 **MINISTERUL EDUCAŢIEI ȘI CERCETĂRII AL REPUBLICII MOLDOVA**

**Universitatea Tehnică a Moldovei**

**Facultatea Calculatoare, Informatică şi Microelectronică**

**Departamentul Ingineria Software și Automatică**

**Programul de studii: Tehnologia informației**

RAPORT

# LUCRARE DE LABORATOR NR. 4

# la Sisteme de Operare

**Tema:Procesarea fluxurilor textuale în SO GNU/Linux***.*

A efectuat:

st. gr. TI-211 Popa Cătălin

A verificat: Victor Colesnic

UTM, Chișinău 2023

**Tema:** Procesarea fluxurilor textuale în SO GNU/Linux.

**Scopul lucării:** Studierea procesului, intrarea și ieșirea standart a procesului, legarea procesului de I/O, utilizarea ieșirii unui proces ca parametru al unui alt proces, expresii regulate și filtrarea fluxurilor textuale.

**Sarcinile pentru lucrarea de laborator:**

Noțiuni

1. **stdin** – Este canalul de intrare standart, care permite unui program să primească date de la utilizator sau din alte surse. De obicei, acest canal este conectat la tastatură sau la un alt program care furnizează date.

2. **stdout** – Este canalul de ieșire standart, utilizat pentru a afișa rezultatele programului. De obicei, acest canal este conectat la ecran sau la un alt program care preia datele.

3. **stderr** – Este canalul de eroare standart, utilizat pentru a afișa mesaje de eroare. Acest canal este, de asemenea, conectat la ecran sau la un alt program care preia mesaje de eroare.

Instrucțiuni

1. **sort** – Utilizată pentru a sorta liniile unui fișier sau a unei intrări standart într-un anumit mod.

2. **uniq** – Elimină liniile dublicate dintr-un fișier care a fost sortat.

3. **cut** – Este folosit pentru a extrage câmpuri specifice dintr-un fișier text. Prin câmp, înțelegem secvență de simboluri până la spațiu.

4. **head** – Afișeză primele linii dintr-un fișier la stdout.

5. **tail** – Afișeză ultimele linii dintr-un fișier la stdout.

6. **tr** – înlocuiește unele simboluri cu altele.

Destinația utilitarelor

1. **grep** - Utilară pentru căutarea și sortarea liniilor introduse și afișarea anumitor tipare(șiruri de caractere) ca rezultat al procesării.

2. **sed** – (Stream Editor) dispozitivul stdin sau dintr-un fișier text, execută unele operații asupra liniilor după care afișează rezultatul pe stdout sau într-un fișier. Sed determină, reieșind din spațiul de adrese dat, liniile asupra cărora vor fi executate operațiile. Spațiul de adresă al liniilor este desemnat de numărul de ordine al acestora sau de șablon. De exemplu, comanda 3d va obliga sed să elimine linia a treia, iar instrucțiunea /windows/d va conduce la eliminarea tuturor liniilor, care conțin cuvântul "**windows**". Cel mai frecvent sunt utilizate comenzile: **p** – imprimare (pe stdout), **d** – eliminare/ștergere și **s** – înlocuire.

3. **awk** – Comanda awk – utilitară pentru căutarea contextuală și transformare a textului, instrument pentru extragerea și/sau procesarea câmpurilor (coloanelor) în fișiere text structurate. Awk împarte fiecare linie în câmpuri individuale. În mod implicit, câmpurile sunt secvențe de caractere separate prin spații, dar există posibilitatea de a desemna alt caracter ca separator de câmp. Awk analizează și procesează fiecare câmp aparte.

**Clase de caractere *POSIX****.*

Clasele de caractere POSIX sunt utilizate în expresiile regulate pentru a defini grupuri de caractere cu anumite proprietăți comune. Acestea sunt folosite cu instrumente precum ***grep*** și ***sed***. Exemple de clase de caractere POSIX includ:

− [:class:] metodă alternativă de a specifica un diapazon de caractere.

− [:alnum:] corespunde caracterelor alfanumerice. Este echivalentă expresiei [A-Za-z0-9].

− [:alpha:] corespunde caracterelor alfabetului. Este echivalentă expresiei [A-Za-z].

− [:blank:] corespunde caracterului spațiu sau tabulare.

− [:cntrl:] corespunde caracterelor de control.

− [:digit:] corespunde setului de cifre zecimale. Este echivalentă expresiei [0-9].

− [:lower:] corespunde caracterelor alfabetului din registrul inferior. Echivalentă expresiei [a-z].

− [:space:] corespunde caracterelor de spațiere (spațiu și tabulare orizontală).

− [:graf:] corespunde unui set de caractere din gama ASCII 33 - 126.

− [:print:] corespunde unui set de caractere din gama ASCII 32 – 126.

− [:upper:] corespunde caracterelor alfabetului din registrul superior. Echivalentă expresiei [A-Z].

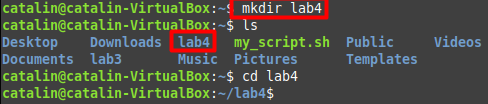
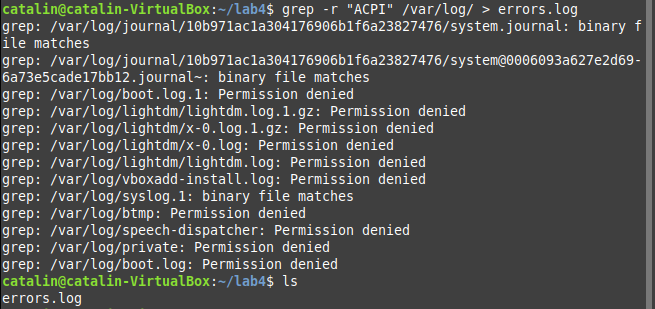
− [:xdigit:] corespunde setului de cifre hexazecimale. Este echivalentă expresiei [0-9A-Fa-f].

Aceste noțiuni și instrumente sunt fundamentale în lucrul cu sistemul de operare GNU/Linux și în prelucrarea textului în mediul Unix.

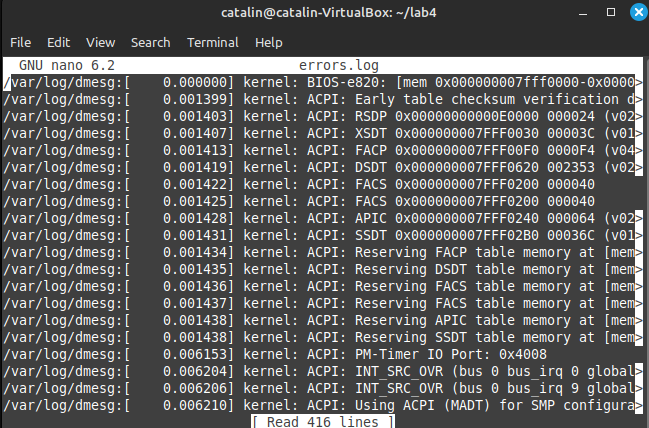
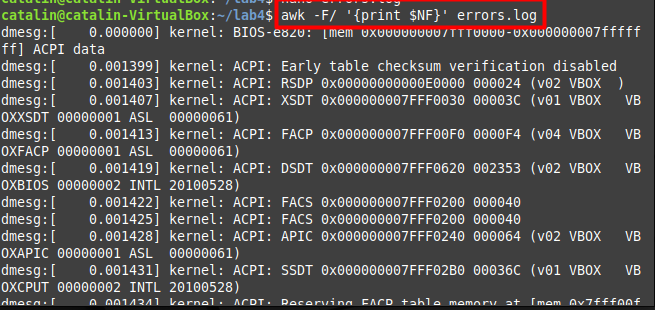
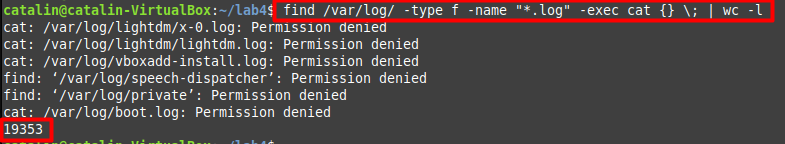
Comanda "uniq -c" în limbajul Unix/Linux este folosită pentru a număra aparițiile consecutive ale liniilor într-un fișier sau flux de date și pentru a afișa numărul de apariții al fiecărei linii alături de linia respectivă.

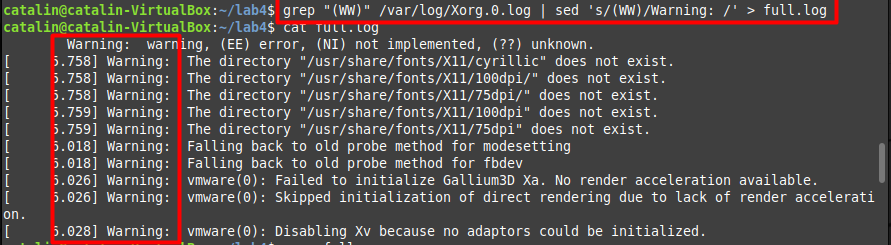
Mai precis:"uniq" este comanda principală, care elimină liniile duplicate consecutive dintr-un fișier sau flux de intrare.

"-c" este opțiunea care cere comandei să afișeze, înaintea fiecărei linii un număr care reprezintă numărul de apariții consecutive ale liniei respective.

1. a) Creați un nou subdirector unde veți plasa toate script-urile și fișierele pentru extragerea rezultatelor.
2. 
3. **Figura 1** – Creare directoriu nou.
4. b) Scrieți script-uri, care vor rezolva următoarele probleme:
5. **1)** Creați fișierul **errors.log**, în care veți plasa toate liniile din toate fișierele directorului **/var/log/** accesibile pentru citire, care conțin secvența de caractere **ACPI**. Afișați pe ecran liniile din **errors.log**, care conțin numele fișierelor, excluzând calea acestor fișiere.
6. **grep** – caută șiruri de caractere **-r** – căutarea trebuie să fie recursivă. **/var/log/** – fișiere de jurnal
7. **>** - operatorul de redirecționare. Ia ieșirea standart a comenzii **(grep -r "ACPI" /var/log/)** și o redirecționează către fișierul errors.log. Dacă fișierul există, conținutul va fi suprascris.
8. 
   1. **Figura 2** – Creare fisier errors.log.

Aceasta va căuta recursiv în toate fișierele din directoriul /var/log/ și va adăuga liniile care conțin ACPI în fișierul errors.log.

* + 1. 
    2. **Figura 3** – Afisare fisier errors.log.
    3. 
    4. **Figura 4** – Afisare liniile din errors.log fara calea acestora.
    5. **/** - separator **$NF** – numele fișierului fără calea completă.
    6. **awk** – este comandă AWK, care este un limbaj de programare și un utilitar de procesare a textului în linia de c omandă în sistemele Unix și Linux.
    7. **-F/** - Este opțiunea AWK care specifică un separator de câmpuri. În acest caz, separatorul este "/", ceea ce înseamnă că AWK va trata câmpurile din linia de intrare separate de caracterele "**/**".
    8. **{print $NF}** - Aceasta este expresia AWK care specifică acțiunile pe care AWK trebuie să le efectueze pe fiecare linie de intrare. În acest caz, print $NF înseamnă să imprime (afișeze) ultimul câmp din linie ($NF este variabila AWK care reprezintă ultimul câmp).
    9. **2)** Calculați numărul total de linii în fișierele directorului **/var/log/** cu extensia **log**. Afișați acest număr pe ecran.
    10. 
    11. **Figura 5** – Afisare numar de linii.
    12. **find /var/log/ -type f -name "\*.log"** - Această parte a comenzii folosește comanda find pentru a găsi toate fișierele (-type f) din directorul /var/log/ care au extensia .log (-name "\*.log").
    13. **-exec cat {} \;** - Pentru fiecare fișier găsit, se utilizează -exec pentru a executa comanda cat pentru a afișa conținutul fișierului. {} este înlocuit cu numele fiecărui fișier găsit.
    14. **|** - Operatorul pipe (|) este folosit pentru a direcționa ieșirea comenzii find către intrarea comenzii următoare (cat în acest caz).
    15. **wc -l** - Comanda wc (word count) este utilizată pentru a număra liniile (-l) din intrarea primită prin pipe.

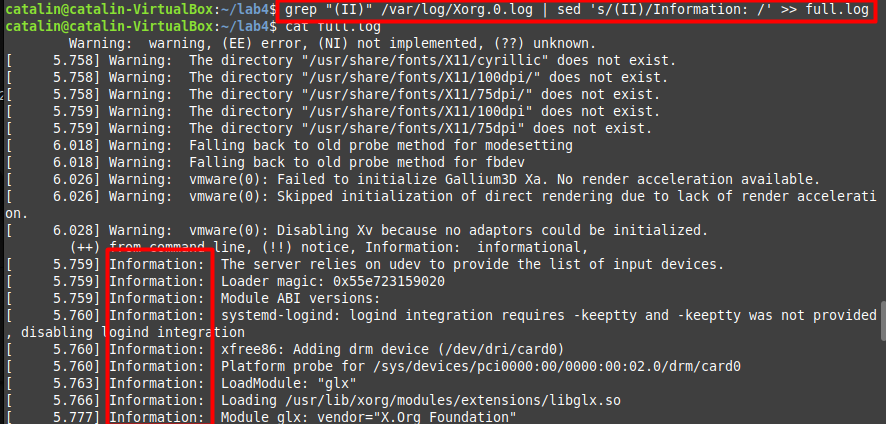
1. **3)** Creați fișierul **full.log**, în care veți plasa liniile din fișierul **/var/log/Xorg.0.log**, care conțin avertizări și mesaje informaționale, înlocuind marcherii avertizărilor (**WW**) și mesajelor informaționale (**II**) cu cuvintele **Warning**: și **Information:**, astfel încât în fișierul rezultat mai întâi să apară toate mesajele informaționale, iar apoi toate avertizările. Afișați conținutul fișierului **full.log** pe ecran.
2. 
3. **Figura 6** – Creare fisier full.log cu cuvintele Warning.

- **grep "(WW)" /var/log/Xorg.0.log** - Utilizează `grep` pentru a căuta liniile din fișierul `/var/log/Xorg.0.log` care conțin marcajul "(WW)", care indică avertizări (warnings).

- **|** - Pipe operator, redirecționează ieșirea comenzii anterioare către comanda următoare.

- **sed 's/(WW)/Warning: /`**- Utilizează `sed` pentru a înlocui toate aparițiile "(WW)" cu "Warning: " în liniile găsite anterior.

- **>** - Operator de redirecționare, scrie rezultatul într-un fișier nou sau suprascrie un fișier existent (în acest caz, creează sau suprascrie `full.log`).

1. 
2. **Figura 7** – Creare fisier full.log cu cuvintele Information.

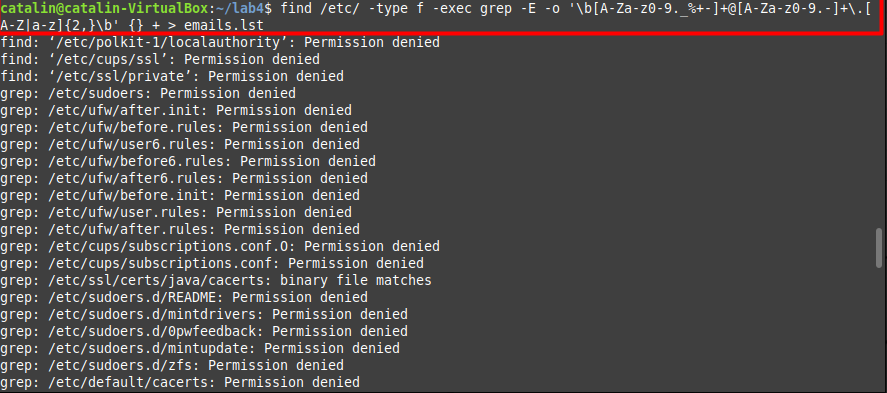
- **grep "(II)" /var/log/Xorg.0.log** - Utilizează `grep` pentru a căuta liniile din fișierul `/var/log/Xorg.0.log` care conțin marcajul "(II)", care indică mesaje informaționale.

- **|** - Pipe operator, redirecționează ieșirea comenzii anterioare către comanda următoare.

- **sed 's/(II)/Information: /**' - Utilizează `sed` pentru a înlocui toate aparițiile "(II)" cu "Information: " în liniile găsite anterior.

- **>>** - Operator de redirecționare, adaugă rezultatul la sfârșitul unui fișier existent sau creează un fișier nou (în acest caz, adaugă la `full.log`).

1. **4)** Creați fișierul **emails.lst**, în care veți plasa toate adresele de poștă electronică, depistate în fișierele directorului **/etc**. Adresele vor urma consecutiv, separate prin virgulă. Afișați conținutul fișierului **emails.lst** pe ecran.

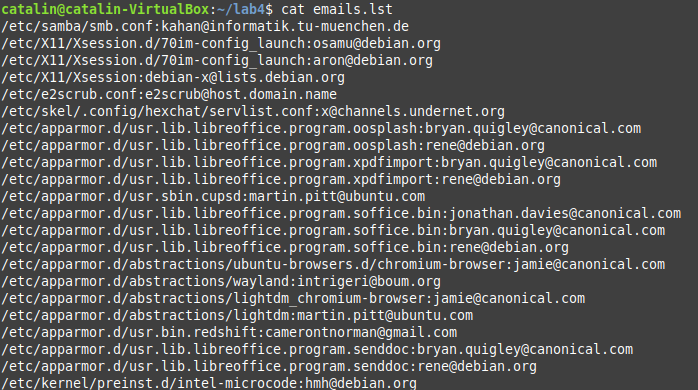


**Figura 8** – Adresele de poștă electronică.

**find /etc/ -type f** - Găsește toate fișierele din directorul /etc/.-

**exec grep -E -o '\b[A-Za-z0-9.\_%+-]+@[A-Za-z0-9.-]+\.[A-Z|a-z]{2,}\b' {} +** - Utilizează grep pentru a căuta adrese de e-mail în fișierele din directorul /etc/. Opțiunea -E permite utilizarea expresiilor regulate extinse, iar opțiunea -o determină grep să afișeze doar textul care se potrivește cu expresia regulată.Expresia regulată ('\b[A-Za-z0-9.\_%+-]+@[A-Za-z0-9.-]+\.[A-Z|a-z]{2,}\b') este o expresie simplificată pentru a identifica adresele de e-mail. Aceasta se potrivește cu șiruri care încep cu caractere alfanumerice, pot conține caractere speciale cum ar fi punctul, sublinierea, procentul, plusul și minusul, urmate de simbolul '@', și apoi un domeniu care începe cu caractere alfanumerice, poate conține caractere speciale și este urmat de un punct, urmat de cel puțin două caractere alfabetice.

* - emails.lst: Redirectează rezultatele căutării în fișierul emails.lst.



**Figura 9** – Afișare emailuri.

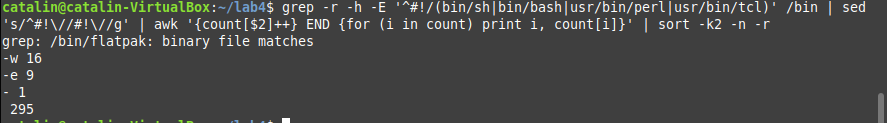
1. **5)** Găsiți în directorul **/bin** toate fișierele, ce reprezintă script-uri și afișați pe ecran numele interpretoarelor cu numărul ce indică frecvența lor de utilizare (în script-uri). Script-urile pot începe cu una din următoarele linii:

#!/bin/sh

#!/bin/bash

#!/usr/bin/perl

#!/usr/bin/tcl.



**Figura 10** - Găsire scripturi.

**grep -r -h -E '^#!/(bin/sh|bin/bash|usr/bin/perl|usr/bin/tcl)' /bin** - Folosește grep pentru a căuta recursiv în directorul /bin toate fișierele care încep cu una dintre liniile specificate.

-r: Căutare recursivă.

-h: Afișează doar numele fișierelor, fără a afișa numele fișierului când se caută în mai multe fișiere.

-E: Activează expresiile regulate extinse.

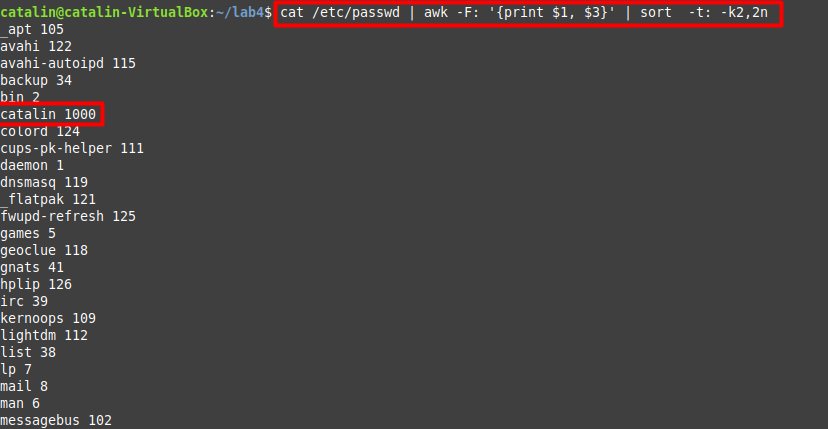
^#!/(bin/sh|bin/bash|usr/bin/perl|usr/bin/tcl): Specifică căutarea pentru liniile care încep cu una dintre linii.

**sed 's/^#!\//#!\//g**’ - Utilizează sed pentru a ajusta formatul liniei #! în cazul în care apare sub forma #!/bin/sh, etc.

**awk '{count[$2]++} END {for (i in count) print i, count[i]}'** - Utilizează awk pentru a număra și afișa frecvența utilizării pentru fiecare interpretor.

**sort -k2 -n -r** - Utilizează sort pentru a sorta rezultatele după numărul de utilizări în ordine descrescătoare.

**6)** Afișați lista utilizatorilor sistemului cu indicarea **UID** al fiecăruia, cu sortarea după **UID**. Informațiile despre utilizatori sunt păstrate în fișierul **/etc/passwd**. În fiecare linie a acestui fișier primul câmp conține numele utilizatorului, iar câmpul al treilea - **UID,** separatorul – două puncte (**:**).



**Figura 11** – Afișare utilizatori.

**cat /etc/passwd** - Afisează conținutul fișierului /etc/passwd, care conține informații despre utilizatori.

**awk -F: '{print $1, $3}'** - Utilizează awk pentru a extrage numele utilizatorului (primul câmp, $1) și UID-ul (al treilea câmp, $3) din fiecare linie a fișierului /etc/passwd. Separatorul specificat cu -F: indică că separatorul de câmpuri este două puncte (:).

**sort -t: -k2,2n** - Utilizează sort pentru a sorta rezultatele după al doilea câmp (UID). Opțiunea -t: specifică că separatorul de câmpuri este două puncte, iar -k2,2n indică să sortăm după al doilea câmp numeric în ordine crescătoare.

**7)** Afișați 5 cel mai frecvent întâlnite cuvinte din **man bash** cu lungimea de cel puțin 3 caractere.

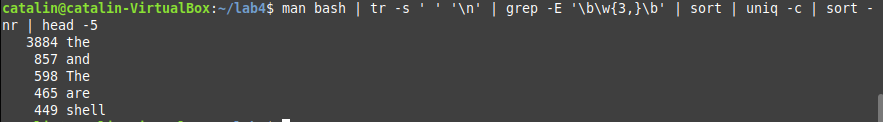


Figura 12 – Afișare 5 cel mai frecvent cuvinte intâlnite.

**man bash** - Afișează manualul pentru shell-ul Bash.

**tr -s ' ' '\n'** - Transformă spațiile în newline-uri, astfel încât fiecare cuvânt să fie pe o nouă linie.

**grep -E '\b\w{3,}\b'** - Folosește grep pentru a găsi cuvinte cu lungimea de cel puțin 3 caractere. Expresia regulată **\b\w{3,}\b** asigură că cuvintele sunt separate de delimitatori de cuvinte și au cel puțin 3 caractere.

**sort** - Sortează cuvintele în ordine lexicografică.

**uniq -c** - Elimină liniile duplicate și numără câte o dată apare fiecare cuvânt.

**sort -nr** - Sortează rezultatele în ordine descrescătoare după numărul de apariții.

**head -5** - Afișează primele 5 linii, adică cele 5 cele mai frecvente cuvinte.