.
- The [Berkeley Fast File System](#) superblock format is identified as either 0x19540119 or 0x011954 depending on version; both represent the birthday of author [Marshall Kirk McKusick](#).
- Executables for the [Game Boy](#) and [Game Boy Advance](#) handheld video game systems have a 48-byte or 156-byte magic number, respectively, at a fixed spot in the header. This magic number encodes a bitmap of the [Nintendo](#) logo.
- [Zip](#) files begin with 'PK', the initials of [Phil Katz](#), author of [DOS](#) compression utility [PKZIP](#).
- Old [Fat binaries](#) (containing code for both [68K](#) processors and [PowerPC](#) processors) on Classic [Mac OS](#) contained the [ASCII](#) code for 'Joy!' (0x4A6F7921) as a prefix.
- [TIFF](#) files begin with either "II" or "MM" depending on the byte order (II for Intel, or little endian, MM for Motorola, or big endian), followed by 0x2A00 or 0x002A (decimal [42](#) as a 2-byte integer in Intel or Motorola byte ordering).
- [Unicode](#) text files, encoded with [UTF-8](#), often start with \xEF\xBB\xBF (i»¿ in a [Latin-1](#) text editor), the UTF-8 encoding of the [Byte Order Mark](#). If they are encoded with [UTF-16](#), they often start with the [Byte Order Mark](#) coded as \xFE\xFF or \xFF\xFE depending on [endianness](#).



Εξέταση Αρχείων

- Εντολή *head* (επιλογή *-n*)
 - Εμφάνιση αρχικών γραμμών αρχείων
 - 10 γραμμές εξ' ορισμού
 - Προσφέρει ένα γρήγορο τρόπο να δεις εάν είναι αυτό το αρχείο που γυρεύεις.
 - Χρήση: *head -n filename*
 - Τυπώνει τις πρώτες *n* γραμμές



Εξέταση Αρχείων

- Εντολή *tail* (επιλογή *-n*)
 - Εμφάνιση τελικών γραμμών αρχείων
 - 10 γραμμές εξ' ορισμού
 - Προσφέρει ένα γρήγορο τρόπο να δεις εάν είναι αυτό το αρχείο που γυρεύεις από το τέλος.
 - Χρήση: *tail -n filename*
 - Τυπώνει τις τελευταίες *n* γραμμές



Εξέταση Αρχείων

- Εντολή *cat* (επιλογές *-s*, *-n*)
 - *cat* → *concatenate* (συνένωση)
 - εμφάνιση πολλών αρχείων, το ένα μετά το άλλο
 - *cat* <filename1> <filename2> ...
 - Επιλογή *-s*
 - Συμπύσσει (*suppress*) πολλαπλές κενές γραμμές σε μια μόνο
 - Επιλογή *-n*
 - προσθέτει αρίθμηση γραμμών στην έξοδο (όχι στο ίδιο το αρχείο).



Εξέταση Αρχείων

- Εντολή **wc** (επιλογές **-l**, **-w**, **-c**)
 - Εκθέτει τον αριθμό των γραμμών (**line**), λέξεων (**word**) και χαρακτήρων (**char**) ενός αρχείου
 - Χρήση: **wc** *<filename>*

```
bash-3.1$ wc test-math.c  
20 30 174 test-math.c
```

```
bash-3.1$
```



Εξέταση Αρχείων

- Εντολή **sort** (επιλογές *-r*, *-n*, *-k*)
 - Ταξινόμηση των γραμμών αρχείου σε αλφαβητική σειρά
 - Χρήση: **sort** [επιλογές] <filename>
 - Είναι χρήσιμη για αρχεία δεδομένων ή έξοδο από προγράμματα
 - Επιλογή **-r (reverse)**
 - ταξινόμηση σε φθίνουσα σειρά (larges=>smallest)
 - Επιλογή **-n (sort numeric)**
 - ταξινόμηση με αριθμητική σειρά
 - Επιλογή **-k [number]**
 - ταξινόμηση με βάση τη στήλη #number
 - Ακολουθεί παράδειγμα



Εξέταση Αρχείων

- Εντολή **uniq** (επιλογή -c)
 - Αφαιρεί *συνεχόμενα* διπλά αντίγραφα γραμμών (**διπλότυπα**)
 - Γιατί πρέπει να είναι *συνεχόμενα*;
 - Χρήση: **uniq** [επιλογές] <filename>
 - Επιλογή **-c (count)**
 - Εκθέτει κάθε εμφάνιση με τον αριθμό των *διπλότυπων* γραμμών.
 - Επιλογή **-u (unique)**
 - *Τυπώνει μόνο τις γραμμές που εμφανίζονται μια φορά.*



Εξέταση Αρχείων

```
bash-3.1$ cat test-uniq.test
```

```
Dan
```

```
Dan
```

```
Dan
```

```
Athena
```

```
Ryu
```

```
Ryu
```

```
Athena
```

```
Nick
```

```
bash-3.1$ uniq test-uniq.test
```

```
Dan
```

```
Athena
```

```
Ryu
```

```
Athena
```

```
Nick
```

```
bash-3.1$ uniq -c test-uniq.test
```

```
3 Dan
```

```
1 Athena
```

```
2 Ryu
```

```
1 Athena
```

```
1 Nick
```

Τι χρειάζεται για να
μείνει μια φορά η
εγγραφή της “Athena”?



Χαρακτήρας Τερματισμού Γραμμής

- *Το newline character είναι διαφορετικό σε διαφορετικά Λειτουργικά Συστήματα!*
 - **LF (\n)**: Multics, Unix and Unix-like systems (Linux, OS X, FreeBSD, AIX, Xenix, etc.), BeOS, Amiga, RISC OS, and other
 - **CR+LF (\r\n)**: **Microsoft Windows**, DOS (MS-DOS, PC DOS, etc.), DEC TOPS-10, RT-11, CP/M, MP/M, Atari TOS, OS/2, Symbian OS, PalmOS, Amstrad CPC, and most other early non-Unix and non-IBM Oses
 - **CR (\r)**: Commodore 8-bit machines, Acorn BBC, ZX Spectrum, TRS-80, Apple II family, Oberon, **Mac OS up to version 9**, MIT Lisp Machine and OS-9
 - 0x9B ([Atari 8-bit machines](#)), LF+CR ([RISC OS](#) spooled text)
- *dos2unix and unix2dos - DOS/MAC to UNIX text file format converter*



Κανονικές Εκφράσεις

- Μια **Κανονική Έκφραση (Regular Expression)** είναι ένα πρότυπο που περιγράφει ένα σύνολο συμβολοσειρών.
 - Σημαντική έννοια στα πλαίσια των **μεταγλωττιστών, θεωρίας υπολογισμού**, αλλά και **προγραμματιστικών γλωσσών**.
 - Ο βασικότερος τρόπος εκτέλεσης κανονικών εκφράσεων στο UNIX είναι με χρήση της εντολής **grep (ή egrep)**
 - Κανονικές εκφράσεις κατασκευάζονται χρησιμοποιώντας **μικρότερες εκφράσεις**.
- Στη Θεωρία Υπολογισμού είχαμε μάθει ότι οι κανονικές εκφράσεις (regular expressions, regex ή regexp) χρησιμοποιούνται για την περιγραφή γλωσσών με απλά σύμβολα, το NULL και συνδυασμούς που προκύπτουν με εφαρμογή ένωσης (U), του αστεριού Κλήνυ (Kleene Star) (*) ή και παρενθέσεων
 - Π.χ., $(a^*b^*)^*(abba)(a^*b^*)^*$



Εντολή *grep*

- Εντολή *grep* (επιλογές *-i*, *-n*, *-v*, *-w*, *-l*)
 - Ψάχνει σε αρχεία εισόδου για γραμμές που περιέχουν ένα πρότυπο που δίνεται και τις εμφανίζει στην έξοδο.

grep <options> <pattern> <filename>

- Παραδείγματα

bash-3.1\$ grep -n root /etc/passwd

1: root:x:0:0:root:/root:/bin/bash

12: operator:x:11:0:operator:/root:/sbin/nologin

-v option: invert match (show lines that don't contain the word bash)

bash-3.1\$ grep -v bash /etc/passwd | grep -v 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

news:x:9:13:news:/etc/news:



Εντολή *grep*: Επιλογές (Options)

- **grep Options** (επιλογές *-i*, *-n*, *-v*, *-w*, *-l*, *-C*)
 - **Επιλογή *-i* (*case-insensitive-match*)**
 - *Case-insensitive* - αγνοεί το διαχωρισμό κεφαλαίων και μικρών γραμμάτων
 - **Επιλογή *-n* (*numeric-prefix*)**
 - εκτυπώνει την αρίθμηση των γραμμών στην έξοδο
 - **Επιλογή *-v* (*invert-match*)**
 - εμφανίζει τις γραμμές που ΔΕΝ ταιριάζουν με το πρότυπο που ψάχνουμε
 - **Επιλογή *-w* (*word-match*) – Ψάξιμο για λέξεις**
 - ταιριάζει το πρότυπο με ολοκληρωμένες λέξεις (δεν περιέχουν τους χαρακτήρες της τιμής περιβάλλοντος \$IFS, η οποία εξ' ορισμού περιέχει τα whitespace: newline / tab / space)
 - **Επιλογή *-l* (*latin-el*) (*filename-with-match*)**
 - Απόκρυψη της κανονικής ροής στο Stdout. Τύπωσε μόνο τα **ονόματα των αρχείων** (όχι τα αποτελέσματα) όπου γίνεται match
 - **Επιλογή *-C* #** εκτύπωση γραμμής πριν + μετά από match
 - # Print NUM lines of output context. (1 line before, 1 after, *-C 0* means only the 1st line succeeding the match – see example next)

Εντολή *grep*: Κανονικές Εκφράσεις (*Patterns*)



- Η *grep* χειρίζεται δυο διαφορετικές εκδοχές σύνταξης κανονικών εκφράσεων: τις βασικές και εκτεταμένες.

– Οι **Βασικές κανονικές εκφράσεις (BRE)** μπορούν να χρησιμοποιηθούν από την εντολή *grep* (όμοιο με **GNU grep**)

- Όπως ορίστηκε από τον Ken Thompson στην αρχική υλοποίηση του για το UNIX.

– Οι **Εκτεταμένες Κανονικές Εκφράσεις (ERE)** μπορούν να χρησιμοποιηθούν από την εντολή *egrep*

- Π.χ., `egrep 'word1 | word2' filename`

Κανονικές Εκφράσεις



Extended REGEX (POSIX)

Basic REGEX (POSIX)

Regex Syntax Summary

This table summarizes the meaning of various strings in different regex syntaxes. It is intended as a quick reference, rather than a tutorial or specification. Please report any errors.

String	GNU grep	BRE (grep)	ERE (egrep)	GNU Emacs	Perl	Python	Tcl
.	Any character	Any character except \0		Any character except \n			Any character
[...]	Bracket Expression			Character Set	Character Class		Bracket Expression
\(re\)	Subexpression			Grouping			
re\{...\}	Match re multiple times			Match re multiple times			
(re)		Subexpression			Grouping		
re{...}		Match re multiple times			Match re multiple times		
re{...}?					Nongreedy {}		
\digit				Back-reference			
^				Start of line			
\$				End of line			
re?				<i>re</i> 0 or 1 times			
re*				<i>re</i> 0 or more times			
re+				<i>re</i> one or more times			
l r			<i>l</i> or <i>r</i>		<i>l</i> or <i>r</i>		
*?					Non-greedy *		
+?					Non-greedy +		
??					Non-greedy ?		

...

<http://www.greenend.org.uk/rjk/tech/regexp.html>

Εντολή *grep*: Παραδείγματα



- Παραδείγματα (συνέχεια)

^ Match lines beginning with root

```
bash-3.1$ grep ^root /etc/passwd  
root:x:0:0:root:/root:/bin/bash
```

```
bash-3.1$ grep :$ /etc/passwd  
news:x:9:13:news:/etc/news:
```

: Match lines ending with “:”

\< means “Start of Word”: Match lines that contain the word PATH.

```
bash-3.1$ grep export ~/.profile | grep \<PATH  
export PATH
```

Any line containing export

```
#export PATH=$PATH:/usr/local/ns-allinone-2.1b6/bin  
export PATH=$PATH:/hdd2/cchrys/ns-allinone-  
2.1b9/bin:/hdd2/cchrys/ns-allinone-  
2.1b9/tcl8.3.2/unix:/hdd2/cchrys/ns-allinone-  
2.1b9/tk8.3.2/unix  
#export PATH=$PATH:/Backup/cchrys/OMNeTpp/omnetpp-2.0b5/bin
```

*Σημείωση: Εναλλακτική υλοποίηση με το **grep -w PATH***



Εντολή *grep*: Παραδείγματα

- Παραδείγματα (συνέχεια)

Match lines containing word /

```
bash-3.1$ grep -w / /etc/fstab  
/dev/VolGroup00/LogVol100 /      ext3      defaults      1 1
```

[] Match any line that contains any of the characters yf

```
bash-3.1$ grep [yf] /etc/group
```

```
sys:x:3:root,bin,adm
```

```
tty:x:5:
```

*[^] Match any line that **does not contain** the following chars*

```
ftp:x:50:
```

```
nobody:x:99:
```

```
floppy:x:19:
```

```
nsnobody:x:65534:
```

```
xfs:x:43:
```



grep Before|After

- **Before** N lines

- `grep -B <NUM> <string>`

```
cat ~/.profile | grep -B 1 bin
--
# The original version is saved in .profile.pysave
PATH="/Library/Frameworks/Python.framework/Versions/3.11/bin:${PATH}"
--
# The original version is saved in .profile.pysave
PATH="/Library/Frameworks/Python.framework/Versions/3.11/bin:${PATH}"
```

- **After** N lines

- `grep -A <NUM> <string>`

Παρένθεση περί Εισαγωγικών



- **Μονά εισαγωγικά (' ')**

- χρησιμοποιούνται για να διατηρήσουν τους χαρακτήρες που εμπερικλείονται σε μονά εισαγωγικά ως είναι (κυριολεκτικά)

- **Παράδειγμα**

```
bash-3.1$ echo '$HOME'  
$HOME
```

- **Διπλά Εισαγωγικά (" ")**

- χρησιμοποιούνται για να διατηρήσουν τους χαρακτήρες που εμπερικλείονται σε διπλά εισαγωγικά ως είναι (κυριολεκτικά), εκτός από το \$ (dollar sign), τα ` (*backward single quotes*) και το \ (*backslash*).

- **Παράδειγμα**

```
bash-3.1$ echo "$HOME"  
/home/faculty/dzeina
```

Κανονικές Εκφράσεις



- Μια κανονική έκφραση μπορεί να χρησιμοποιήσει τους ακόλουθους *metacharacters* (*repetition operators*) – **Extended Regular Expression (ERE)** – *used with egrep*

Operator	Effect
.	Matches any single character (1:1)
?	The preceding item will be matched zero times or once (0:1)
*	The preceding item will be matched zero or more times (0:N)
+	The preceding item will be matched one or more times (1:N)
{N}	The preceding item is matched exactly N times (N:N)
{N,}	The preceding item is matched N or more times (N:-)
{N,M}	The preceding item is matched at least N times, but not more than M times (N:M)
–	Represents a range (e.g., [a-z] ή [^a-zA-Z])
^	Matches the empty string at the beginning of a line ; also represents the characters not in the range of list [^]
\$	Matches the empty string at the end of a line
\b	Matches the empty string at the edge of a word
\<	Matches the start-of-word (word that has a preceding whitespace)
\>	Matches the end-of-word (word that has a whitespace following)

POSIX Character Classes



- Finally, certain named classes of characters are predefined within bracket expressions, as follows.

POSIX class	similar to	meaning
<code>[:upper:]</code>	<code>[A-Z]</code>	uppercase letters
<code>[:lower:]</code>	<code>[a-z]</code>	lowercase letters
<code>[:alpha:]</code>	<code>[A-Za-z]</code>	upper- and lowercase letters
<code>[:digit:]</code>	<code>[0-9]</code>	digits
<code>[:xdigit:]</code>	<code>[0-9A-Fa-f]</code>	hexadecimal digits
<code>[:alnum:]</code>	<code>[A-Za-z0-9]</code>	digits, upper- and lowercase letters
<code>[:punct:]</code>		punctuation (all graphic characters except letters and digits)
<code>[:blank:]</code>	<code>[\t]</code>	space and TAB characters only
<code>[:space:]</code>	<code>[\t\n\r\f\v]</code>	blank (whitespace) characters
<code>[:cntrl:]</code>		control characters
<code>[:graph:]</code>	<code>[^[:cntrl:]]</code>	graphic characters (all characters which have graphic representation)
<code>[:print:]</code>	<code>[:graph]</code>	graphic characters and space

GREP Special Characters (BRE vs ERE)



- BRE: .*[]^\$ \ + \| \? BRE Char: \ . | * | ^ | \$ \| | \ + | ?
- ERE: .*[]^\$ + | ? ERE Char: \ . | * | ^ | \$ \| | \ + \| | ?

- **BRE Syntax: grep**

- \$ echo "aab." | grep 'a\+b\.'
- aab.
- \$ echo "aab+" | grep 'a\+b\+'
- aab+

- **ERE Syntax: grep -E or egrep**

- echo "aab." | grep -E 'a+b\.'
- aab.
- echo "aab+" | grep -E 'a+b\+'
- aab+



Εντολή *grep and egrep*

- BRE: .*[]^\$V | + \| | ? BRE Char: \ . | * | ^ | \$ | \| | V | + | | ?
- ERE: .*[]^\$V + | ? ERE Char: \ . | * | ^ | \$ | \| | V | + | \| | ?

ΤΡΟΠΟΣ Α: GREP (BRE) with «Backslash on REGEX Symbols |, +»

```
bash-3.1$ ls -l | grep '^-' | grep '+\|W\|+' # '+' or 'W'+
-rw-r--r--  1 cspgcc1 cspg      0 Jan 31 16:46 file+1.txt
-rw-r-----  1 cspgcc1 cspg    371 Feb 25  2005 WS_FTP.LOG
```

ΤΡΟΠΟΣ Α: EGREP (ERE) with «Backslash on Characters +»

```
bash-3.1$ ls -l | grep '^-' | egrep '\+|W+'
-rw-r--r--  1 cspgcc1 cspg      0 Jan 31 16:46 file+1.txt
-rw-r-----  1 cspgcc1 cspg    371 Feb 25  2005 WS_FTP.LOG
```

Εντολή *grep*: Παραδείγματα



- Παραδείγματα (συνέχεια) Case-insensitive matching of words beginning with i

```
bash-3.1$ ls -l | grep -i '\<i'
-rwx----- 1 cchrys tspecial 247605 Jan 12 2005 iasted1.pdf
-rwx----- 1 cchrys tspecial 180597 Jan 12 2005 iasted2.pdf
-rwx----- 1 cchrys tspecial 179801 Dec 30 2003
ICIS_chrysostomou.pdf
-rwx----- 1 cchrys tspecial 156920 Dec 30 2003
ICIS_pitsillides.pdf
drwxr-x--- 2 cchrys tspecial 4096 Aug 19 2002 idcc
drwxr-x--- 5 cchrys tspecial 4096 Jul 17 2001 IPE
```

```
bash-3.1$ ls -l | grep '^d' | grep -i '\<i'
drwxr-x--- 2 cchrys tspecial 4096 Aug 19 2002 idcc
drwxr-x--- 5 cchrys tspecial 4096 Jul 17 2001 IPE
```

Same but first filtering out directories (line starting with d)



Εντολή *grep*

- Παραδείγματα (συνέχεια)

Single, Double quotes important when set here ("\$HOME")

```
bash-3.1$ grep '^c.*h$' /usr/share/dict/words | more
```

```
cabbalah  
cable-stitch  
cablish  
caddish  
cadish  
caducibranch  
cafeneh  
cafh  
cailleach  
cailliach  
calabash  
calash  
calenturish  
calfish  
calipash  
caliph  
calligraph  
callipash  
calvish  
calycanth  
camelish  
cameograph  
camooch  
--More--
```

Start with C, one char (any), 0:M chars, end with h

```
bash-3.1$ grep '^c..h$' /usr/share/dict/words | more
```

```
cafh  
caph  
cash  
cath  
chih  
coch  
cosh  
coth  
croh  
csch  
cush
```

Start with C, two chars (any), end with h

Σύνοψη Διάλεξης



- *Εντολές που είδαμε σήμερα:*
 - *passwd, man, whatis, apropos, info, help, users, who, whoami, finger, w,*
- *Επανάληψη από ΕΠΛ232:*
 - *>|, grep/egrep, file, head, tail, cat, more, wc, sort, uniq,*
- *Τι κάνει η κάθε μια από τις πιο κάτω εντολές;*
(Επανάληψη από ΕΠΛ232 στο σπίτι)
 - *jobs, ps, fg, bg, kill, top,*
 - *alias, cut, tr, tee, comm, diff, iconv, file*
 - *mail, crontab*
 - *which, whereis, find, exec, xargs*