

**UNIVERSITETI POLITEKNIK I TIRANËS**

**FAKULTETI I TEKNOLOGJISË DHE INFORMACIONIT**

**DEPARTAMENTI I INXHINIERISË INFORMATIKE**

**Punë Laboratori nr. 1**

**Lënda:** Sisteme Operative

**Grupi:** III-B

**Tema:** Instalimi i sistemit operativ *Linux Ubuntu* (në VirtualBox). Ambjentimi me komandat kryesore të *Linux-it* në terminal.

**Punoi:**  **Pranoi**:

Piro Gjikdhima MSc.Megi Tartari

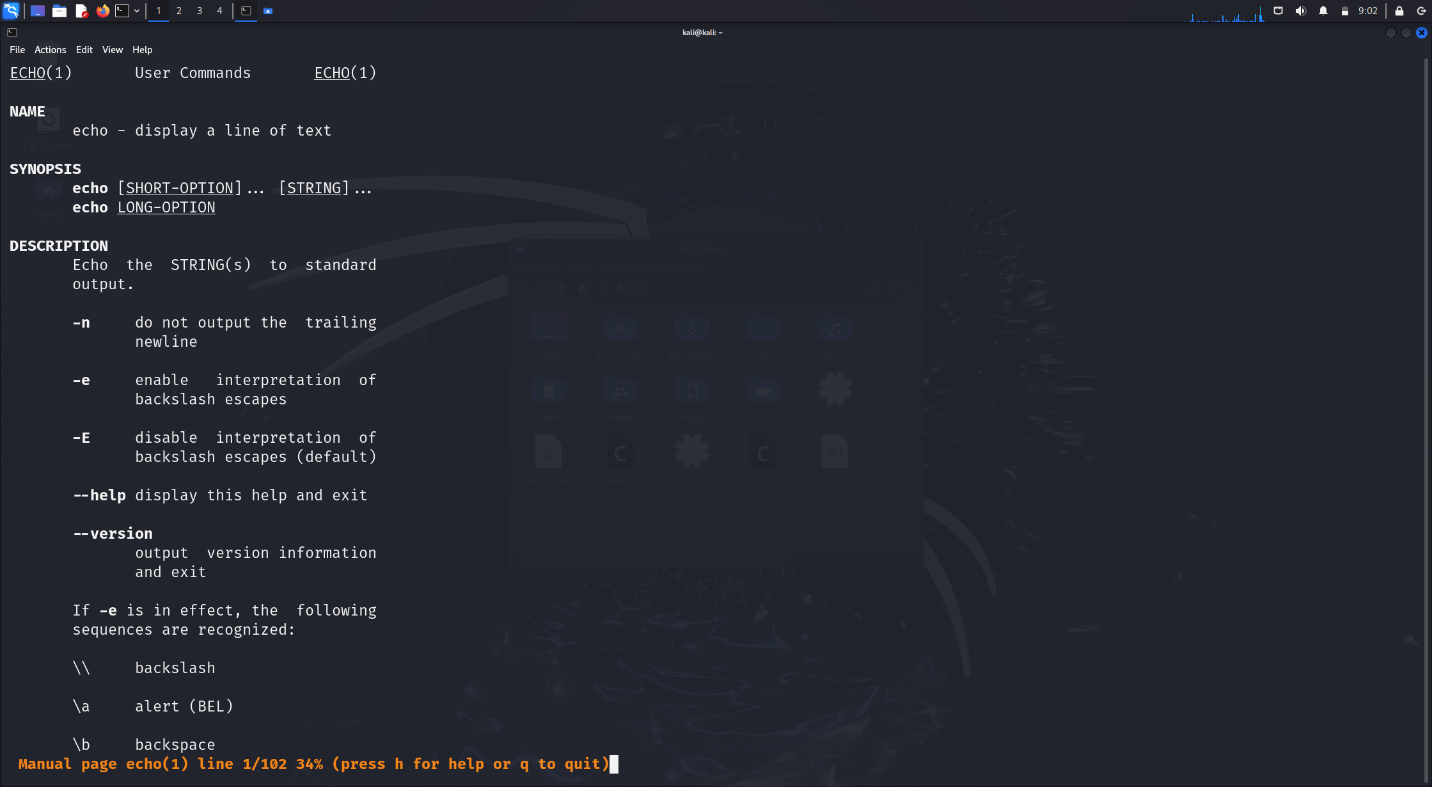
**Teori:**

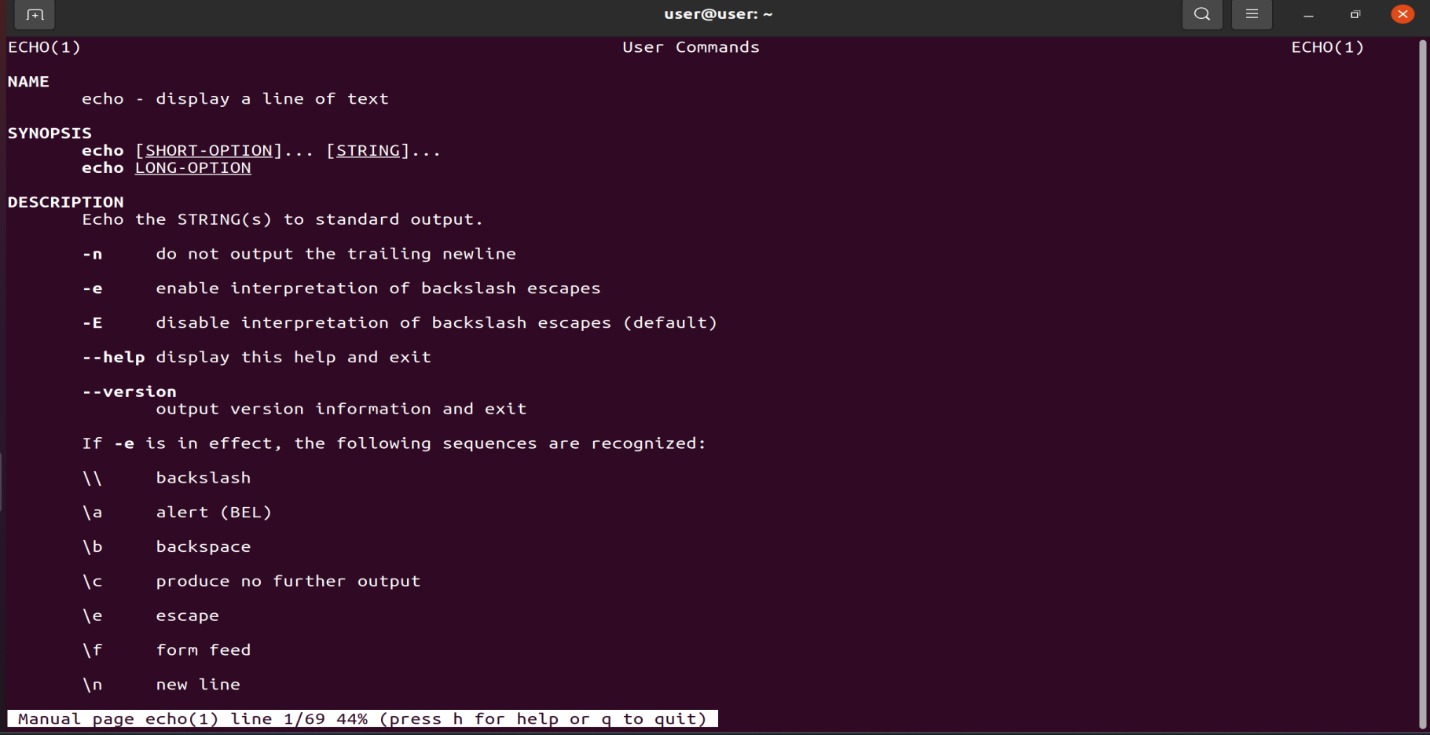
**Pika 1**: Hap terminal në *Linux* dhe shtyp komandat më poshtë:

* ***$ echo Hello World***
* ***$ man echo*** (***man*** online manual për komandat)
* ***$ echo $ SHELL*** (printon shell-in që po përdor)

A screenshot of a computer

Description automatically generated





Studio komandat në vijim referuar dokumentit *unix\_tutorial.pdf* si dhe anë të komandës ***man***:

***man, who, cat, cd, cp, ps, ls, mv, rm, mkdir, rmdir, echo, more, date, time, kill, history, chmod, chown, finger, pwd, cal, logout, shutdown***.

* ***man-*** an interface of the system reference manuals
* ***who-*** show who is logged on
* ***cat-*** concatenate files and print on the standard output
* ***cd-*** offers several ways to navigate and change the working directory using the terminal window
* ***cp-*** copy files and directories
* ***ps-*** raport a snapshop of the current process
* ***ls-*** list directory content
* ***mv-*** move (rename) files
* ***rm-*** remove files or directories
* ***mkdir-*** make directories
* ***rmdir-*** remove directories
* ***echo-*** display a line of text
* ***more-*** file perusal filter for crt viewing
* ***date-*** print or set the system date and time
* ***time-*** run programs and summarize system resource usage
* ***kill-*** send a signal to a process
* ***history-*** GNU history library
* ***chmod-*** change file mode bits
* ***chown-*** change file owner and group
* ***finger-*** gives details of all the users who are logged in
* ***pwd-*** print name of current/working directory
* ***cal-*** displays a calendar and the date of Easter
* ***logout-*** write utmp and wtmp entries
* ***shutdown-*** Halt, poweroff or reboot the machine
* ***grep-*** print line that matches patterns

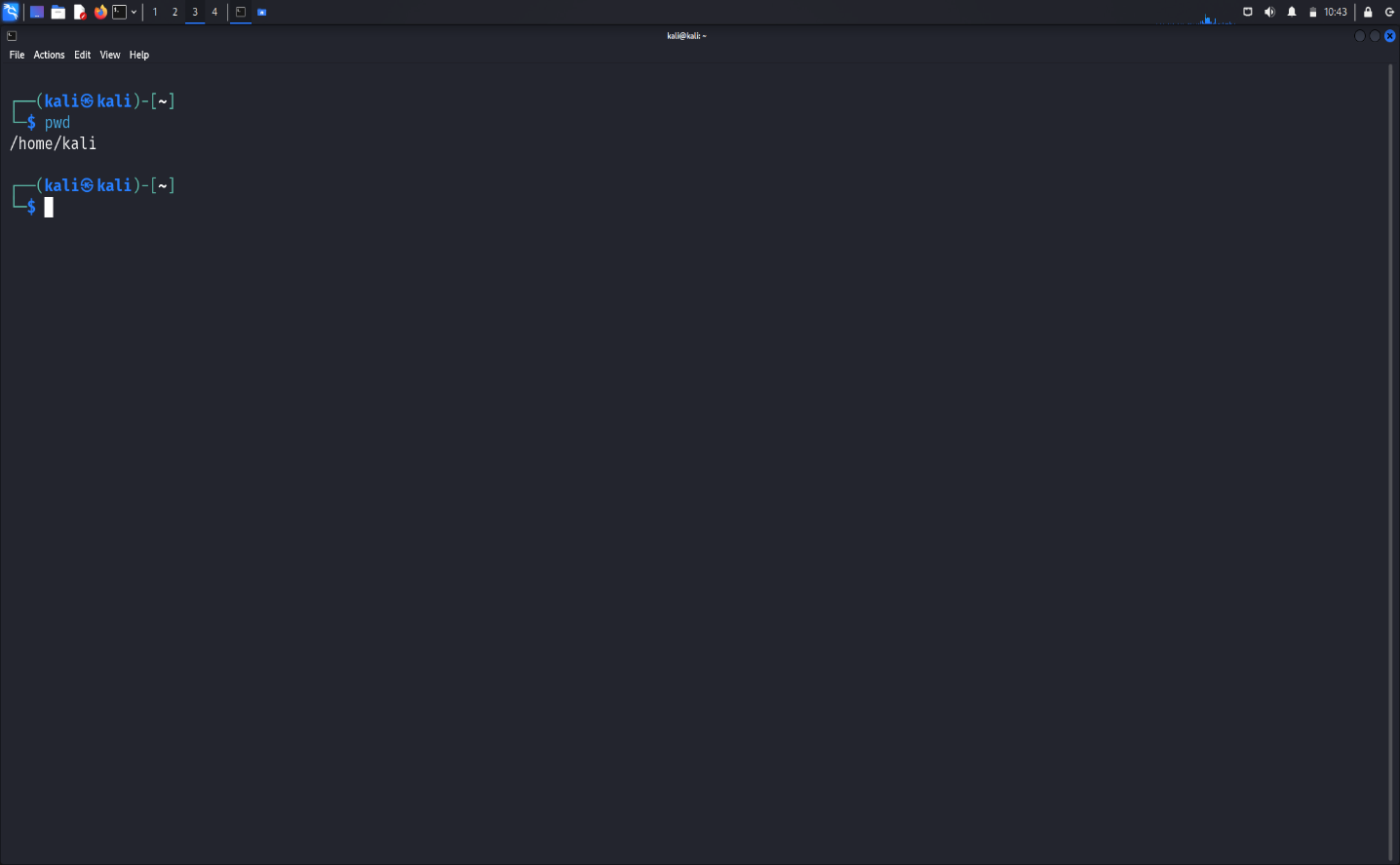
Studio komandën ***grep*** dhe përdore për të kërkuar sa rreshta të një file mbajnë një fjalë kyçe.

Lexoni ose vetem shfaqni permbajtjen e file:cat

A screen shot of a computer

Description automatically generated

Verifikoni ne cilen direkto jeni aktualisht:pwd

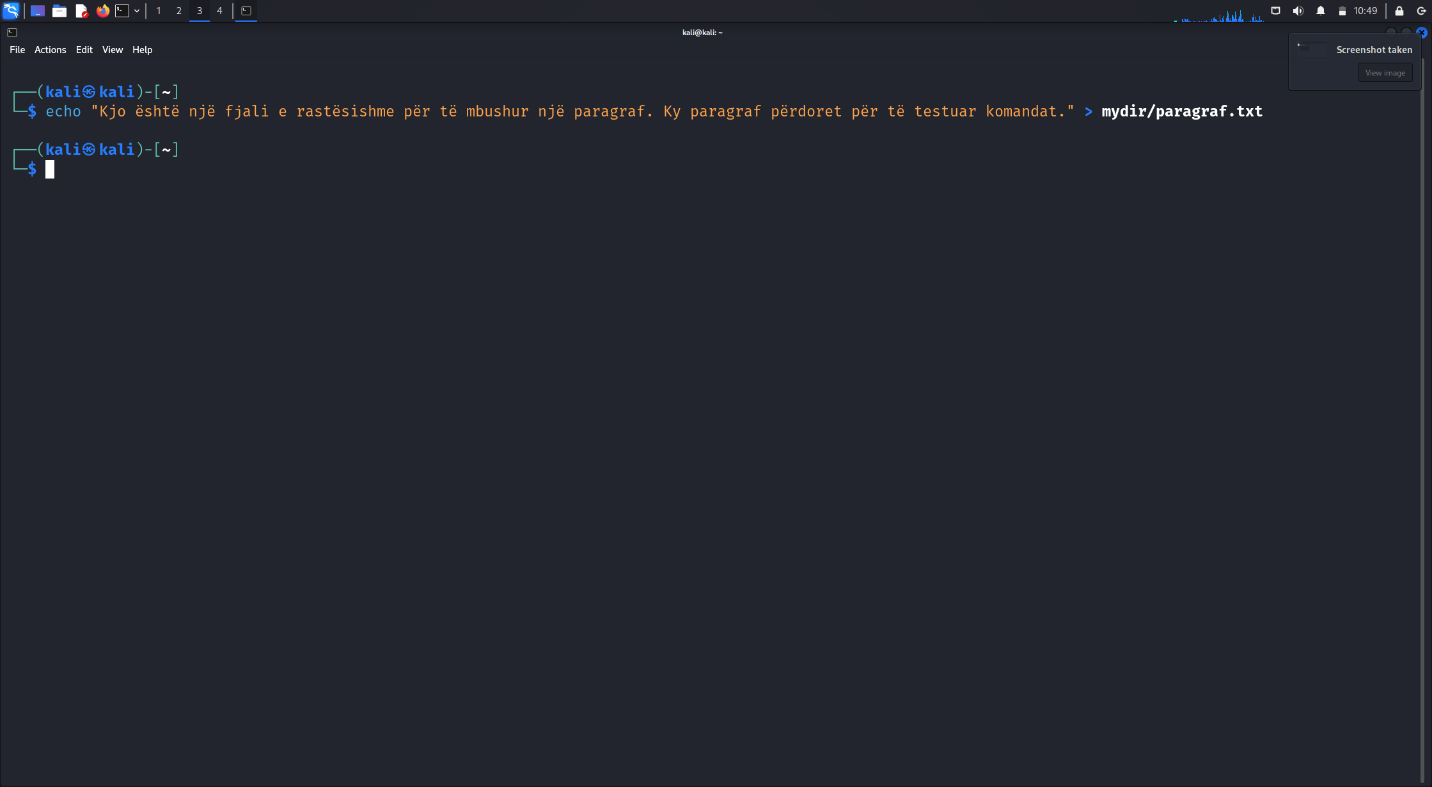


Shkoni tek home directory: cd ~ dhe krijoni nje nendirektori.

A black screen with white text

Description automatically generated

Krijoni nje text file me nje paragraph cfaredo.

A screenshot of a computer

Description automatically generated

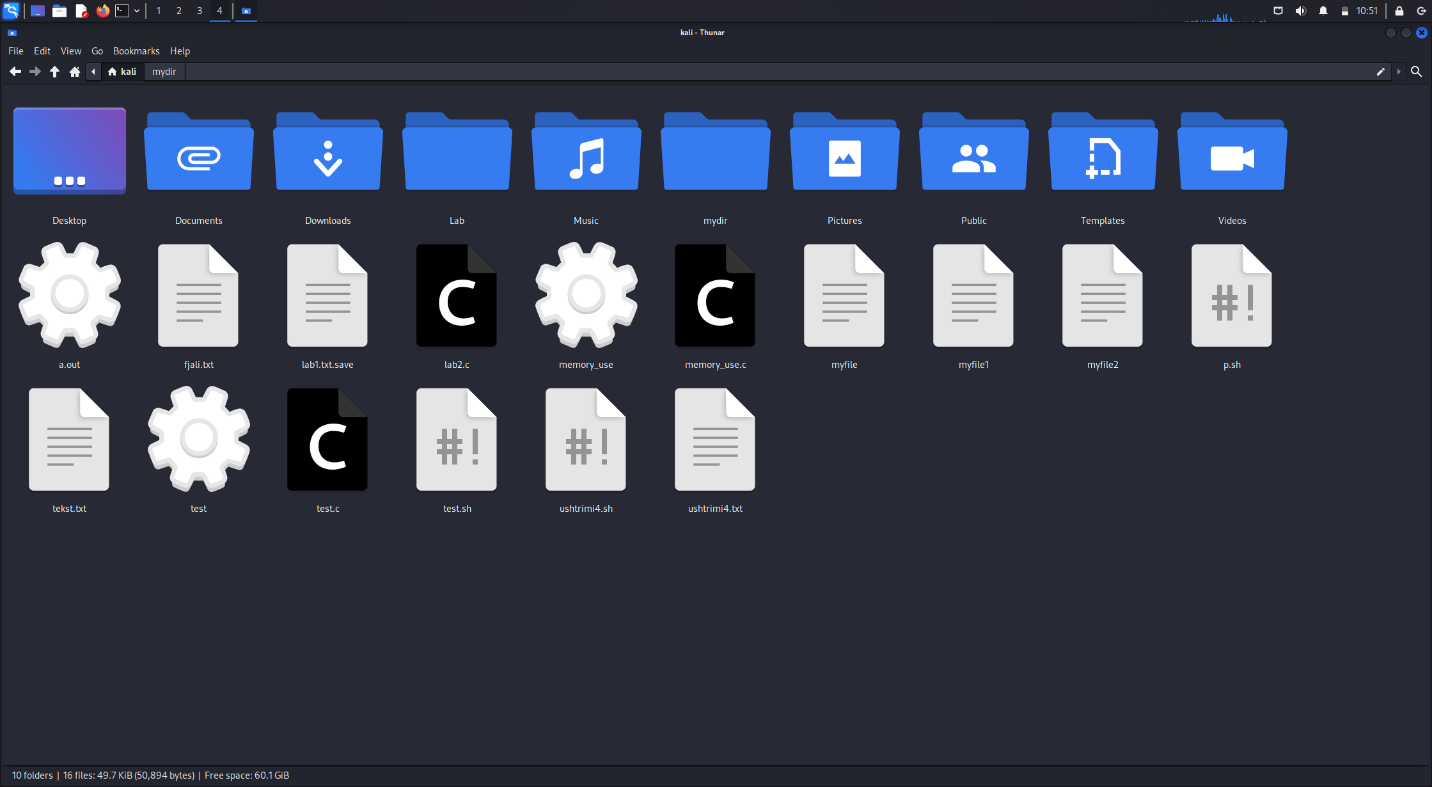
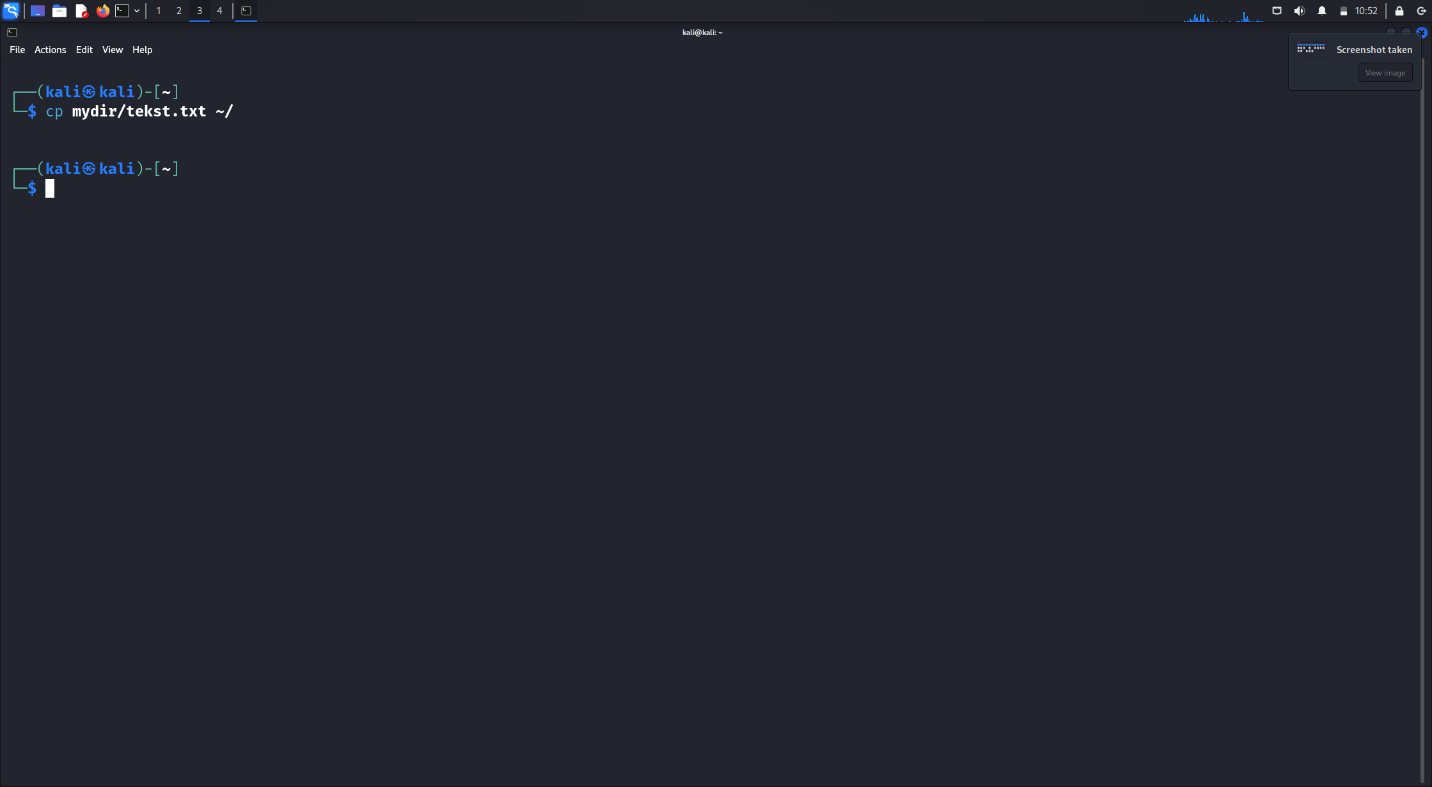
Beni rename te tij

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

Kopjojeni kete file dhe kalojeni ne nje direktori me siper:cp

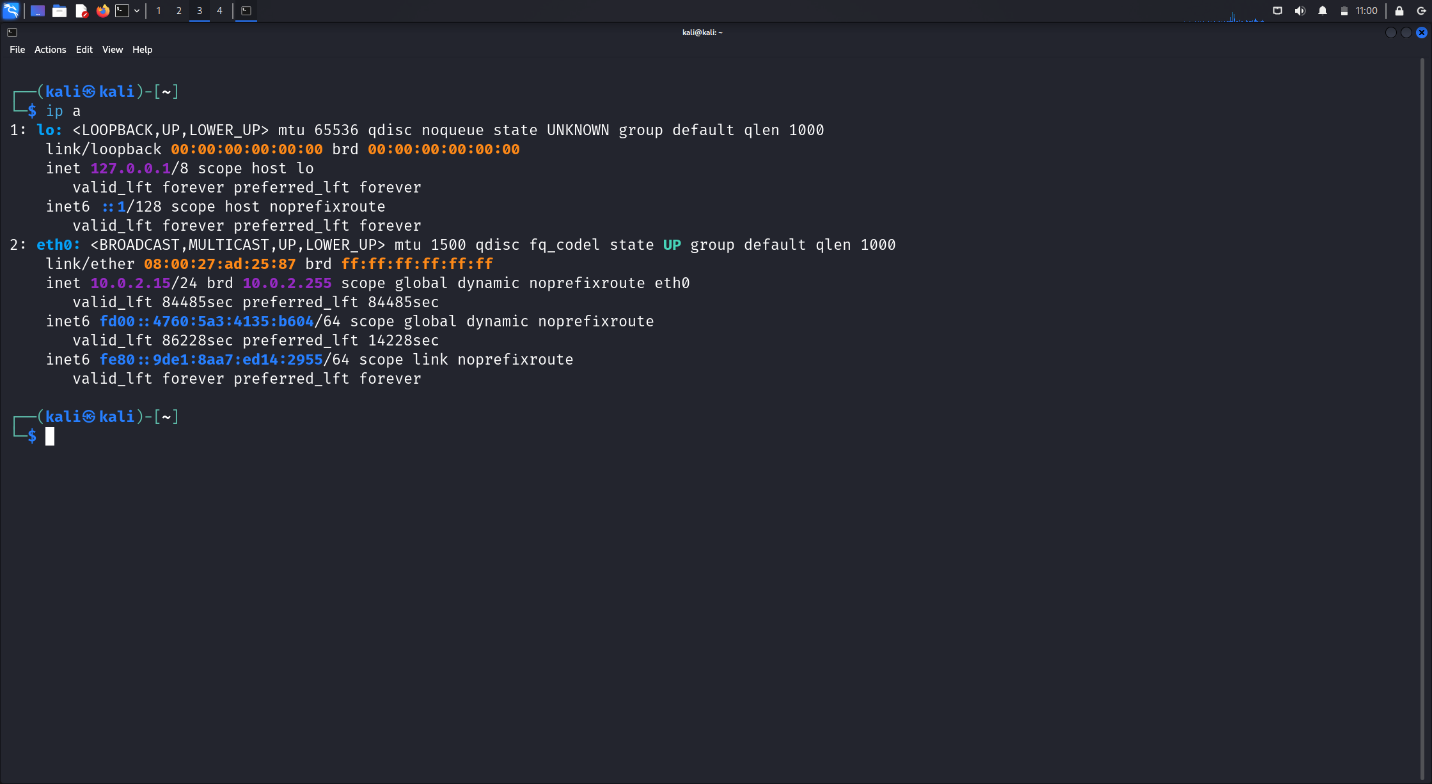


Fshijeni nendirektorine:rmdir

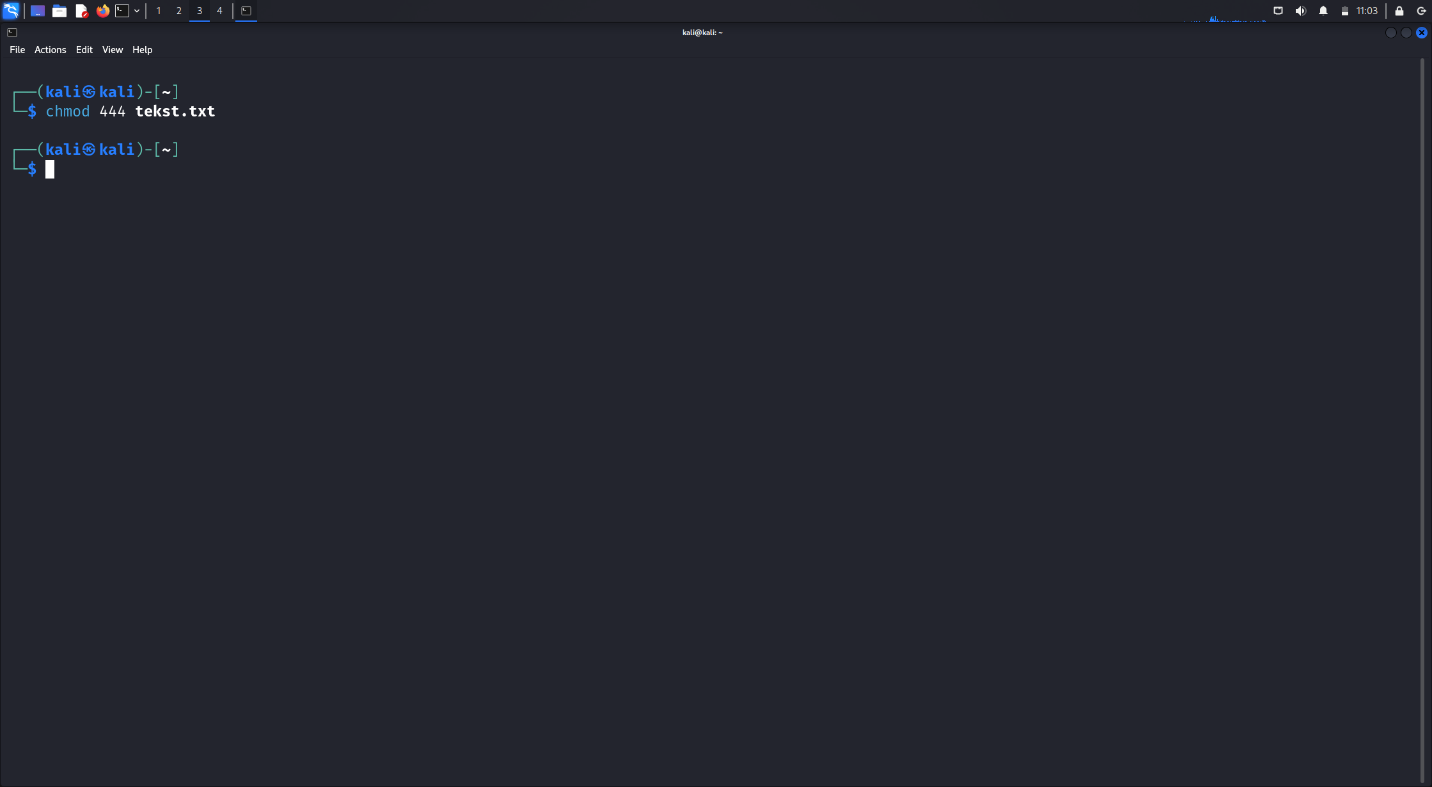
A screenshot of a computer

Description automatically generated

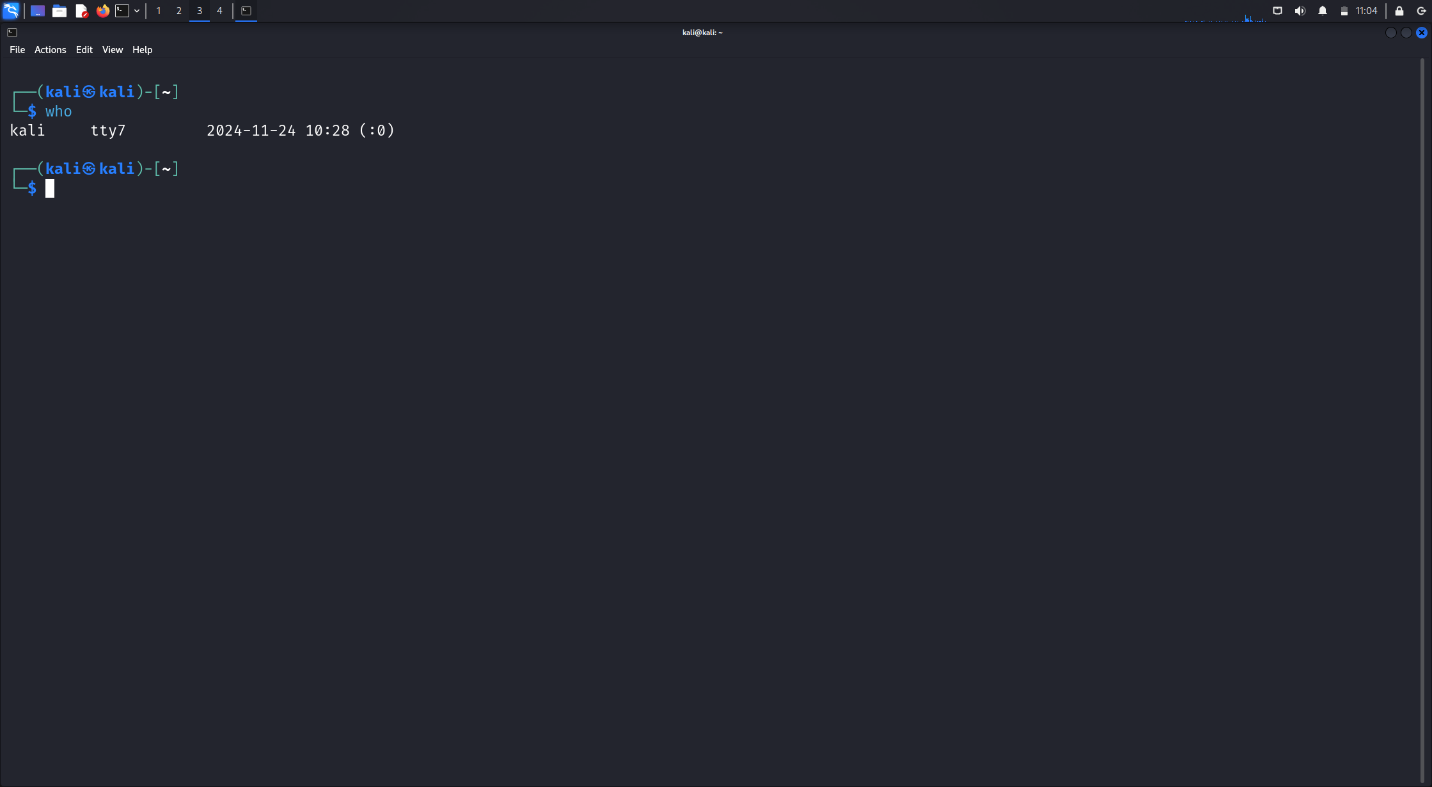
Cekoni konfigurimin e kartave te rrjetit: ip a ose tek /etc/netplan

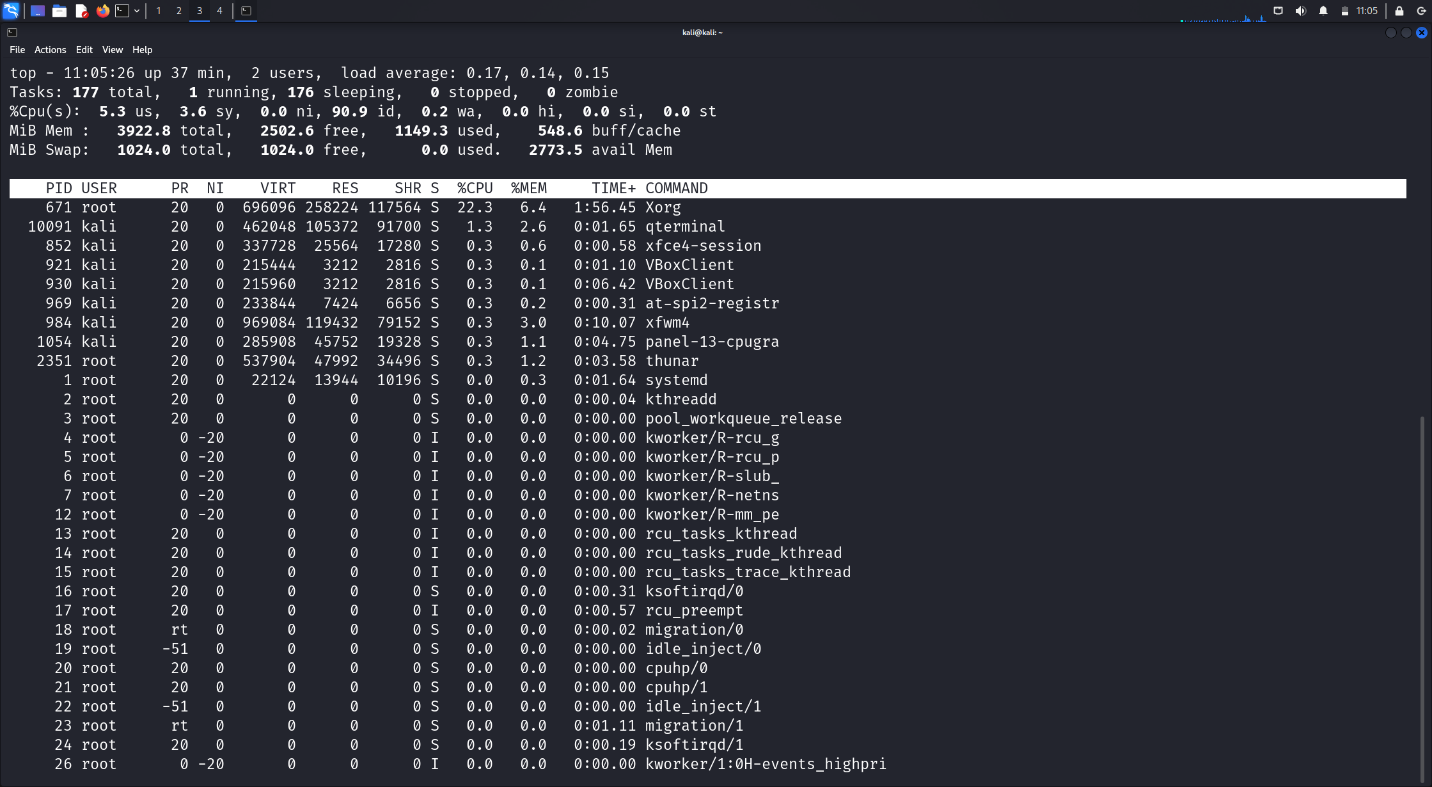


Bejeni file readonly.



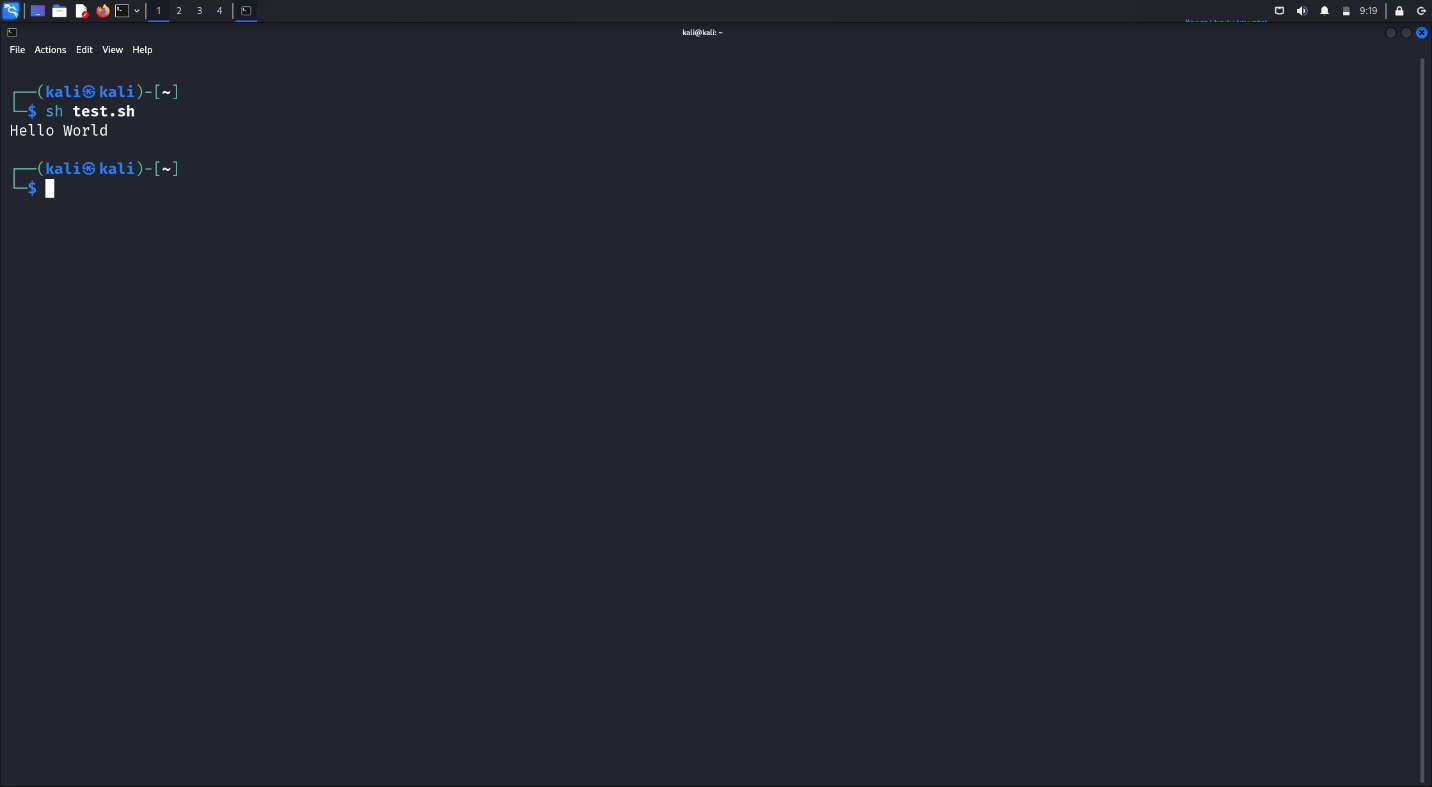
Cilet user jane te loguar aktualisht:who



Cilat jane proceset qe po ekzekutohen dhe sa ram cpu po harxhojne?  


**Pika 2:** Shkruaj një shell script që printon Hello World në ekran.





**Ushtrimi 1:**

Përdor komandën ***who*** dhe ridrejto outputin në file-in *myfile1* dhe më pas përdor komandën ***more*** për të parë përmbajtjen e file-it.

Shkruajmë ne terminal komandën e dhënë më poshtë për të ridrejtuar outputin në *myfile1*:

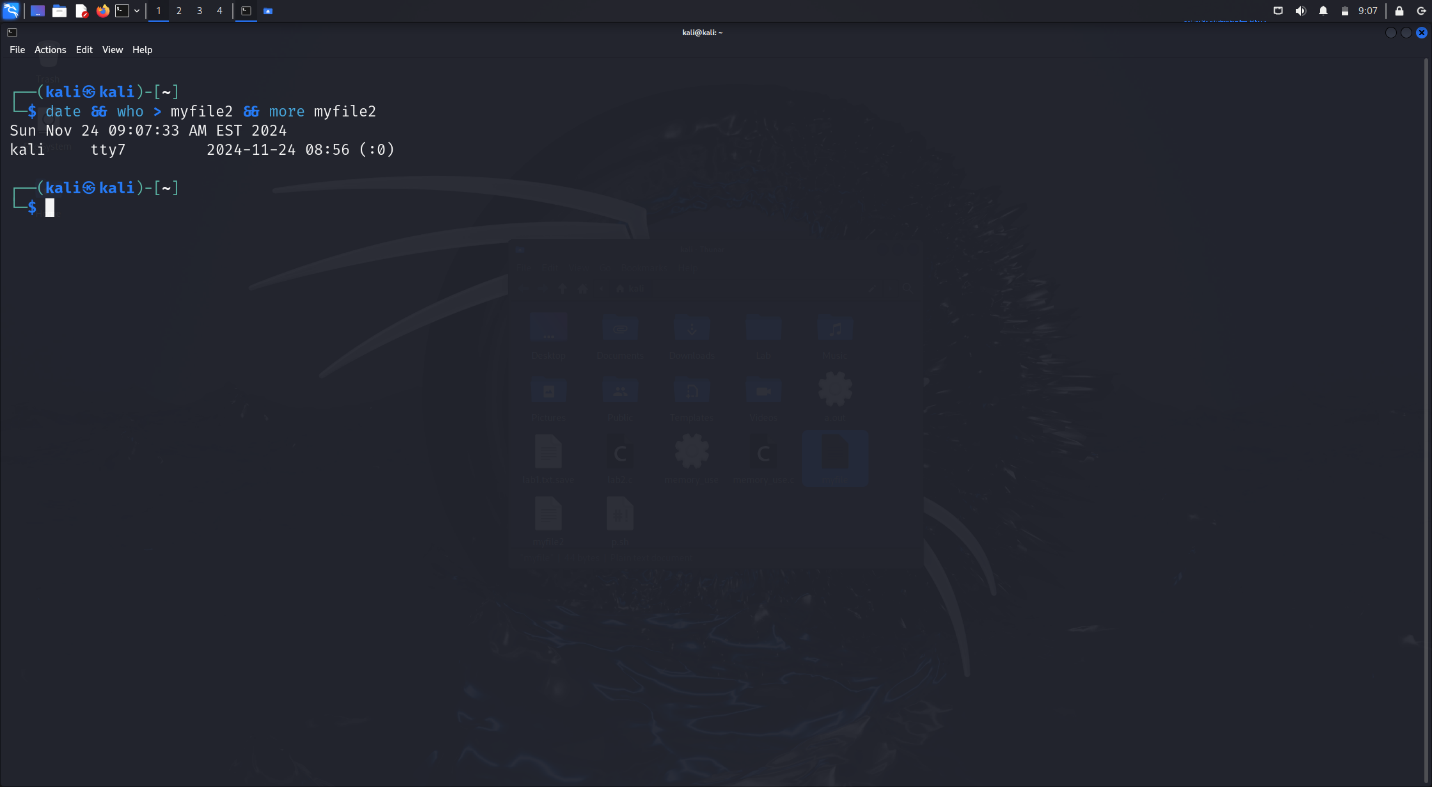
Tani pëdorim komandën ***more*** dhe shohim outputin e file-it që është user-i që po përdorim dhe të dhënat e datës dhe orës që është përdorur komanda ***who***.

A screenshot of a computer

Description automatically generated

**Ushtrimi 2:**

Përdor komandën ***date*** dhe ***who*** në sekuencë në një rresht në mënyrë që output-i i ***date*** të shfaqet në ekran kurse ai i ***who*** të ridrejtohet tek file-i *myfile2*. Përdor komandën ***more*** për të parë përmbajtjen e file-it *myfile2*.



Në këtë rast ne përdorim *AND* (&&) që nënkupton detyrimin e të tre komandave të kryhen.

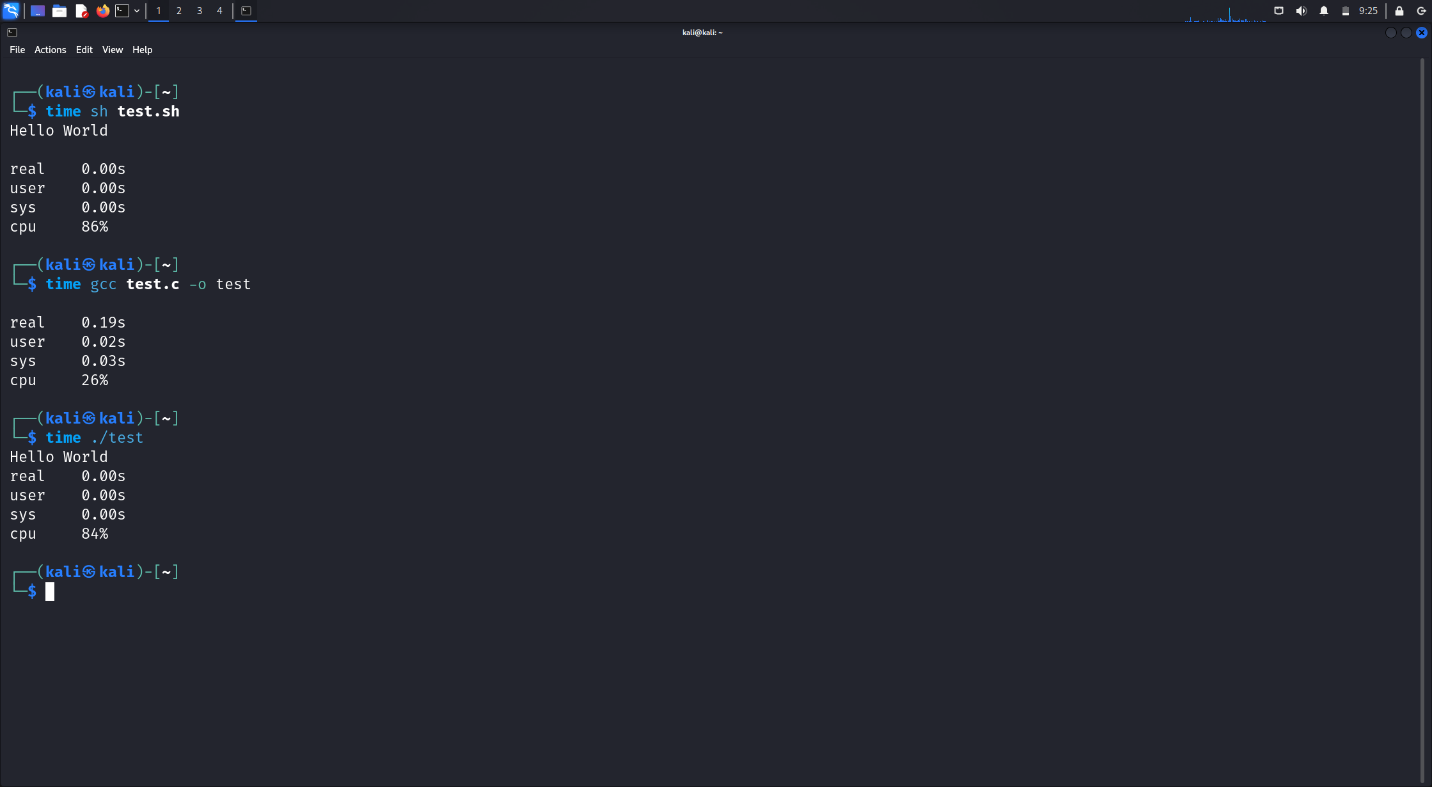
**Ushtrimi 3:**

Shkruaj një program në C që printon “Hello World” dhe krahaso kohët e ekzekutimit me shell scriptin e krijuar me komandën ***time.***

*Programi në C i krijuar:*



Për të ekzekutuar shell scriptin përdorim komandën ***$sh***. Përpara te dy komandave do vendosim komandën ***time*** të masim kohën.

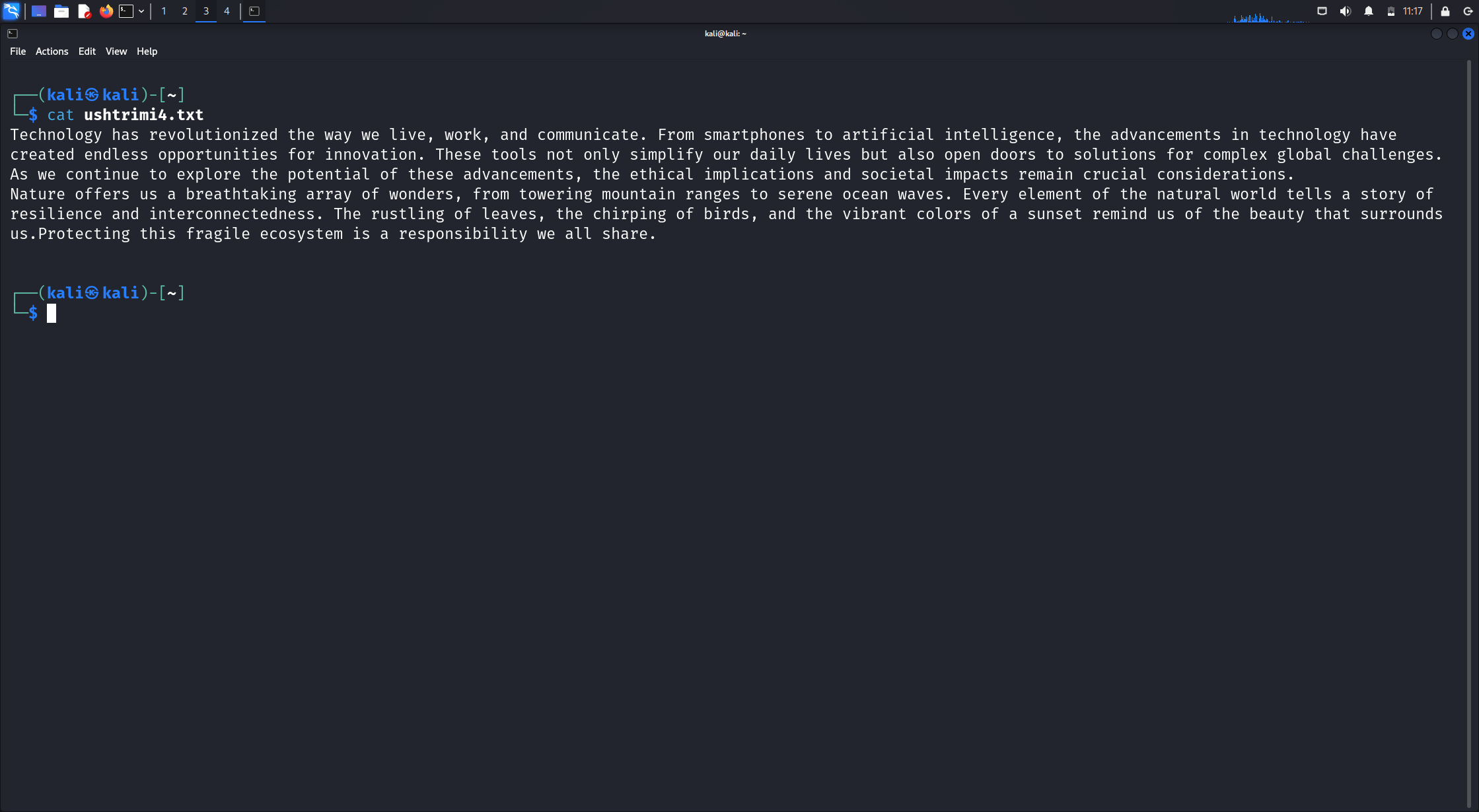


Siç shohim dhe më sipër në lidhje me ekzekutimin e programeve, shell scripti do më shumë kohë për t’u ekzekutuar sesa programi në C. Por shell scripti ka një build-in kompilator dhe në kohë reale ekzekutohet më shpejt dhe me një rrjesht komande, kurse në rastin e programit C për sa kohë nuk kemi krijuar file-in që më pas mund ta ekzekutojmë do një kohë pak më të gjatë. Për të krijuar këtë file, përdorim komandën ***gcc***.

**Ushtrimi 4:**

Shkruaj një shell script që merr si argumenta nje emër file-i dhe dy numra që përfaqësojnë rreshtin e fillimit dhe të mbarimit dhe shfaq të gjithë rreshtat e file-it ndërmjet rreshtit të fillimit dhe mbarimit.

*File-i që do shqyrtojmë:*

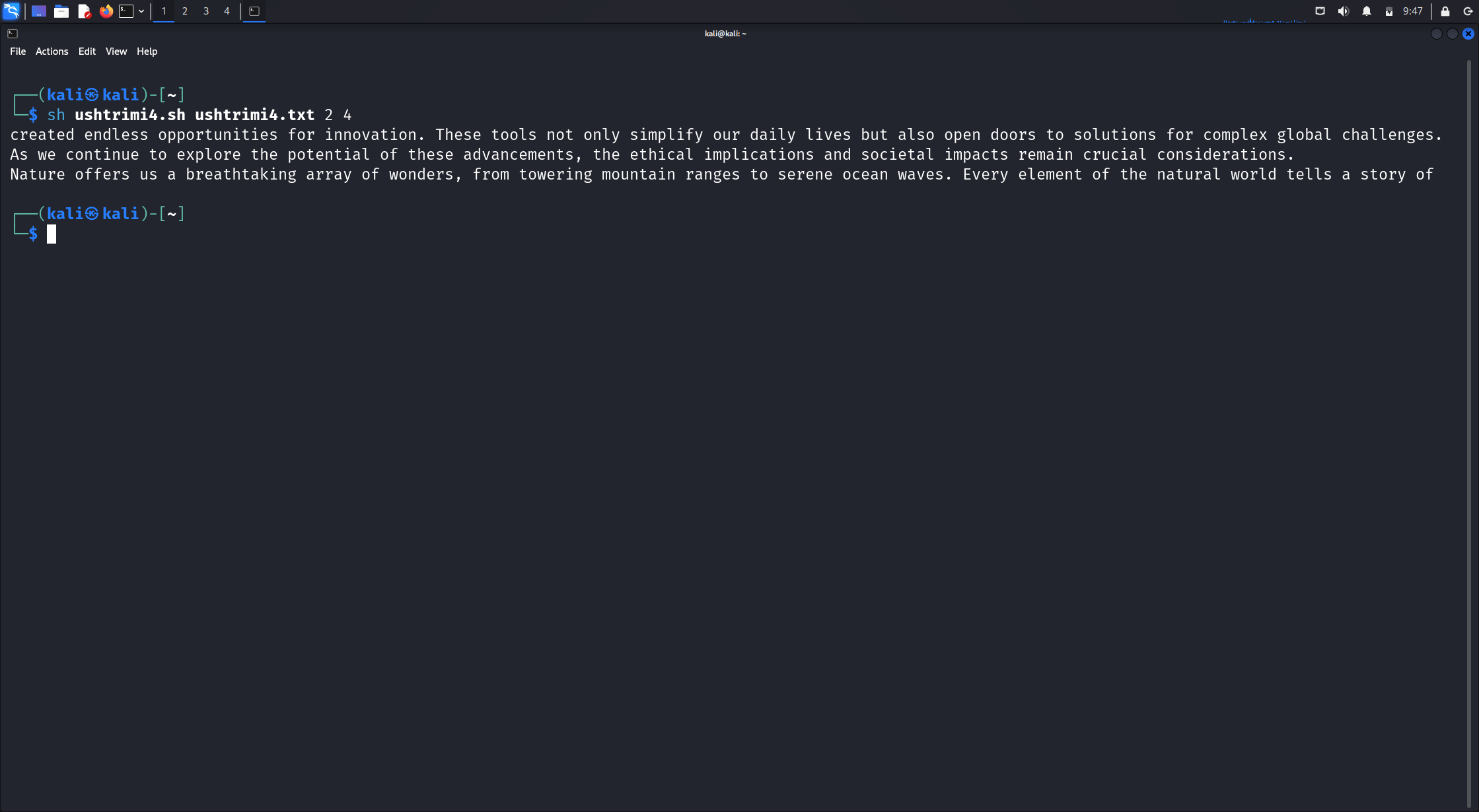


Shkruajmë në nano kodin e dëshiruar të shell script-it:



Siç vëmë re në kodin e mësipërm si fillim kontrollojmë argumentat që dëshirojmë me një kusht *if*. Më pas i’u japim vlerë tre variablave me argumentat e marra, dhe çekojmë nëse file-i i marrë ekziston në librari. Gjithashtu vendosim dhe një kusht për rrjeshtat. Me komandën sed -n “$p” kryejmë funksionin që dëshirojmë.

Tani le të ekzekutojmë kodin me file-in *ushtrimi4lab1.txt* dhe si rrjesht fillimi *2* e mbarimi *5*.



Në rastin që nuk vendosëm argumentat na i kërkoi programi siç tregohet dhe në figurë. Në rastin e dytë kodi u ekzekutua pa asnjë error.