**Ministerul Educaţiei și Cercetării al Republicii Moldova**

**Universitatea Tehnică a Moldovei**

**Facultatea Calculatoare, Informatică și Microelectronică**

**RAPORT**

Lucrarea de laborator nr.

*la Sisteme de operare*

**Au efectuat: Vlasitchi Stefan**

**st. gr. TI-216**

**A verificat:**

**asist. univ.**

**Chişinău - 2023**

**Lucrare de laborator nr. 4**

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

**Sarcini practice:**

## Efectuarea sarcinilor

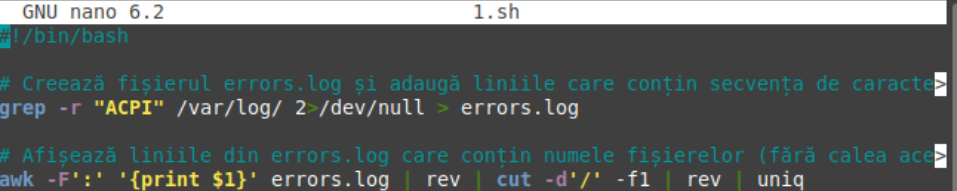
**Sarcina 1**

Figura 1. Codul scriptului 1.sh

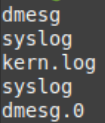
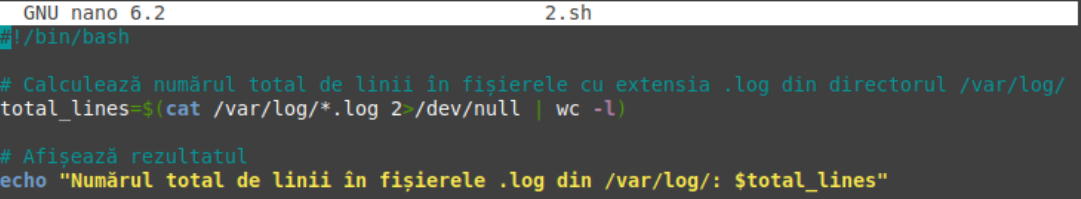


Figura 2. Rezultatul scriptului 1.sh

**Sarcina 2**

****Figura 3. Codul scriptului 2.sh

Figura 4. Rezultatul scriptului 2.sh

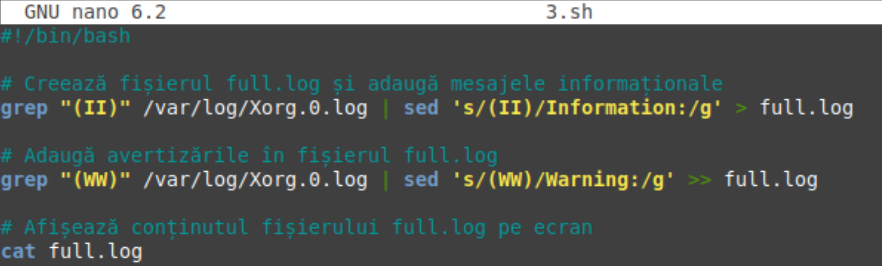
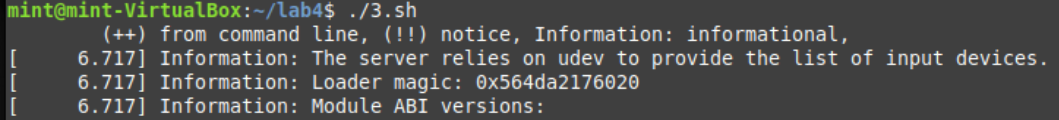
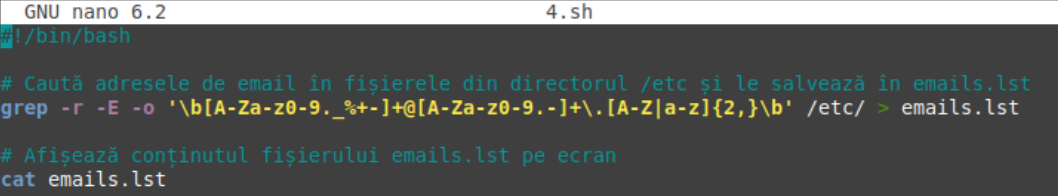
**Sarcina 3**

Figura 5. Codul scriptului 3.sh

****Figura 6. Rezultatul scriptului 3.sh

**Sarcina 4**

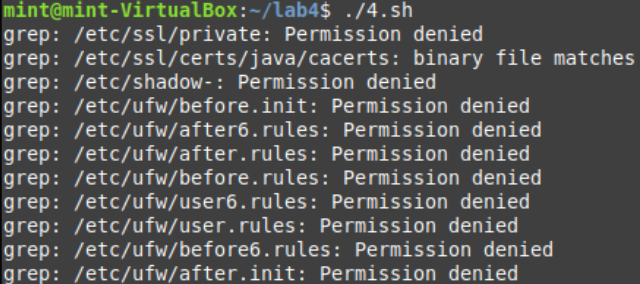
Figura 7. Codul scriptului 4.sh

Figura 8. Rezultatul scriptului 4.sh

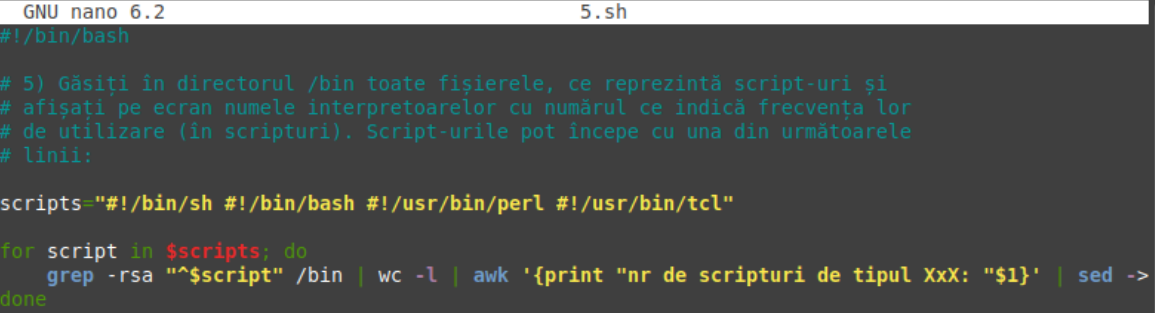
**Sarcina 5**

Figura 9. Codul scriptului 5.sh

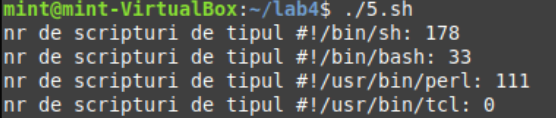


Figura 10. Rezultatul scriptului 5.sh

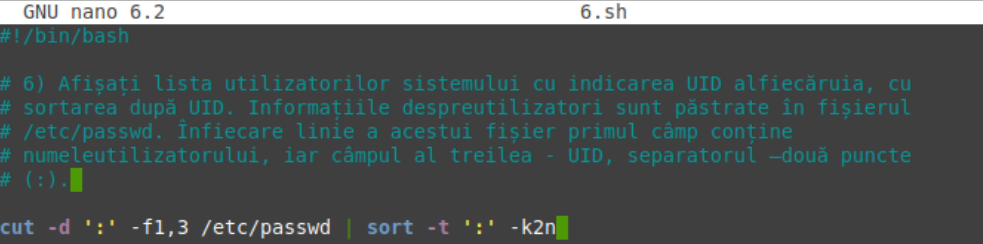
**Sarcina 6**

Figura 11. Codul scriptului 6.sh

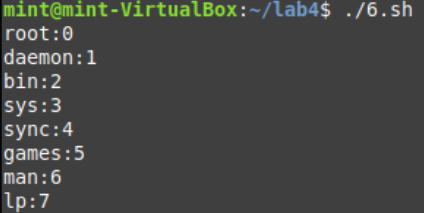


Figura 12. Rezultatul scriptului 6.sh

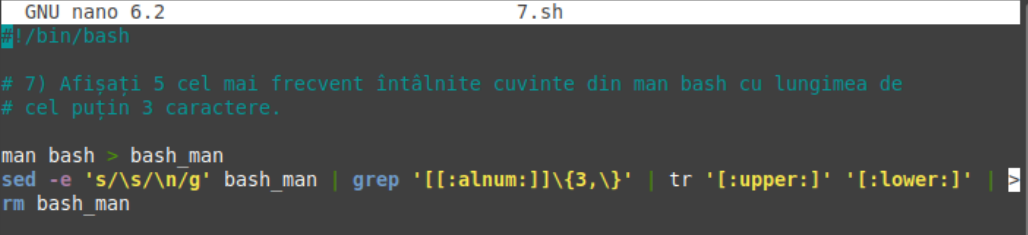
**Sarcina 7**

Figura 13. Codul scriptului 7.sh

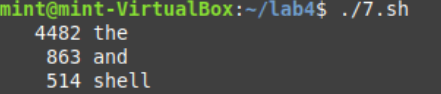


Figura 2. Rezultatul scriptului 7.sh

**Concluzie**

În această lucrare de laborator, s-au explorat și implementat diverse comenzi și scripturi în mediul Linux, având ca scop dezvoltarea abilităților practice în gestionarea sistemului de operare. S-a creat un subdirector nou pentru organizarea scripturilor și fișierelor relevante, facilitând astfel extragerea și analiza datelor din diverse surse. Prin rezolvarea problemelor specifice, cum ar fi identificarea liniilor din fișierele de log ce conțin secvența de caractere "ACPI" sau crearea unui fișier `emails.lst` ce conține adresele de poștă electronică din directorul `/etc`, am dobândit o înțelegere mai profundă a comenzilor și utilitarelor disponibile în mediul Linux. De asemenea, prin manipularea conținutului fișierului `full.log`, în care au fost înlocuite marcajele avertizărilor și mesajelor informaționale conform cerințelor, s-a evidențiat flexibilitatea și puterea scripturilor în automatizarea sarcinilor repetitive. Prin analiza și sortarea utilizatorilor sistemului după UID, am consolidat abilitățile în manipularea datelor din fișiere, oferind o perspectivă practică asupra modului în care informațiile sunt organizate în sistemul de operare. În final, prin identificarea celor mai frecvent întâlnite cuvinte din manualul Bash, am consolidat cunoștințele legate de analiza textului și utilizarea filtrelor pentru extragerea informațiilor relevante.

Aici se folosește comanda **grep** pentru a căuta în fișierele din directorul /etc expresii regulate care corespund unor adrese de e-mail.

În comanda `**grep** -**IEiorhs** '([[:alnum:]\_.-]+@[[:alnum:]\_.-]+?\.[[:alpha:].]{2,6})' /etc`, opțiunile și argumentele specificate sunt:

- `-I`: Ignoră fișierele binare.

- `-E`: Folosește expresii regulate extinse în loc de expresii regulate simple.

- `-i`: Face ca potrivirile să fie nesensibile la majuscule și minuscule.

- `-o`: Afișează doar potrivirile în loc de întreaga linie.

- `-r`: Caută recursiv în subdirectoare.

- `-h`: Afișează numele fișierului doar pentru potrivirile în mai mult de un fișier.

- `-s`: Suprimă erorile legate de fișierele care nu pot fi citite (de exemplu, fișiere protejate).

Împreună, aceste opțiuni și argumente configurează `grep` să:

- Ignore fișierele binare.

- Folosească expresii regulate extinse.

- Fie nesensibil la majuscule și minuscule.

- Afișeze doar potrivirile (adresele de e-mail) și nu întreaga linie.

- Caută recursiv în subdirectoare.

- Afișeze numele fișierului doar pentru potrivirile în mai mult de un fișier.

- Suprime eventualele erori legate d e fișierele care nu pot fi citite.

Aceste opțiuni și argumente sunt folosite pentru a face `grep` să efectueze căutarea în modul dorit în întregul director `/etc` și subdirectoarele acestuia pentru potrivirile expresiei regulate date, adică adrese de e-mail.

Expresia regulată `([[:alnum:]\_.-]+@[[:alnum:]\_.-]+?\.[[:alpha:].]{2,6})` este folosită pentru a potrivi adresele de e-mail într-un text. Vom analiza semnificația fiecărui caracter din această expresie regulată:

1. `(` și `)`: Delimitează un grup de potrivire. Orice potrivire găsită în acest grup va fi inclusă în rezultatul final.

2. `[[:alnum:]\_.-]+`: Aceasta potrivește partea locală a adresei de e-mail. În detaliu:

- `[[:alnum:]]`: Potrivește orice caracter alfanumeric (litere și cifre).

- `\_.-`: Potrivește caracterele underscore (`\_`), punct (`.`) și cratimă (`-`).

- `+`: Indică că trebuie să existe cel puțin un astfel de caracter înaintea caracterului `@`.

3. `@`: Acesta potrivește caracterul literal `@`, care separă partea locală a adresei de domeniul de e-mail.

4. `[[:alnum:]\_.-]+?`: Acesta potrivește domeniul adresei de e-mail. Este similar cu primul grup, dar și aici se permite caracterelor underscore, punct și cratimă. Caracterul `+` indică că trebuie să existe cel puțin un astfel de caracter înainte de caracterul `.` care urmează.

5. `\.`: Acesta potrivește literal caracterul punct (`.`), care separă domeniul generic de extensie (de exemplu, "com" în "example.com").

6. `[[:alpha:].]{2,6}`: Acest grup potrivește extensia de domeniu, care trebuie să conțină cel puțin 2 și cel mult 6 caractere alfa (litere) sau puncte.

7. `)`: Marchează sfârșitul grupului de potrivire.

În ansamblu, această expresie regulată caută adrese de e-mail într-un text, luând în considerare diverse caractere și formate posibile pentru partea locală și domeniu. Este important de menționat că această expresie regulată poate identifica multe adrese de e-mail valide, dar poate raporta și adrese invalide sau potrivi alt tip de texte care seamănă cu adrese de e-mail.

sed -i 's/$/,/' $FILE

sed -i 's/$/,/' $FILE: Utilizează sed pentru a înlocui (s/old/new/) fiecare sfârșit de linie ($) cu o virgulă. Astfel, se adaugă o virgulă la sfârșitul fiecărei linii.

bashCopy code

sed -i '$s/,$//' $FILE

sed -i '$s/,$//' $FILE: Acest comandament elimină virgula adăugată anterior doar de pe ultima linie. $s/,$// înseamnă să înlocuiască (s/old/new/) virgula de la sfârșitul liniei ($) cu nimic (adică să o șteargă).