Laborator 2

```
biancapinghireac@vbox:~/SO/lab2$ echo ceva > file.txt
biancapinghireac@vbox:~/SO/lab2$ cat file.txt
ceva
biancapinghireac@vbox:~/SO/lab2$ cat file.txt | gzip > file.txt.gz
biancapinghireac@vbox:~/SO/lab2$ zcat file.txt.gz
ceva
```

Prin echo mesaj > fisier, scriem in fisier mesajul

Gzip comprima fisierul dat

Zcat si zless afiseaza fisierul comprimat (zless – interactiv)

```
biancapinghireac@vbox:~/SO/lab2$ cat < file.txt
ceva
biancapinghireac@vbox:~/SO/lab2$ cat file.txt
ceva
```

Cele 2 comenzi afiseaza acelasi lucru

```
biancapinghireac@vbox:~/SO/lab2$ echo "linia 2" >> file.txt
biancapinghireac@vbox:~/SO/lab2$ cat file.txt
ceva
linia 2
biancapinghireac@vbox:~/SO/lab2$ echo -n "linia 3" >> file.txt
biancapinghireac@vbox:~/SO/lab2$ echo " continuare" >> file.txt
biancapinghireac@vbox:~/SO/lab2$ echo "linia 4" >> file.txt
biancapinghireac@vbox:~/SO/lab2$ cat file.txt
ceva
linia 2
linia 3 continuare
linia 4
```

Echo automat adauga simbolul de new line la finalul textului pe care vrei sa il adaugi cu “>>” la un fisier deja existent

Daca folosesti -n atunci nu adauga simbolul new line la final

```
biancapinghireac@vbox:~/SO/lab2$ echo << EOF
> ala bala
> portocala
> EOF
```

Simbolul “<<”(here-document) indica secventa de oprire in citire (in cazul acesta se opreste la intalniera lui “EOF”)

```
biancapinghireac@vbox:~/SO/lab2$ mkdir extra
biancapinghireac@vbox:~/SO/lab2$ touch extra/Makefile
biancapinghireac@vbox:~/SO/lab2$ touch extra/Makefile2
biancapinghireac@vbox:~/SO/lab2$ touch extra/func.c
biancapinghireac@vbox:~/SO/lab2$ touch extra/main.c
biancapinghireac@vbox:~/SO/lab2$ tar -cvf extra.tar extra/
extra/
extra/Makefile
extra/Makefile2
extra/func.c
extra/main.c
```

Tar - arhivare

-c -constructia arhivei

-v -verbose (afiseaza tot)

-f -urmat de numele arhivei care se creaza

```
biancapinghireac@vbox:~/SO/lab2$ mkdir dup
biancapinghireac@vbox:~/SO/lab2$ tar -xf extra.tar -C dup
biancapinghireac@vbox:~/SO/lab2$ ls -lR dup/
dup/:
total 0
drwxr-xr-x. 1 biancapinghireac biancapinghireac 58 Mar  5 11:39 extra

dup/extra:
total 0
-rw-r--r--. 1 biancapinghireac biancapinghireac 0 Mar  5 11:39 func.c
-rw-r--r--. 1 biancapinghireac biancapinghireac 0 Mar  5 11:39 main.c
-rw-r--r--. 1 biancapinghireac biancapinghireac 0 Mar  5 11:38 Makefile
-rw-r--r--. 1 biancapinghireac biancapinghireac 0 Mar  5 11:38 Makefile2
```

Daca folosim in loc de -c, -x – se face dezarhivare

-C indica folderul destiantie

Comenziile din ls:

-l pentru "long list"

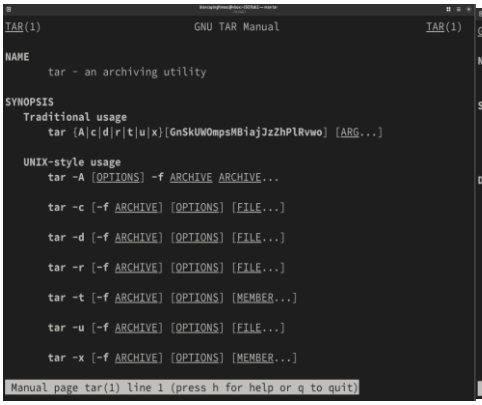
-R pentru parcurgerea recursivă a directoarelor găsite

```
biancapinghireac@vbox:~/SO/lab2$ tar -c extra/ | gzip > extra.tgz
biancapinghireac@vbox:~/SO/lab2$ gunzip -c extra.tgz | tar -x -C dup/
biancapinghireac@vbox:~/SO/lab2$ tar -zcf extra.tar.gz extra/
biancapinghireac@vbox:~/SO/lab2$ tar -zxf extra.tar.gz -C dup/
```

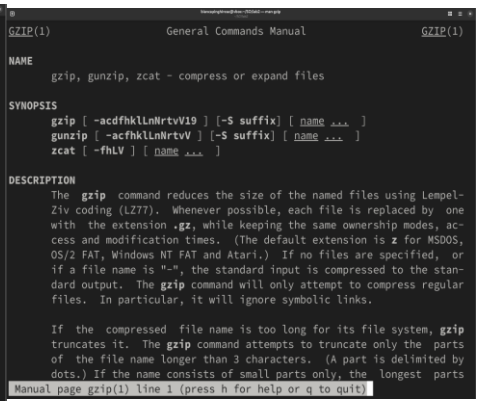
(toate aceste comenzi sunt echivalente si realizeaza arhivarea si comprimarea directorului “extra”)

Arhivarea o cuplăm cu comprimarea folosind comenziile: gzip , gunzip

Manualul comenzilor folosite:



```
TAR(1)                          GNU TAR Manual                          TAR(1)

NAME
       tar - an archiving utility

SYNOPSIS
   Traditional usage
       tar {A|c|d|r|t|u|x}[GnSkUWOmpsMBiajJzZhPlRvwo] [ARG...]

   UNIX-style usage
       tar -A [OPTIONS] -f ARCHIVE ARCHIVE...

       tar -c [-f ARCHIVE] [OPTIONS] [FILE...]

       tar -d [-f ARCHIVE] [OPTIONS] [FILE...]

       tar -r [-f ARCHIVE] [OPTIONS] [FILE...]

       tar -t [-f ARCHIVE] [OPTIONS] [MEMBER...]

       tar -u [-f ARCHIVE] [OPTIONS] [FILE...]

       tar -x [-f ARCHIVE] [OPTIONS] [MEMBER...]

 Manual page tar(1) line 1 (press h for help or q to quit)
```



```
GZIP(1)                  General Commands Manual                  GZIP(1)

NAME
       gzip, gunzip, zcat - compress or expand files

SYNOPSIS
       gzip [ -acdfhklLnNrtvV19 ] [-S suffix] [ name ... ]
       gunzip [ -acfhklLnNrtvV ] [-S suffix] [ name ... ]
       zcat [ -fhLV ] [ name ... ]

DESCRIPTION
       The  gzip  command reduces the size of the named files using Lempel-
       Ziv coding (LZ77).  Whenever possible, each file is replaced by  one
       with  the extension .gz, while keeping the same ownership modes, ac-
       cess and modification times.  (The default extension is z for MSDOS,
       OS/2 FAT, Windows NT FAT and Atari.)  If no files are specified,  or
       if a file name is "-", the standard input is compressed to the stan-
       dard output.  The gzip command will only attempt to compress regular
       files.  In particular, it will ignore symbolic links.

       If  the  compressed  file name is too long for its file system, gzip
       truncates it.  The gzip command attempts to truncate only the  parts
       of  the file name longer than 3 characters.  (A part is delimited by
       dots.) If the name consists of small parts only, the  longest  parts

 Manual page gzip(1) line 1 (press h for help or q to quit)
```

```
biancapinghireac@vbox:~/SO/lab2$ zip -r extra.zip extra/*
  adding: extra/func.c (stored 0%)
  adding: extra/main.c (stored 0%)
  adding: extra/Makefile (stored 0%)
  adding: extra/Makefile2 (stored 0%)
biancapinghireac@vbox:~/SO/lab2$ unzip extra.zip
Archive:  extra.zip
replace extra/func.c? [y]es, [n]o, [A]ll, [N]one, [r]ename: A
 extracting: extra/func.c
 extracting: extra/main.c
 extracting: extra/Makefile
 extracting: extra/Makefile2
biancapinghireac@vbox:~/SO/lab2$ ls -l
total 32
drwxr-xr-x. 1 biancapinghireac biancapinghireac    10 Mar  5 11:44 dup
drwxr-xr-x. 1 biancapinghireac biancapinghireac    58 Mar 16 18:02 extra
-rw-r--r--. 1 biancapinghireac biancapinghireac 10240 Mar  5 11:39 extra.tar
-rw-r--r--. 1 biancapinghireac biancapinghireac   208 Mar 13 16:49 extra.tar.gz
-rw-r--r--. 1 biancapinghireac biancapinghireac   208 Mar 13 16:49 extra.tgz
-rw-r--r--. 1 biancapinghireac biancapinghireac   640 Mar 16 18:02 extra.zip
-rw-r--r--. 1 biancapinghireac biancapinghireac    40 Mar  5 11:24 file.txt
-rw-r--r--. 1 biancapinghireac biancapinghireac    25 Mar  5 11:16 file.txt.gz
```

Arhivam directorul folosing comanda zip si dezarhivam cu comanda unzip;

tar: Fiind doar un arhivator, este foarte rapid. Operația principală este concatenarea fișierelor, ceea ce este relativ simplu și eficient.

gzip: Oferă o compresie bună și o viteză decentă. Este o alegere populară pentru compresia fișierelor individuale, dar și în combinație cu **tar** (de exemplu, **tar -czf extra.tar.gz extra/**).

zip: Este mai versatil decât **gzip**, deoarece poate comprima fișiere individuale sau directoare întregi într-o singură arhivă. Compresia este în general mai slabă decât **gzip** pentru text și cod sursă, dar poate fi mai rapidă, mai ales la niveluri de compresie mai mici. Este, de asemenea, mai portabil, fiind suportat nativ de majoritatea sistemelor de operare.

SED e un program de editare si transformare a textului

Structura: `sed [optiuni] comenzi [fisier]`

```
biancapinghireac@vbox:~/SO/lab2$ cat > file.txt
words keep on coming
on and on and ON here
and onto here also
biancapinghireac@vbox:~/SO/lab2$ sed 's/on/off/' file.txt
words keep off coming
off and on and ON here
and offto here also
```

Pentru a schimba 2 cuvinte intre ele se foloseste `sed ‘s/regexp/replacement/’ fisier`

```
biancapinghireac@vbox:~/SO/lab2$ sed 's/on /off /' file.txt
words keep off coming
off and on and ON here
and onto here also
biancapinghireac@vbox:~/SO/lab2$ sed 's/on /off /g' file.txt
words keep off coming
off and off and ON here
and onto here also
```

Ca sa nu schimbe si aparitiile in cuvinte, se cauta si spatial

La final se adauga flag-ul ‘g’ (global) pentru a schimba toate aparitiile

```
biancapinghireac@vbox:~/SO/lab2$ sed 's/on /off /gi' file.txt
words keep off coming
off and off and off here
and onto here also
```

‘gi’ – global case-insensitive (adica si daca e litera mica si daca e litera mare)

```
biancapinghireac@vbox:~/SO/lab2$ sed 's/here/(&)/g' file.txt
words keep on coming
on and on and ON (here)
and onto (here) also
```

‘(&)’ formula pentru a pune paranteze, unde & este referinta spre cuvant

```
biancapinghireac@vbox:~/SO/lab2$ sed 's/\([a-zA-Z0-9][a-zA-Z0-9]*\) \([a-zA-Z0-9][a-zA-Z0-9
]*\)/\2 \1/' file.txt
keep words on coming
and on on and ON here
onto and here also
```

Componente ale comenzii:

paranteza este folosită pentru a grupa termenii expresiei regulate, iar pentru a nu fi interpretată ca atare trebuie prefixată cu caracterul backslash \

un cuvânt este definit ca orice literă sau cifră urmat de oricâte alte litere sau cifre (chiar si niciuna)

setul de care tine un un caracter este reprezentat cu paranteze pătrate

steluta ‘*’ reprezintă zero sau mai multe aparitii

pe partea de replacer, cele două contexte potrivite se reprezintă cu caracterul backslash \ urmat de numărul contextului.

```
biancapinghireac@vbox:~/SO/lab2$ sed -i 's/\([^ ][^ ]*\) \([^ ][^ ]*\)/\2 \1/' file.txt
biancapinghireac@vbox:~/SO/lab2$ cat file.txt
keep words on coming
and on on and ON here
onto and here also
```

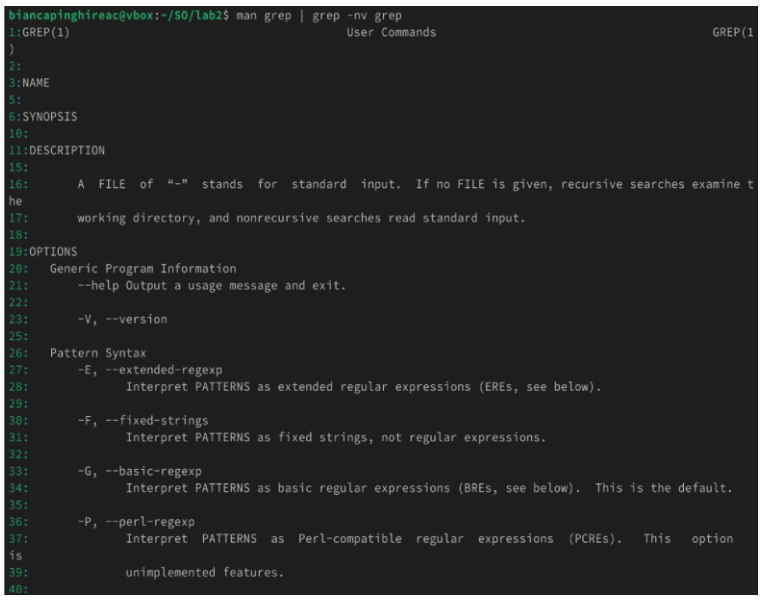
definitia mai precisă se obtine definind setul cu operatorul "orice caracter în afară de", reprezentat de ^ (denumit în engleză "caret", putere) – pentru a lua orice cuvant pana intalneste un spatiu

specificarea in-place (-i) este pentru a modifica direct in fisier

Grep: grep [optiuni] pattern [fisier]

-n – afisarea nr de linii

Cauta in manual toate liniile care contin ‘grep’



Prin -v scrie toate liniile care NU il contin pe grep

Grep poate fi făcut case-insensitive cu optiunea ‘-i’

```
biancapinghireac@vbox:~/SO/lab2$ find / -name "*.h" | sed 's/\(.*\)\/.*/\1/' | sort | uniq
find: '/boot/loader/entries': Permission denied
find: '/boot/lost+found': Permission denied
find: '/boot/efi': Permission denied
find: '/boot/grub2': Permission denied
find: '/proc/tty/driver': Permission denied
find: '/proc/1/task/1/fd': Permission denied
find: '/proc/1/task/1/fdinfo': Permission denied
find: '/proc/1/task/1/ns': Permission denied
find: '/proc/1/fd': Permission denied
find: '/proc/1/map_files': Permission denied
find: '/proc/1/fdinfo': Permission denied
find: '/proc/1/ns': Permission denied
find: '/proc/2/task/2/fd': Permission denied
find: '/proc/2/task/2/fdinfo': Permission denied
find: '/proc/2/task/2/ns': Permission denied
find: '/proc/2/fd': Permission denied
find: '/proc/2/map_files': Permission denied
find: '/proc/2/fdinfo': Permission denied
find: '/proc/2/ns': Permission denied
find: '/proc/3/task/3/fd': Permission denied
find: '/proc/3/task/3/fdinfo': Permission denied
find: '/proc/3/task/3/ns': Permission denied
```

Explicatie: cauta in root toate fisierele de tip header apoi trece la comanda sed si cauta pozitia ultimului caracter slash / si va afisa doar ceea ce se află înaintea lui

Liniile generate sunt sortate(prin ‘sort’) si se retine doar unul dintre duplicate(‘uniq’)

Comenzi in bash:

```
 GNU nano 8.1                                    cmmdc.sh
#!/bin/bash

cmmdc () {

        a=$1
        b=$2
        echo "> $a $b"

        if [ $b -eq 0 ]; then
                return $a
        fi

        c=$[ $a % $b ]
        cmmdc $b $c
        return $?
}
echo "paremetrii sunt: $@, primul: $1, al doilea: $2"

if [ $1 -lt $2 ]; then
        echo "ordine incorecta"
        exit 1
fi

cmmdc $1 $2
echo  "c.m.m.d.c. este: $?"
```

Program care realizeaza cmmdc intre 2 numere a si b

```
biancapinghireac@vbox:~/SO/lab2$ sh cmmdc.sh 273 26
paremetrii sunt: 273 26, primul: 273, al doilea: 26
> 273 26
> 26 13
> 13 0
c.m.m.d.c. este: 13
```

Rulat prin comanda ‘sh’

```
biancapinghireac@vbox:~/SO/lab2$ chmod +x cmmdc.sh
biancapinghireac@vbox:~/SO/lab2$ ./cmmdc.sh 273 26
paremetrii sunt: 273 26, primul: 273, al doilea: 26
> 273 26
> 26 13
> 13 0
c.m.m.d.c. este: 13
```

Daca adaugi permisiunea de executie poate fi rulat fara comanda ‘sh’

```
 GNU nano 8.1                                    cmmdcit.sh
#!/bin/bash

cmmdc () {
        a=$1
        b=$2

        while [ ! $b -eq 0 ]; do
                c=$[ $a % $b ]
                a=$b
                b=$c
        done

        return $a
}

if [ $1 -lt $2 ]; then
        echo "ordine incorecta"
        exit 1
fi

cmmdc $1 $2
echo "c.m.m.d.c. este: $?"
```

```
biancapinghireac@vbox:~/SO/lab2$ nano cmmdcit.sh
biancapinghireac@vbox:~/SO/lab2$ sh cmmdcit.sh 273 26
c.m.m.d.c. este: 13
```

Metoda iterativa

EXERCITIUL 2:

```
 GNU nano 8.1                                                    fi
#!/bin/bash

fibo () {
        n=$1
        a=1
        b=1
        if [ $n -eq 1 ]; then
                return $a
        fi
        if [  $n -eq 2 ]; then
                return $b
        fi
        n=$[ $n - 2 ]
        while [ ! $n -eq 0 ]; do
                c=$[ $a + $b ]
                n=$[ $n - 1 ]
                a=$b
                b=$c
        done
        return $c
}

fibo $1
echo "$?"
```

```
biancapinghireac@vbox:~/SO/lab2$ nano fibo.sh
biancapinghireac@vbox:~/SO/lab2$ sh fibo.sh 10
55
biancapinghireac@vbox:~/SO/lab2$ sh fibo.sh 5
5
biancapinghireac@vbox:~/SO/lab2$ sh fibo.sh 9
34
```

EXERCITIUL 3:

```
 GNU nano 8.1                                          directoare.sh
#!/bin/bash

readarray -t directoare < <(find / -type d | grep '\.config$' | sort | uniq )

suma=0

for director in "${directoare[@]}"; do
        marimeDirector=$(du -s "$director" | cut -f1)
        suma=$[ $suma + $marimeDirector ]
done

echo "Suma dimensiunilor tuturor directoarelor .config unice este: $suma"
```

```
Suma dimensiunilor tuturor directoarelor .config unice este: 48
```

Prima comanda creaza un array ‘directoare’ in care pune toare rezultatele cautarii;

Cautarea cauta toate fisierele de tip director (-type d) de tipul .config, le sorteaza si le tine doar pe cele unice

Trecem prin toate componentele ditrectorului similar cu limbajul de programare Python (obiect in array:)

Pentru fiecare director gasit, cautam dimensiunea acestuia cu formula(disk usage) `du -s “$director”` iar prima coloanal din rezultatul comenzii reprezinta marimea directorului (asa ca folosim comanda `cut`)

EXERCITIUL 4:

```
  GNU nano 8.1                                                  s
#!/bin/bash

touch file.txt

for i in {1..100}; do
        echo "linia $i: continut" >> file.txt
done
```

```
biancapinghireac@vbox:~/SO/lab2$ nano scriere.sh
biancapinghireac@vbox:~/SO/lab2$ sh scriere.sh
biancapinghireac@vbox:~/SO/lab2$ cat file.txt
keep words on coming
and on on and ON here
onto and here also
linia 1: continut
linia 2: continut
linia 3: continut
linia 4: continut
linia 5: continut
linia 6: continut
linia 7: continut
linia 8: continut
linia 9: continut
linia 10: continut
linia 11: continut
linia 12: continut
linia 13: continut
linia 14: continut
linia 15: continut
linia 16: continut
linia 17: continut
linia 18: continut
linia 19: continut
linia 20: continut
linia 21: continut
linia 22: continut
linia 23: continut
linia 24: continut
linia 25: continut
linia 26: continut
linia 27: continut
```

EXERCITIUL 5:

Folosim ‘/g’ pentru a ne asigura ca vor fi apelate comenzile pe toate liniile fisierului;

Folosim ‘-i’ pentru a schimba direct in fisier