

# Εισαγωγή στα Λειτουργικά Συστήματα



SET ΔΙΑΦΑΝΕΙΩΝ 18

ΑΝΑΣΚΟΠΗΣΗ ΦΪΛΤΡΩΝ

ΑΝΤΩΝΗΣ ΣΙΔΗΡΟΠΟΥΛΟΣ

# head

2

- **SYNTAX:** `head [OPTION] . . . [FILE] . . .`
- **Συνήθη χρήση:**
  - `head -x` (θα τυπώσει τις x πρώτες γραμμές από την είσοδο)

# tail

3

- **SYNTAX:** `tail [OPTION] ... [FILE] ...`
- **Συνήθη χρήση:**
  - `tail -x` (θα τυπώσει τις x τελευταίες γραμμές από την είσοδο)

- **SYNTAX:** `wc [OPTION] . . . [FILE] . . .`
- **Συνήθη χρήση:**
  - μετράει από την είσοδο πλήθος byte, λέξεων και γραμμών.
  - `-l` : μετράει τις γραμμές
  - `-w`: μετράει τις λέξεις
  - `-c`: μετράει τους χαρακτήρες (bytes)
  - `-m`: μετράει τους χαρακτήρες (characters) – Προσοχή στα UTF αρχεία.

# grep/egrep

5

- **SYNTAX:**

`grep [OPTION] ... REGEXP [FILE] ...`

- **Συνήθη χρήση:**

- Εμφανίζει μόνο τις γραμμές στις οποίες ταιριάζει η ΚΕ (REGEXP)
- -v: αντιστροφή αποτελέσματος
- -i: ignore case
- -n: εμφάνιση αριθμών γραμμών.

# sed

6

- **SYNTAX:**

`sed [OPTION] ... COMMAND [FILE] ...`

- **Συνήθη χρήση:**

- Στην sed υπάρχουν πολλές εντολές. Οι περισσότερες χρησιμοποιούμενες: p (print), d (delete), s (substitute)
- Εφαρμόζει την εντολή ή τις εντολές που ορίστηκαν στην είσοδο και εμφανίζει όλο το αρχείο.
- -n: δεν εμφανίζει όλο το αρχείο. Μόνο τις γραμμές που κάνουμε p (print)
- -r: Χρήση ERE (Εκτεταμένου συνόλου κανονικών εκφράσεων)

- **SYNTAX:**

`tr [OPTION] ... SET1 [SET2]`

- **Συνήθη χρήση:**

- Μετατροπή ενός συνόλου χαρακτήρων σε ένα άλλο. πχ:

`tr o-3 AQWE`

Ο χαρακτήρας `o`  $\rightarrow$  `A`, `1`  $\rightarrow$  `Q`, `2`  $\rightarrow$  `W`, `3`  $\rightarrow$  `E`

- `tr -s x`

όπου υπάρχουν πολλές συνεχόμενες εμφανίσεις του χαρακτήρα `x`, θα γίνουν μια.

- `tr -d x`

ο χαρακτήρας `x` θα διαγραφεί.

# sort

8

- **SYNTAX:**

`sort [OPTION] . . . [FILE] . . .`

- **Συνήθη χρήση:**

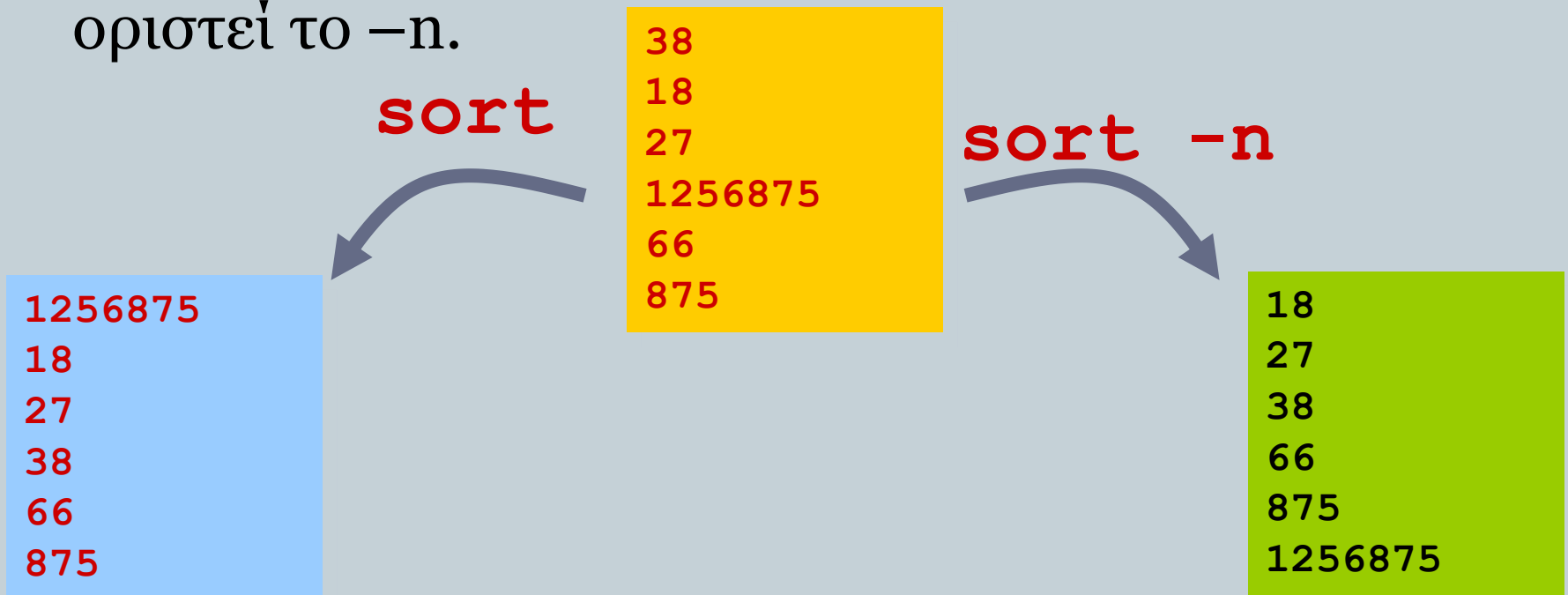
- Ταξινόμηση των γραμμών εισόδου
- `sort -k2n,3 -k5r`: κάνε ταξινόμηση με βάση το 2<sup>ο</sup> (αριθμός) και 3<sup>ο</sup> πεδίο και σε περίπτωση ισότητας με βάση το 5<sup>ο</sup> πεδίο με αντίστροφη σειρά.
- `-f`: ignore case
- `-r`: reverse
- `-n`: numeric sort
- `-tx`: Ο χαρακτήρας x είναι ο διαχωριστής πεδίων
- `-kx`: συνέκρινε από το πεδίο x και μετά
- `-kx,y`: συνέκρινε από το πεδίο x έως το y



# Numeric vs. Alphabetic

9

- Εξ ορισμού, χρησιμοποιείται η αλφαβητική διάταξη – για να γίνει αριθμητική ταξινόμηση πρέπει να οριστεί το  $-n$ .



# Sort με ανακατεύθυνση εξόδου

10

```
sort file1 > file2
```

```
sort < file1 > file2
```

- Οι πάνω εντολές έχουν το ίδιο αποτέλεσμα. Στην 1<sup>η</sup> περίπτωση η sort διαβάσει το αρχείο file1. στην 2<sup>η</sup> περίπτωση το shell δίνει ως είσοδο της sort τα περιεχόμενα του file1.

```
sort file1 > file1 (ΛΑΘΟΣ)
```

```
sort file1 -o file1
```

- Στην 1<sup>η</sup> περίπτωση το file1 μηδενίζεται πριν αρχίσει να εκτελείται η sort. Άρα όταν θα το διαβάσει θα είναι κενό.
- Στην 2<sup>η</sup> περίπτωση λέμε στην sort να διαβάσει τα περιεχόμενα του file1 και μετά το αποτέλεσμα να το αποθηκεύσει στο file1. Δουλεύει λόγω του τρόπου λειτουργίας της sort.

# uniq

11

- **SYNTAX:**

`uniq [OPTION]... [INPUT [OUTPUT]]`

- **Συνήθη χρήση:**

- Εμφάνιση (και μέτρηση) των μοναδικών γραμμών εισόδου
- Οι ίδιες γραμμές πρέπει να είναι συνεχόμενες, συνεπώς συνηθίζεται τα δεδομένα πρώτα να ταξινομούνται.
- **-c**: μέτρηση των ίδιων γραμμών.

```
asidirop@antonis-PC:/tmp$ uniq -c file1
  2 AAA
  1 BBB
  1 AAA
asidirop@antonis-PC:/tmp$ sort file1 | uniq -c
  3 AAA
  1 BBB
asidirop@antonis-PC:/tmp$
```

**file1**

AAA  
AAA  
BBB  
AAA

# cut

12

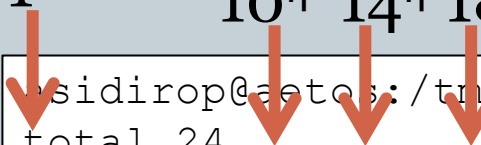
- **SYNTAX:**  
`cut OPTION... [FILE]...`
- **Συνήθη χρήση:**
  - Εμφάνιση συγκεκριμένων στηλών από την είσοδο.
- **Έχει 2 καταστάσεις λειτουργίας**
  - Με τον προσδιοριστή `-c` επιλέγει στήλες χαρακτήρων (columns)
  - Με τον προσδιοριστή `-f` επιλέγει στήλες πεδίων (fields)

# cut

13

- Εμφάνιση των στηλών 1-10 και 14-18.
- Οι αριθμοί των στηλών πρέπει να είναι σε αύξουσα σειρά

1<sup>η</sup> 10<sup>η</sup> 14<sup>η</sup> 18<sup>η</sup>



```
asidirop@aetos:/tmp$ ls -l
total 24
-rw-r--r-- 1 asidirop conit  18 May 30 12:58 hours
-rw-r--r-- 1 asidirop conit  24 May 30 13:00 hours2
-rw-r--r-- 1 asidirop conit  45 May 30 13:02 hours3
-rw-r--r-- 1 asidirop conit 368 May 30 13:07 list
-rw----- 1 gpolititis x1112   0 May 31 13:03 mutt-aetos
asidirop@aetos:/tmp$ ls -l | cut -c1-10,14-18
total 24
-rw-r--r--asidi
-rw-r--r--asidi
-rw-r--r--asidi
-rw-r--r--asidi
-rw-----gpoli
```

- Παραδείγματα

- `cut -f1-3,5` αποσπά τα πεδία 1 έως 3 και 5 με διαχωριστή το TAB.
- Όταν χρησιμοποιείται στη μορφή αυτή ΠΡΕΠΕΙ ΝΑ ΔΗΛΩΘΕΙ Ο ΔΙΑΧΩΡΙΣΤΗΣ ΠΕΔΙΟΥ με την επιλογή `-d` (delimiter) π.χ.
- `cut -d: -f2-4 file` διαχωριστής το :
- Ο εξ ορισμού διαχωριστής είναι το tab
- Όταν διαχωριστής είναι το κενό διάστημα δηλώνεται μέσα σε εισαγωγικά δηλ. `-d" "` (ή μονά)

# cut

15

- Εμφανίζει τα πεδία 1 και 3 με διαχωριστή το “:”.

```
asidirop@antonis-PC:/tmp$ cat /etc/group | cut -d ':' -f1,3  
root:0  
daemon:1  
bin:2  
sys:3  
adm:4  
tty:5  
disk:6  
lp:7
```

# cut

16

- Από την έξοδο της `ls -l` θέλουμε να εμφανίσουμε τις στήλες 1 (άδειες) και 7 (ημέρα) με διαχωριστή το space.
- **κάτι δεν πάει καλά!!!!**

```
asidirop@aetos:/tmp$ ls -l
total 24
-rw-r--r-- 1 asidirop conit 18 May 30 12:58 hours
-rw-r--r-- 1 asidirop conit 24 May 30 13:00 hours2
-rw-r--r-- 1 asidirop conit 45 May 30 13:02 hours3
-rw-r--r-- 1 asidirop conit 368 May 30 13:07 list
-rw----- 1 gpolitis x1112 0 May 31 13:03 mutt-aetos
asidirop@aetos:/tmp$ ls -l | cut -d ' ' -f1,7
total
-rw-r--r-- May
-rw-r--r-- May
-rw-r--r-- May
-rw-r--r-- 30
-rw----- 0
```



# cut

17

- Το 5<sup>ο</sup> πεδίο είναι το κενό string. Αυτό διότι μετά την λέξη conit (4<sup>ο</sup>) υπάρχουν 2 spaces.

1 <sup>ο</sup>	2 <sup>ο</sup>	3 <sup>ο</sup>	4 <sup>ο</sup>	5 <sup>ο</sup>	6 <sup>ο</sup>	7 <sup>ο</sup>
total	24					
-rw-r--r--	1	asidirop	conit	18	May	30 12:58 hours
-rw-r--r--	1	asidirop	conit	24	May	30 13:00 hours2
-rw-r--r--	1	asidirop	conit	45	May	30 13:02 hours3
-rw-r--r--	1	asidirop	conit	368	May	30 13:07 list
-rw-----	1	gpolititis	x1112	0	May	31 13:03 mutt-aetos

```
asidirop@aetos:/tmp$ ls -l | cut -d ' ' -f1,7
```

1 <sup>ο</sup>	2 <sup>ο</sup>	3 <sup>ο</sup>	4 <sup>ο</sup>	5 <sup>ο</sup>	6 <sup>ο</sup>	7 <sup>ο</sup>
total						
-rw-r--r--		May				
-rw-r--r--		May				
-rw-r--r--		May				
-rw-r--r--		30				
-rw-----		0				

# cut

18

- Για να μπορέσουμε να χειριστούμε τέτοιες περιπτώσεις με την cut θα πρέπει να εξαλείψουμε τις πολλαπλές εμφανίσεις του space. Αυτό γίνεται με την tr.

```
asidirop@aetos:/tmp$ ls -l | tr -s ' '
total 24
-rw-r--r-- 1 asidirop conit 18 May 30 12:58 hours
-rw-r--r-- 1 asidirop conit 24 May 30 13:00 hours2
-rw-r--r-- 1 asidirop conit 45 May 30 13:02 hours3
-rw-r--r-- 1 asidirop conit 368 May 30 13:07 list
-rw----- 1 gpolititis x1112 0 May 31 13:03 mutt-aetos
asidirop@aetos:/tmp$ ls -l | tr -s ' ' | cut -d ' ' -f1,7
total
-rw-r--r-- 30
-rw-r--r-- 30
-rw-r--r-- 30
-rw-r--r-- 30
-rw----- 31
```