Universitatea Tehnică a Moldovei

Facultatea Calculatoare Informatică şi Microelectronică

Departamentul Ingineria Software și Automatică

**RAPORT**

Lucrarea de laborator nr. 7

# La disciplina „ Sistemul de operare”

Tema: "Manipularea fișierelor în SO GNU/Linux"

A efectuat: st. gr. SI-211 S. Chirita

A verificat: V. Beșliu

Chișinău – 2023

Lucrarea de laborator nr. 7

#### Scopul lucrării: Studierea comenzilor pentru manipularea fișierelor și directoarelor, utilizarea mecanismului de legături, adresarea directă și indirectă a directoarelor.

**Sarcină pentru lucrarea de laborator:**

**1.** Script-ul rmtrash.

▪ Script-ului i se transmite un parametru - numelefișierului ce va fi creat în directorul curent (lab7).

▪ Script-ul verifică dacă a fost creat un director ascunstrash în directorul home/user ($HOME) al▪▪utilizatorului. Dacă nu este creat, îl creează.Apoi, în directorul curent, script-ul creează un fişiercu numele parametrului. Creează şi un hard link cătrefișierul creat şi directorul ascuns trash, cu un numeunic ce constă din cifre (utilizaţi (date +%s)) și ștergefișierul din directorul curent.Apoi, în fișierul ascuns trash.log din directorulhome/user al utilizatorului este plasată o linie careconține calea completă a fișierului șters și numelelink-ului hard creat.

2. Script-ul untrash.

▪ Script-ului i se transmite un parametru – numele fișierului care urmează să fie restabilit (fără caleacompletă - numai numele).

▪ Script-ul în fișierul trash.loggăsește toate înregistrările ce conțin nume de fișiere ce vor fi ca parametri pentru script și va afișa aceste nume de fișiere câte unul, cu o cerere de confirmare.

▪ Dacă utilizatorul acceptă cererea, se execută o încercare de a restabili fișierul cu calea completă specificată anterior (creați un link hard în directorul corespunzător către fișierul din trash și ștergeți fișierul corespunzător din trash). Dacă directorul specificat în calea completă către fișier nu mai există, fișierul este restabilit în directorul home/user al utilizatorului cu afișarea mesajului corespunzător.

3. Script-ul backup.

▪ Script-ul va crea în /home/user/ directorul cu numele Backup-YYYYMM-DD, unde YYYY-MM-DD – data la care a fost rulat script-ul, dacă nu există nici un director în /home/user/ cu un nume care să corespundă unei date mai mici de 7 zile față de cea curentă. Dacă în /home/user/ este deja un director activ de backup (creat nu mai devreme de 7 zile de la data lansării script-ului), atunci nu se creează un nou director. Puteți utiliza comanda date pentru a determina data curentă.

▪ Dacă a fost creat un nou director, script-ul va copia în acest director toate fișierele din directorul /home/user/source/ (pentru testarea script-ului creați un director și un set de fișiere în el). După aceasta, script-ul va scrie următoarele informații în fișierul /home/user/backup-report: o linie cu informații despre crearea noului director cu copiile de rezervă cu indicarea numelui și datei creării acestuia; lista fișierelor din /home/user/source/ care au fost copiate în acest director.

▪ Dacă directorul nu a fost creat (există un director activ de backup), scriptul trebuie să copieze toate fișierele din /home/user/source/ în acest director folosind următoarele reguli: dacă nu există nici un fișier cu acest nume în directorul de backup, el este copiat din /home/user/source. Dacă există un fișier cu acest nume, dimensiunea acestuia este comparată cu dimensiunea fișierului cu același nume din directorul de backup curent. Dacă dimensiunile sunt egale, fișierul nu este copiat. Dacă dimensiunile sunt diferite, atunci fișierul este copiat cu crearea automată a copiei versiunii, astfel încât ambele versiuni ale fișierului să apară în directorul curent de backup (fișierul existent este redenumit, adăugându-i-se o extensie suplimentară ".YYYY-MM-DD" (data la care scriptul a fost pornit), iar cel copiat își păstrează numele).

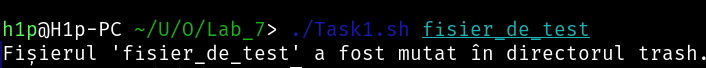
După copiere, se scrie o linie în fișierul /home/user/backup-report despre efectuarea modificărilor în directorul curent de backup, cu indicarea numelui său și data efectuării modificărilor.

4. Script-ul upback.

Scriptul trebuie să copieze în directorul /home/user/restore/ toate fișierele din directorul de backup (care are cea mai recentă dată în nume), cu excepția fișierelor cu versiunile anterioare.

**Script-ul rmtrash.**

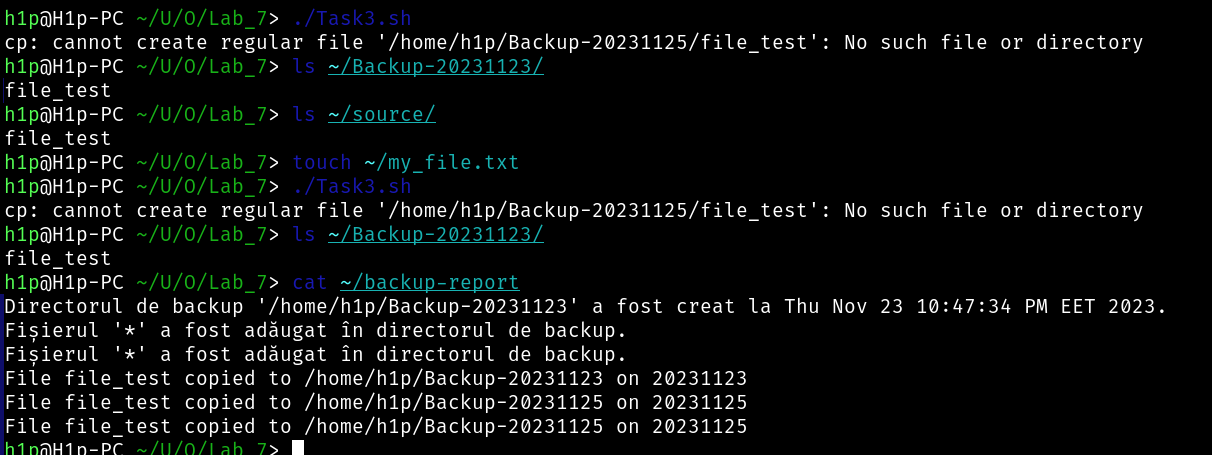
**trash\_dir="$HOME/trash"   
if [ ! -d "$trash\_dir" ]; then   
   mkdir "$trash\_dir"   
fi   
  
file\_name="$1"   
  
if [ -z "$file\_name" ]; then   
   echo "Eroare: Numele fișierului trebuie specificat."   
   exit 1   
fi   
  
if [ ! -e "$file\_name" ]; then   
   touch "$file\_name"   
fi   
  
  
link\_name="$trash\_dir/$(date +%s)"   
ln "$file\_name" "$link\_name"   
  
rm "$file\_name"   
  
echo "$(realpath "$file\_name") $link\_name" >> "$trash\_dir/trash.log"   
  
echo "Fișierul '$file\_name' a fost mutat în directorul trash."**

**Script-ul untrash.**

**trash\_dir="$HOME/trash"   
if [ ! -d "$trash\_dir" ]; then   
   echo "Eroare: Directorul trash nu există."   
   exit 1   
fi   
  
file\_name="$1"   
  
if [ -z "$file\_name" ]; then   
   echo "Eroare: Numele fișierului trebuie specificat."   
   exit 1   
fi   
  
log\_file="$trash\_dir/trash.log"   
if [ ! -f "$log\_file" ]; then   
   echo "Eroare: Fișierul trash.log nu există."   
   exit 1   
fi   
  
entries=$(grep "$file\_name" "$log\_file")   
  
if [ -z "$entries" ]; then   
   echo "Nu s-au găsit fișiere în trash.log cu numele '$file\_name'."   
   exit 0   
fi   
  
echo "Fișiere găsite în trash.log cu numele '$file\_name':"   
echo "$entries"   
  
read -p "Doriți să restaurați toate fișierele? [y/N]: " response   
case $response in   
   [yY])   
       echo "$entries" | while read -r line; do   
           source\_file=$(echo "$line" | awk '{print $1}')   
           trash\_file=$(echo "$line" | awk '{print $2}')   
              
           restore\_dir=$(dirname "$source\_file")   
           if [ ! -d "$restore\_dir" ]; then   
               echo "Directorul '$restore\_dir' nu există. Se va restaura în directorul home/user."   
               restore\_dir="$HOME"   
           fi   
  
           ln "$trash\_file" "$source\_file" && rm "$trash\_file"   
           echo "Fișierul '$source\_file' a fost restaurat cu succes în '$restore\_dir'."   
       done   
       ;;   
   \*)   
       echo "Nu s-au efectuat restaurări."   
       ;;   
esac**

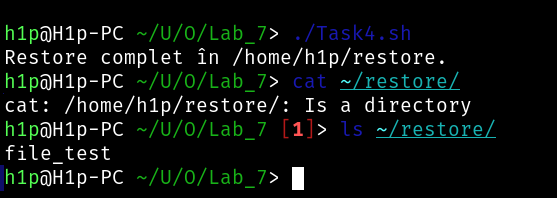
****

**backup\_date=$(date +"%Y%m%d")   
  
source\_dir="$HOME/source"   
  
backup\_dir="$HOME/Backup-$backup\_date"   
  
backup\_report="$HOME/backup-report"   
  
latest\_backup=$(find "$HOME" -maxdepth 1 -type d -name "Backup-\*" -ctime -7 -print -quit)   
  
if [ -z "$latest\_backup" ]; then   
 mkdir "$backup\_dir"   
  
 cp -r "$source\_dir"/\* "$backup\_dir"   
  
 echo "Backup directory $backup\_dir created on $backup\_date" >> "$backup\_report"   
 echo "Files copied:" >> "$backup\_report"   
 ls -l "$backup\_dir" >> "$backup\_report"   
else   
 for file in "$source\_dir"/\*; do   
   filename=$(basename "$file")   
   destination="$backup\_dir/$filename"   
      
   if [ -e "$destination" ]; then   
     size\_source=$(stat -c %s "$file")   
     size\_destination=$(stat -c %s "$destination")   
  
     if [ "$size\_source" -ne "$size\_destination" ]; then   
       mv "$destination" "$destination.$backup\_date"   
       cp "$file" "$destination"   
       echo "File $filename updated in $backup\_dir on $backup\_date" >> "$backup\_report"   
     fi   
   else   
     cp "$file" "$destination"   
     echo "File $filename copied to $backup\_dir on $backup\_date" >> "$backup\_report"   
   fi   
 done   
fi**



**Script-ul upback.**

**restore\_dir="$HOME/restore"   
  
latest\_backup=$(find "$HOME" -maxdepth 1 -type d -name "Backup-\*" | sort -r | head -n 1)   
  
if [ -z "$latest\_backup" ]; then   
 echo "Nu există niciun director de backup."   
 exit 1   
fi   
  
cp -r "$latest\_backup"/\* "$restore\_dir"   
  
find "$restore\_dir" -type f -name "\*.\*" -delete   
  
echo "Restore complet în $restore\_dir."**



**Concluzie:** În cadrul temei "Manipularea fișierelor în SO GNU/Linux", am analizat și aplicat diverse comenzi și tehnici specifice sistemului de operare. Am explorat modalități de manipulare a fișierelor, crearea, ștergerea, redenumirea și mutarea acestora. De asemenea, am investigat comenzi pentru gestionarea directoarelor, navigarea în ierarhia acestora și aplicarea de operații specifice.