CSc 3320: Systems Programming

Fall 2021

Midterm 1: Total points = 100

Submission instructions:

1. Create a Google doc for your submission.
2. Start your responses from page 2 of the document and copy these instructions on page 1.
3. Fill in your name, campus ID and panther # in the fields provided. If this information is missing TWO POINTS WILL BE DEDUCTED.
4. Keep this page 1 intact. If this *submissions instructions* page is missing in your submission TWO POINTS WILL BE DEDUCTED.
5. Start your responses to each QUESTION on a new page.
6. If you are being asked to write code copy the code into a separate txt file and submit that as well. The code should be executable. E.g. if asked for a C program then provide myfile.c so that we can execute that script. In your answer to the specific question, provide the steps on how to execute your file (like a ReadMe).
7. If you are being asked to test code or run specific commands or scripts, provide the evidence of your outputs through a screenshot and/or screen video-recordings and copy the same into the document.
8. Upon completion, download a .PDF version of the google doc document and submit the same along with all the supplementary files (videos, pictures, scripts etc).
9. Scripts/Code without proper comments, indentation and titles (must have the name of the program, and name & email of the programmer on top the script).

Full Name: Faisal Musa

Campus ID: fmusa1

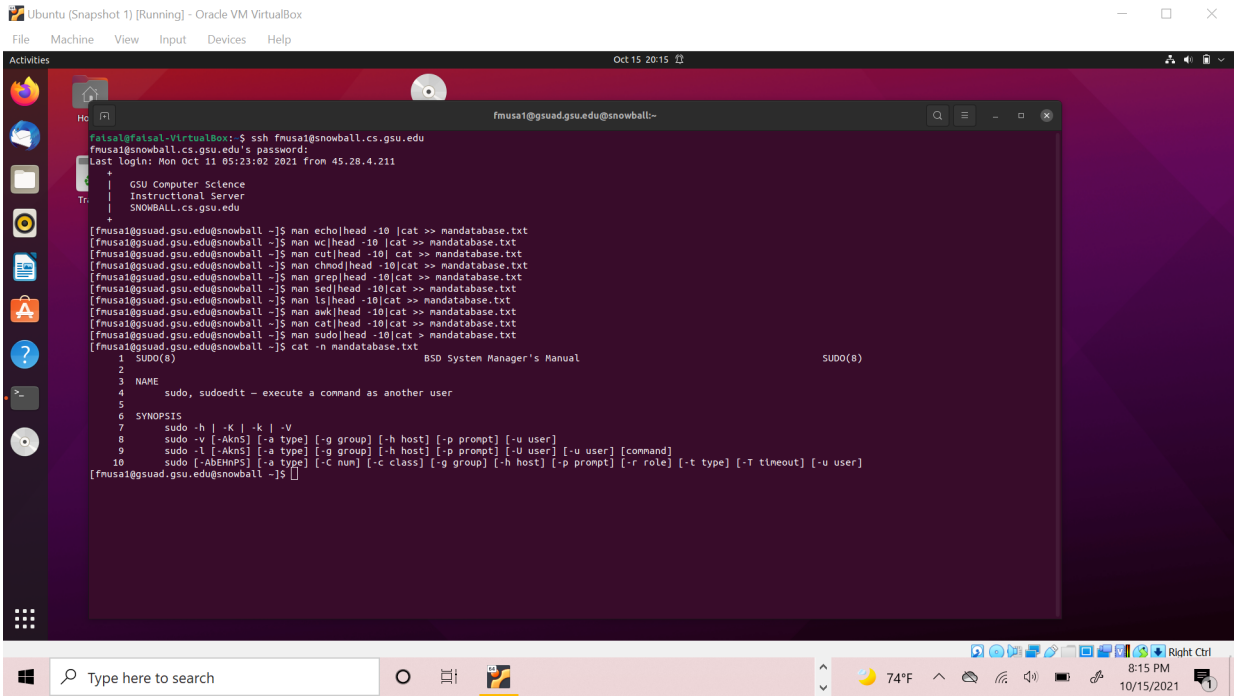
Panther #: 002396860

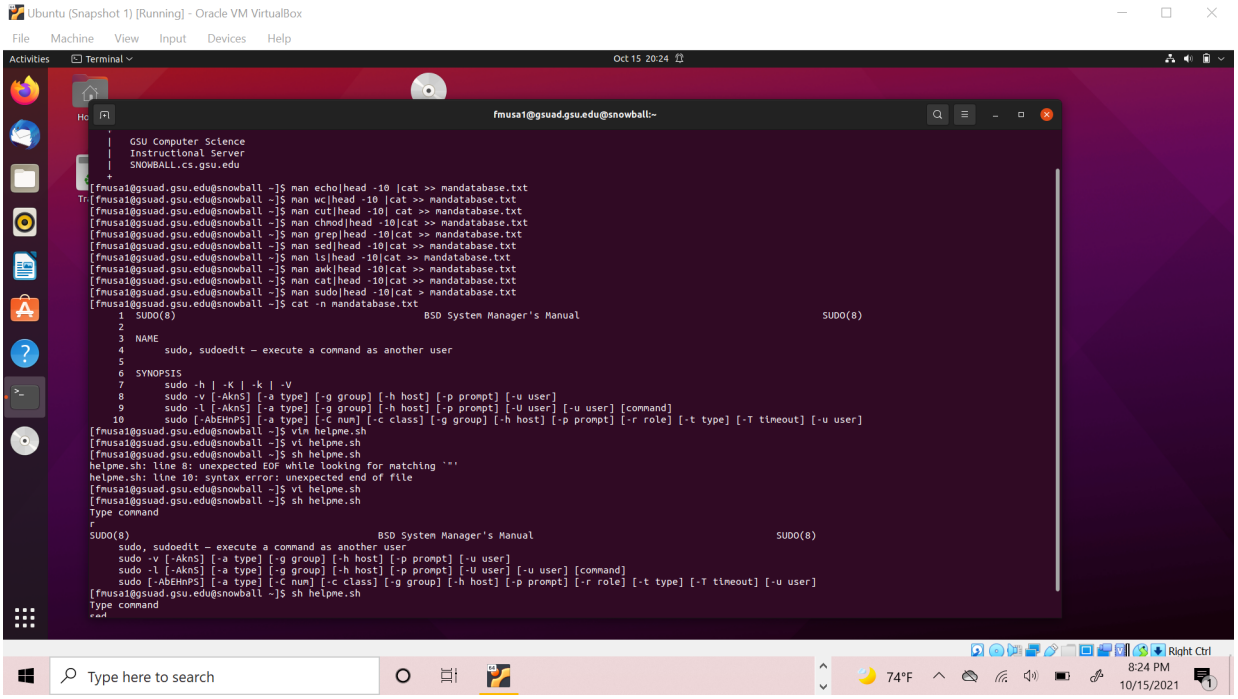
Questions 1-5 are 20pts each

1. (20 pts) Pick any of your 10 favourite unix commands. For each command run the *man* command and copy the text that is printed into a mandatabase.txt. Write a shell script *helpme.sh* that will ask the user to type in a command and then print the manual's text associated with that corresponding command. If the command the user types is not in the database then the script must print

   sorry, I cannot help you

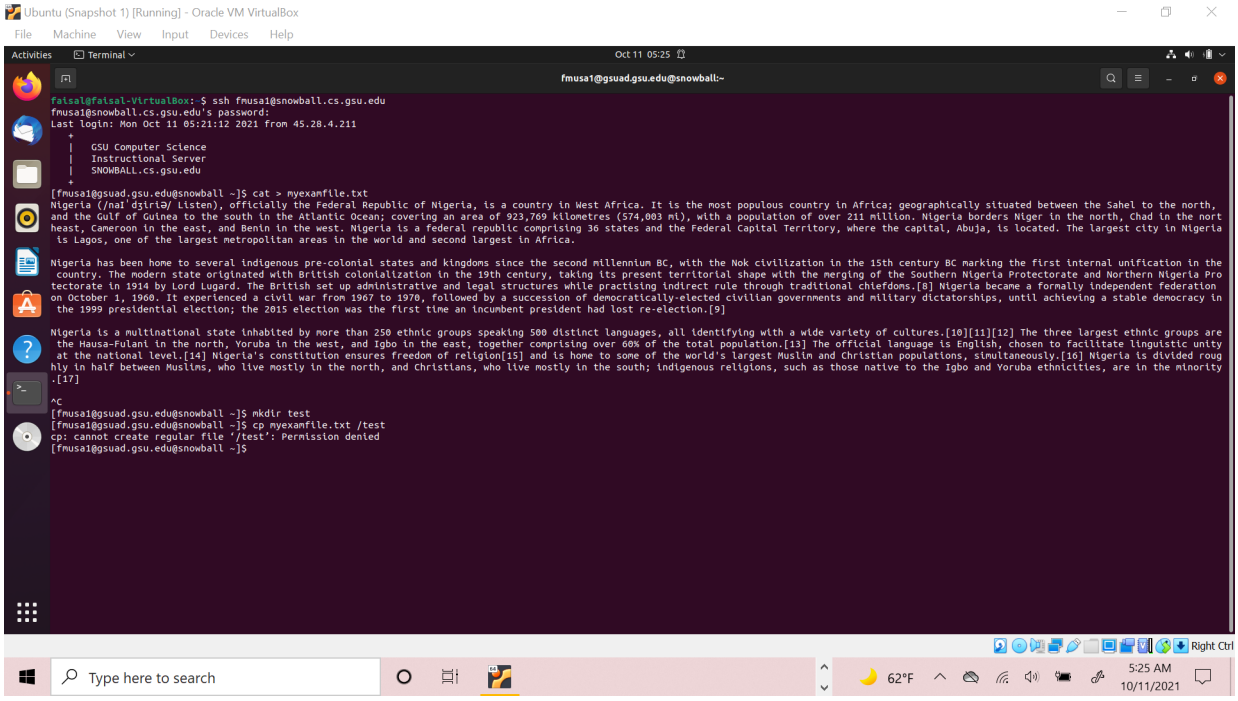
   man ls|head -10| cat > mandatabase.txt
   man cd|head -10| cat > mandatabase.txt
   man echo|head -10| cat >> mandatabase.txt
   man mkdir|head -10| cat >> mandatabase.txt
   man chmod|head -10| cat > mandatabase.txt
   man sudo|head -10| cat > mandatabase.txt
   man cat|head -10| cat >> mandatabase.txt
   man grep|head -10| cat > mandatabase.txt
   man wc|head -10| cat >> mandatabase.txt
   man sed|head -10| cat >> mandatabase.txt

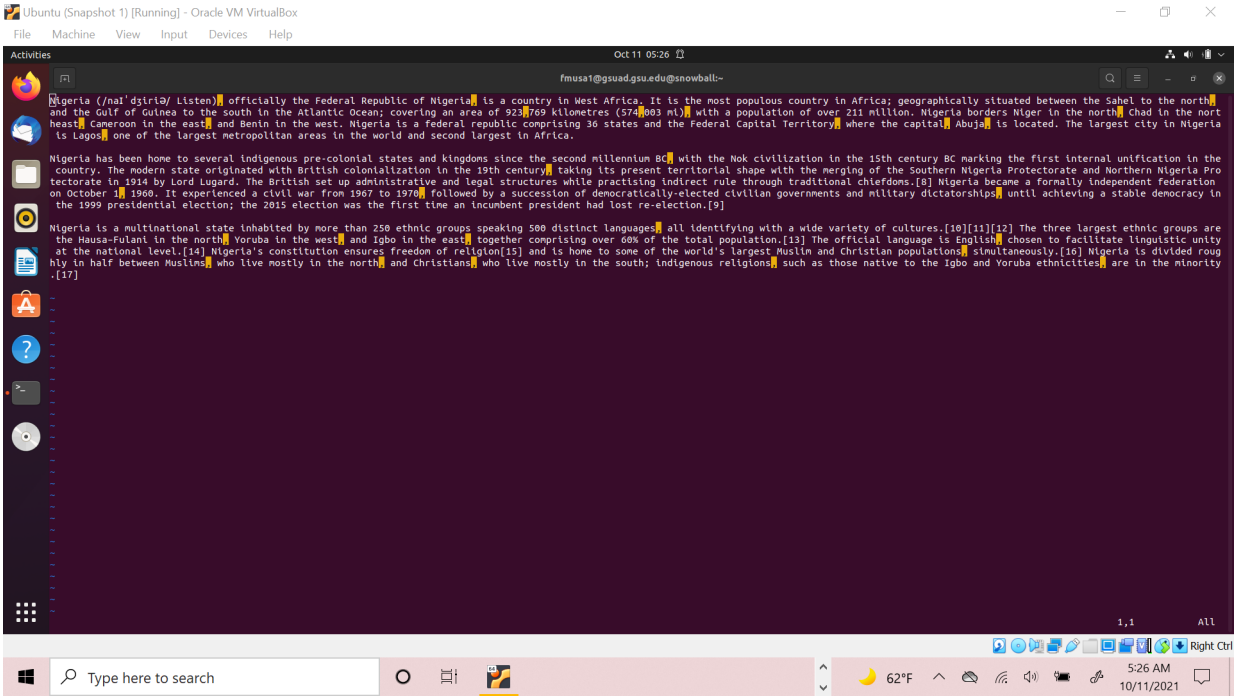

Ubuntu (Snapshot 1) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Oct 15 20:15
fmusa1@gsuad.gsu.edu@snowball:~
faisal@faisal-VirtualBox: $ ssh fmusa1@snowball.cs.gsu.edu
fmusa1@snowball.cs.gsu.edu's password:
Last login: Mon Oct 11 05:23:02 2021 from 45.28.4.211
GSU Computer Science
Instructional Server
SNOWBALL.cs.gsu.edu
[fmusa1@gsuad.gsu.edu@snowball ~]$ man echo|head -10 |cat >> mandatabase.txt
[fmusa1@gsuad.gsu.edu@snowball ~]$ man wc|head -10 |cat >> mandatabase.txt
[fmusa1@gsuad.gsu.edu@snowball ~]$ man cut|head -10| cat >> mandatabase.txt
[fmusa1@gsuad.gsu.edu@snowball ~]$ man chmod|head -10|cat >> mandatabase.txt
[fmusa1@gsuad.gsu.edu@snowball ~]$ man grep|head -10|cat >> mandatabase.txt
[fmusa1@gsuad.gsu.edu@snowball ~]$ man sed|head -10|cat >> mandatabase.txt
[fmusa1@gsuad.gsu.edu@snowball ~]$ man ls|head -10|cat >> mandatabase.txt
[fmusa1@gsuad.gsu.edu@snowball ~]$ man awk|head -10|cat >> mandatabase.txt
[fmusa1@gsuad.gsu.edu@snowball ~]$ man cat|head -10|cat >> mandatabase.txt
[fmusa1@gsuad.gsu.edu@snowball ~]$ man sudo|head -10|cat > mandatabase.txt
[fmusa1@gsuad.gsu.edu@snowball ~]$ cat -n mandatabase.txt
1 SUDO(8) BSD System Manager's Manual SUDO(8)
2
3 NAME
4 sudo, sudoedit – execute a command as another user
5
6 SYNOPSIS
7 sudo -h | -K | -k | -V
8 sudo -v [-AknS] [-a type] [-g group] [-h host] [-p prompt] [-u user]
9 sudo -l [-AknS] [-a type] [-g group] [-h host] [-p prompt] [-U user] [-u user] [command]
10 sudo [-AbEHnPS] [-a type] [-C num] [-c class] [-g group] [-h host] [-p prompt] [-r role] [-t type] [-T timeout] [-u user]
[fmusa1@gsuad.gsu.edu@snowball ~]$
Right Ctrl
Type here to search
74°F
8:15 PM
10/15/2021


Ubuntu (Snapshot 1) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Terminal
Oct 15 20:24
fmusa1@gsuad.gsu.edu@snowball:~
GSU Computer Science
Instructional Server
SNOWBALL.cs.gsu.edu
[fmusa1@gsuad.gsu.edu@snowball ~]$ man echo|head -10 |cat >> mandatabase.txt
[fmusa1@gsuad.gsu.edu@snowball ~]$ man wc|head -10 |cat >> mandatabase.txt
[fmusa1@gsuad.gsu.edu@snowball ~]$ man cut|head -10| cat >> mandatabase.txt
[fmusa1@gsuad.gsu.edu@snowball ~]$ man chmod|head -10|cat >> mandatabase.txt
[fmusa1@gsuad.gsu.edu@snowball ~]$ man grep|head -10|cat >> mandatabase.txt
[fmusa1@gsuad.gsu.edu@snowball ~]$ man sed|head -10|cat >> mandatabase.txt
[fmusa1@gsuad.gsu.edu@snowball ~]$ man ls|head -10|cat >> mandatabase.txt
[fmusa1@gsuad.gsu.edu@snowball ~]$ man awk|head -10|cat >> mandatabase.txt
[fmusa1@gsuad.gsu.edu@snowball ~]$ man cat|head -10|cat >> mandatabase.txt
[fmusa1@gsuad.gsu.edu@snowball ~]$ man sudo|head -10|cat > mandatabase.txt
[fmusa1@gsuad.gsu.edu@snowball ~]$ cat -n mandatabase.txt
1 SUDO(8) BSD System Manager's Manual SUDO(8)
2
3 NAME
4 sudo, sudoedit – execute a command as another user
5
6 SYNOPSIS
7 sudo -h | -K | -k | -V
8 sudo -v [-AknS] [-a type] [-g group] [-h host] [-p prompt] [-u user]
9 sudo -l [-AknS] [-a type] [-g group] [-h host] [-p prompt] [-U user] [-u user] [command]
10 sudo [-AbEHnPS] [-a type] [-C num] [-c class] [-g group] [-h host] [-p prompt] [-r role] [-t type] [-T timeout] [-u user]
[fmusa1@gsuad.gsu.edu@snowball ~]$ vim helpme.sh
[fmusa1@gsuad.gsu.edu@snowball ~]$ vi helpme.sh
[fmusa1@gsuad.gsu.edu@snowball ~]$ sh helpme.sh
helpme.sh: line 8: unexpected EOF while looking for matching `"'
helpme.sh: line 10: syntax error: unexpected end of file
[fmusa1@gsuad.gsu.edu@snowball ~]$ vi helpme.sh
[fmusa1@gsuad.gsu.edu@snowball ~]$ sh helpme.sh
Type command
r
SUDO(8) BSD System Manager's Manual SUDO(8)
sudo, sudoedit – execute a command as another user
sudo -v [-AknS] [-a type] [-g group] [-h host] [-p prompt] [-u user]
sudo -l [-AknS] [-a type] [-g group] [-h host] [-p prompt] [-U user] [-u user] [command]
sudo [-AbEHnPS] [-a type] [-C num] [-c class] [-g group] [-h host] [-p prompt] [-r role] [-t type] [-T timeout] [-u user]
[fmusa1@gsuad.gsu.edu@snowball ~]$ sh helpme.sh
Type command
Right Ctrl
Type here to search
74°F
8:24 PM
10/15/2021

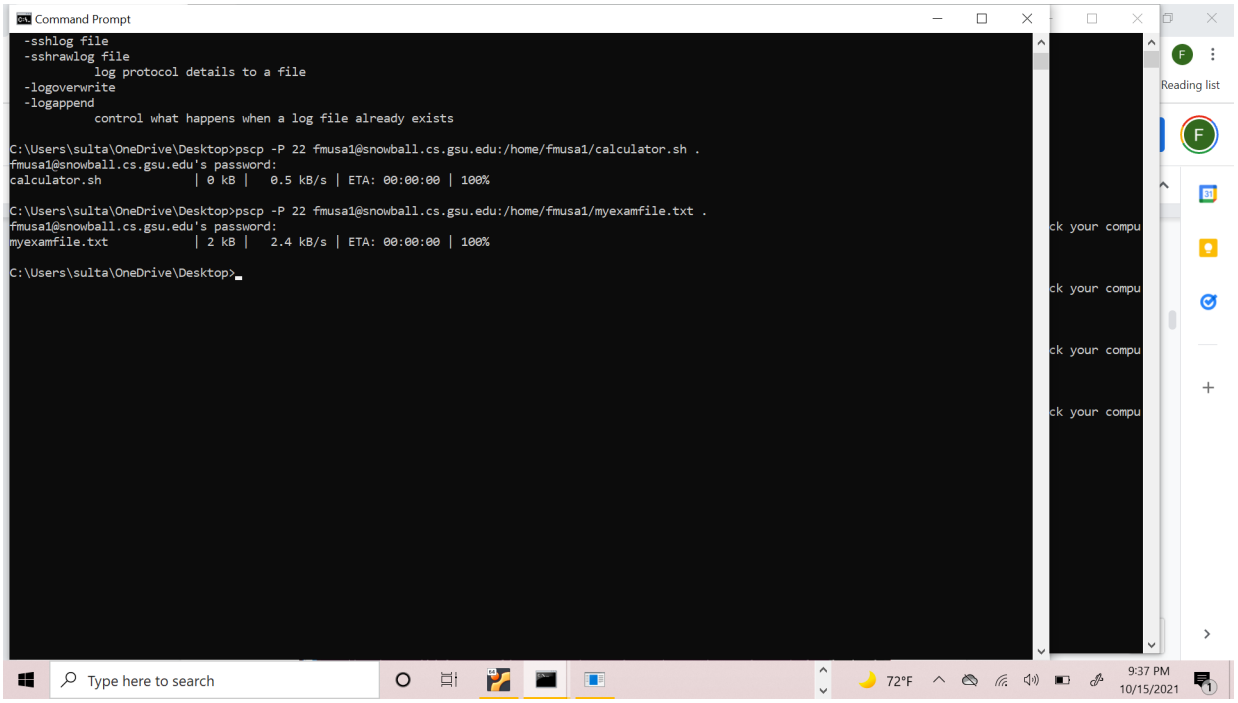
2. (10pts each) On your computer open your favourite Wikipedia page. Copy the text from that page into a text file **myexamfile.txt** and then copy that file to a directory named **midterm** (use mkdir to create the directory if it doesn't exist) in your snowball server home directory (use any FTP tool such as Putty or Filezilla to copy the file from your computer to the remote snowball server machine: see Lab 6).

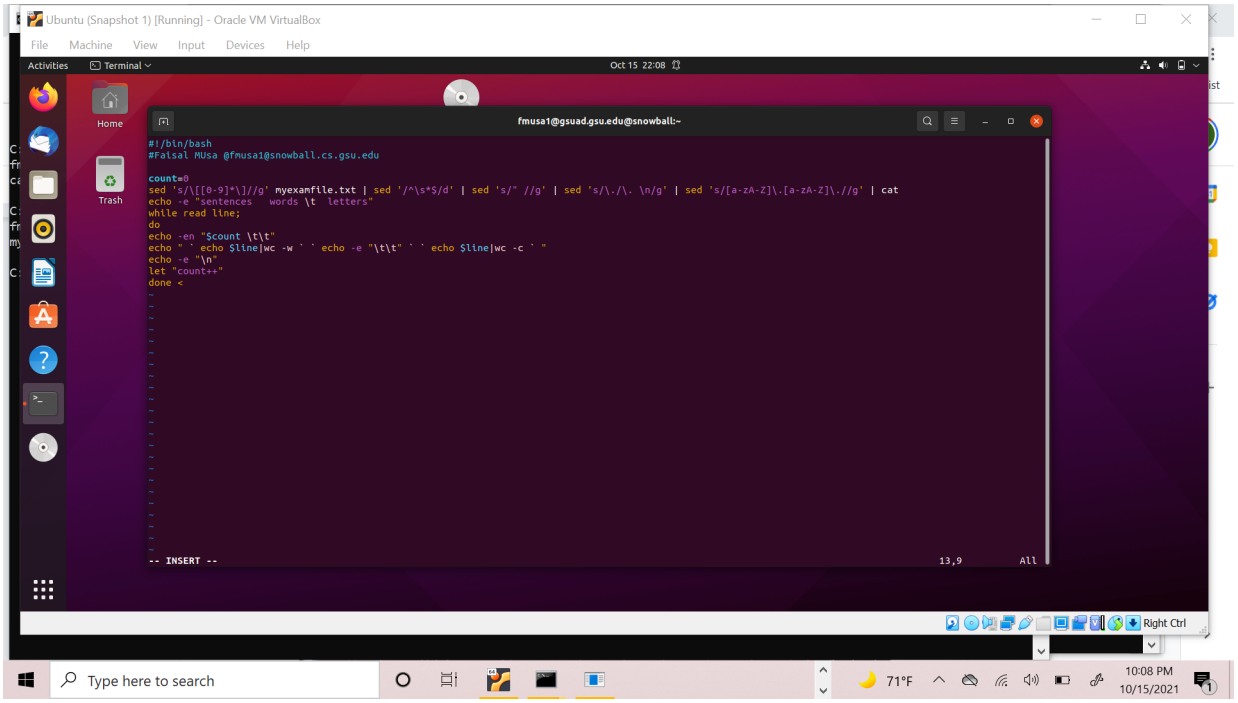
a. Write a shell script that will find the number of statements in the text. A statement is defined as the collection of text between two periods (full-stops).

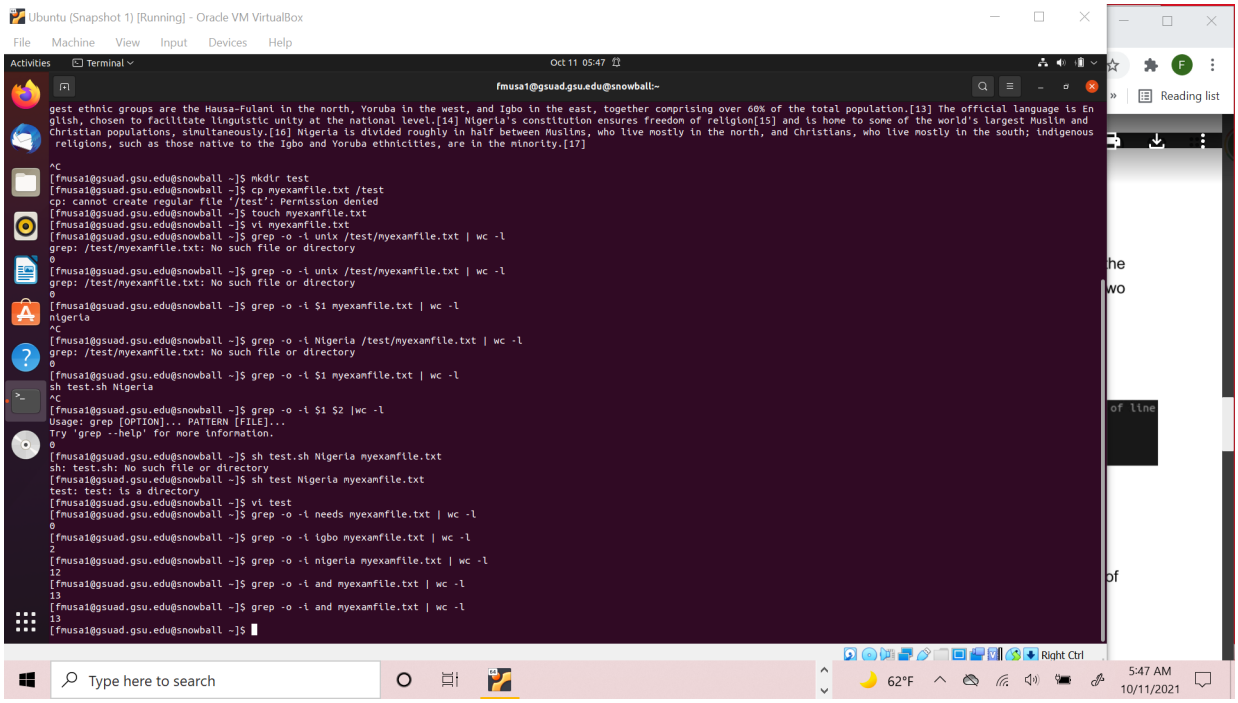




b. Update the script to present a tabular list that shows the number of words and number of letters in each statement.

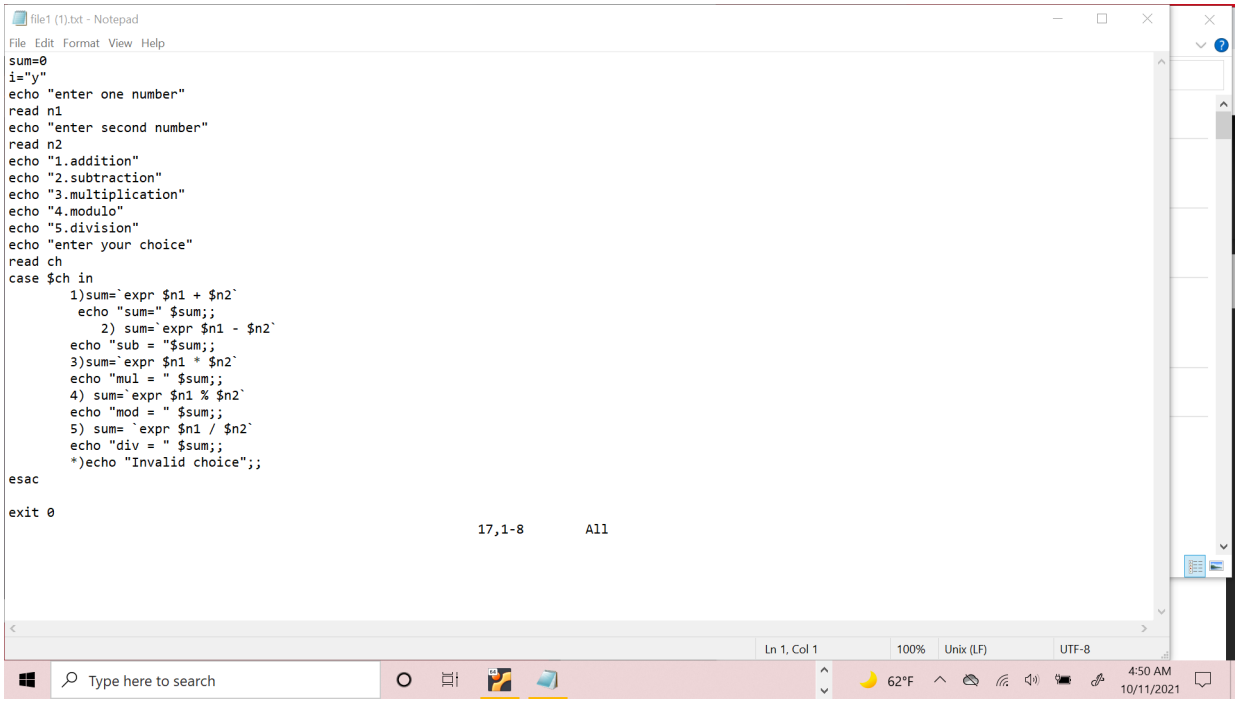


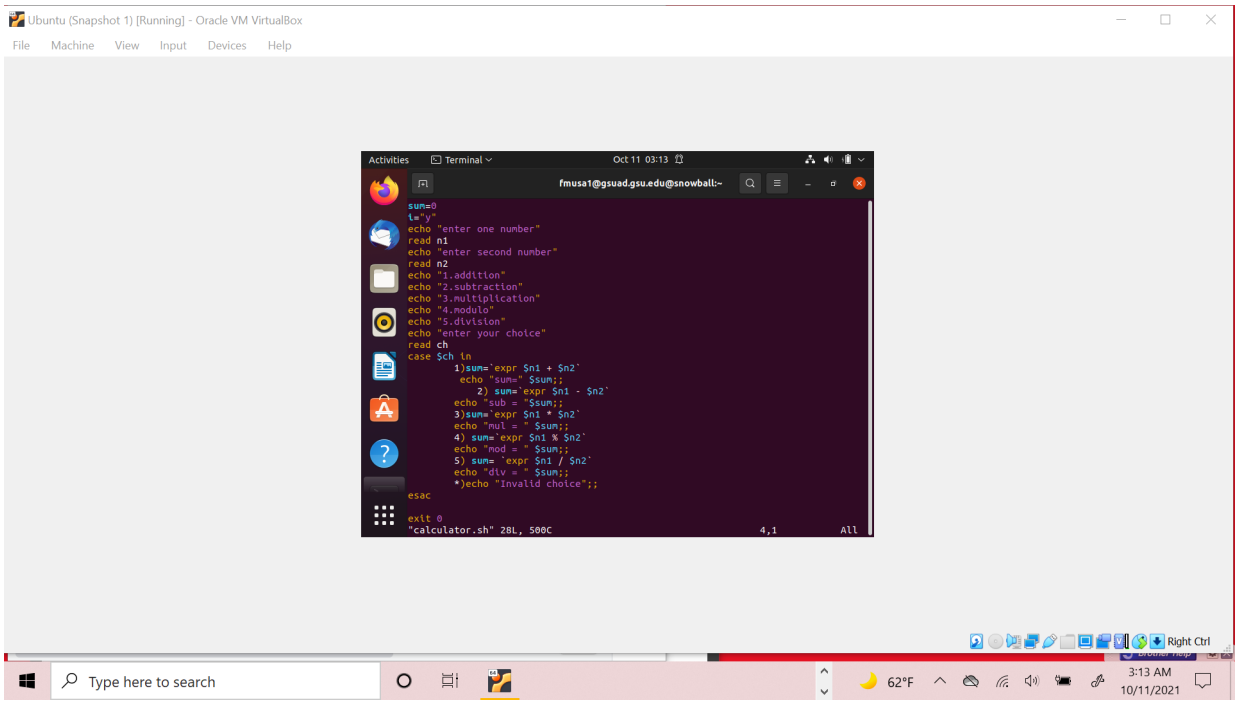

Ubuntu (Snapshot 1) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Oct 15 22:08
Home
Trash
fmusa1@gsuad.gsu.edu@snowball:~
#!/bin/bash
#Faisal MUsa @fmusa1@snowball.cs.gsu.edu
count=0
sed 's/\[[0-9]*\]//g' myexamfile.txt | sed '/^\s*$/d' | sed 's/" //g' | sed 's/\./\. \n/g' | sed 's/[a-zA-Z]\.[a-zA-Z]\.//g' | cat
echo -e "sentences words \t letters"
while read line;
do
echo -en "$count \t\t"
echo " ` echo $line|wc -w ` ` echo -e "\t\t" ` ` echo $line|wc -c ` "
echo -e "\n"
let "count++"
done <
-- INSERT -- 13,9 All
Right Ctrl
Type here to search
71°F
10:08 PM
10/15/2021


Ubuntu (Snapshot 1) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Oct 11 05:47
fmusa1@gsuad.gsu.edu@snowball:~
gest ethnic groups are the Hausa-Fulani in the north, Yoruba in the west, and Igbo in the east, together comprising over 60% of the total population.[13] The official language is En
glish, chosen to facilitate linguistic unity at the national level.[14] Nigeria's constitution ensures freedom of religion[15] and is home to some of the world's largest Muslim and
Christian populations, simultaneously.[16] Nigeria is divided roughly in half between Muslims, who live mostly in the north, and Christians, who live mostly in the south; indigenous
religions, such as those native to the Igbo and Yoruba ethnicities, are in the minority.[17]
^C
[fmusa1@gsuad.gsu.edu@snowball ~]$ mkdir test
[fmusa1@gsuad.gsu.edu@snowball ~]$ cp myexamfile.txt /test
cp: cannot create regular file '/test': Permission denied
[fmusa1@gsuad.gsu.edu@snowball ~]$ touch myexamfile.txt
[fmusa1@gsuad.gsu.edu@snowball ~]$ vi myexamfile.txt
[fmusa1@gsuad.gsu.edu@snowball ~]$ grep -o -i unix /test/myexamfile.txt | wc -l
grep: /test/myexamfile.txt: No such file or directory
0
[fmusa1@gsuad.gsu.edu@snowball ~]$ grep -o -i unix /test/myexamfile.txt | wc -l
grep: /test/myexamfile.txt: No such file or directory
0
[fmusa1@gsuad.gsu.edu@snowball ~]$ grep -o -i $1 myexamfile.txt | wc -l
nigeria
^C
[fmusa1@gsuad.gsu.edu@snowball ~]$ grep -o -i Nigeria /test/myexamfile.txt | wc -l
grep: /test/myexamfile.txt: No such file or directory
0
[fmusa1@gsuad.gsu.edu@snowball ~]$ grep -o -i $1 myexamfile.txt | wc -l
sh test.sh Nigeria
^C
[fmusa1@gsuad.gsu.edu@snowball ~]$ grep -o -i $1 $2 |wc -l
Usage: grep [OPTION]... PATTERN [FILE]...
Try 'grep --help' for more information.
0
[fmusa1@gsuad.gsu.edu@snowball ~]$ sh test.sh Nigeria myexamfile.txt
sh: test.sh: No such file or directory
[fmusa1@gsuad.gsu.edu@snowball ~]$ sh test Nigeria myexamfile.txt
test: test: is a directory
[fmusa1@gsuad.gsu.edu@snowball ~]$ vi test
[fmusa1@gsuad.gsu.edu@snowball ~]$ grep -o -i needs myexamfile.txt | wc -l
0
[fmusa1@gsuad.gsu.edu@snowball ~]$ grep -o -i igbo myexamfile.txt | wc -l
2
[fmusa1@gsuad.gsu.edu@snowball ~]$ grep -o -i nigeria myexamfile.txt | wc -l
12
[fmusa1@gsuad.gsu.edu@snowball ~]$ grep -o -i and myexamfile.txt | wc -l
13
[fmusa1@gsuad.gsu.edu@snowball ~]$ grep -o -i and myexamfile.txt | wc -l
13
[fmusa1@gsuad.gsu.edu@snowball ~]$
Reading list
Right Ctrl
Type here to search
62°F
5:47 AM
10/11/2021

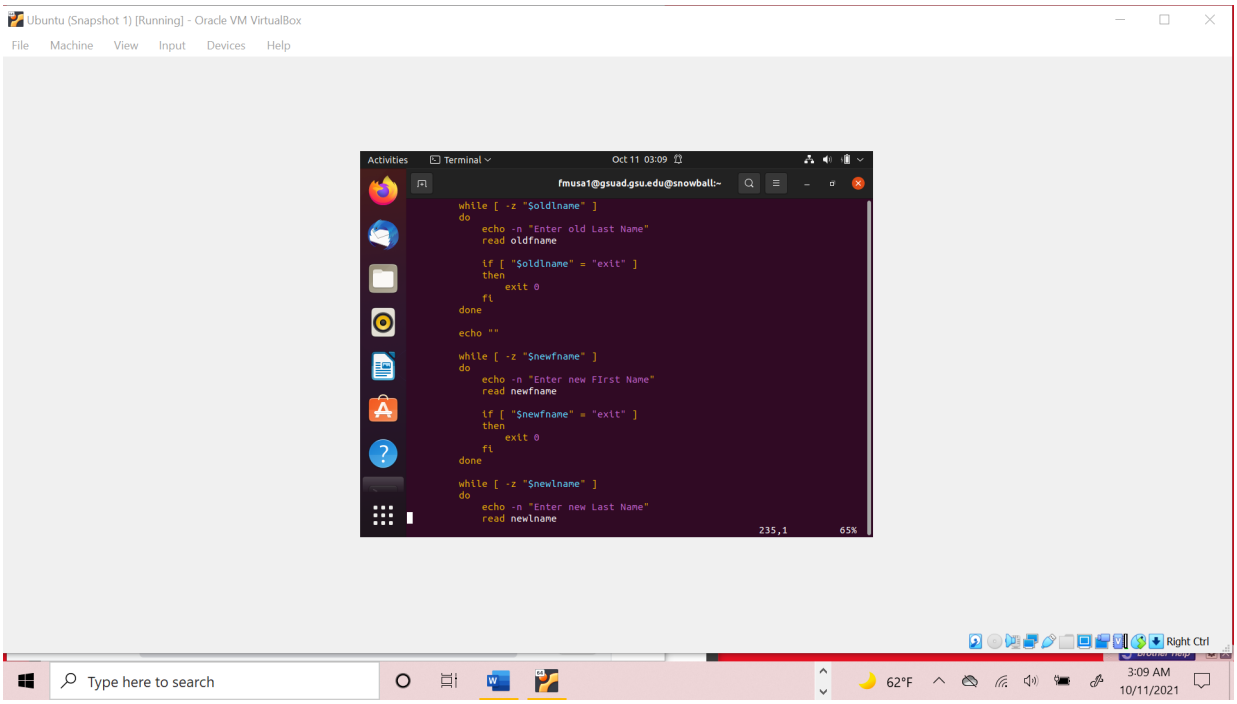
3. (20pts) Design a calculator using a shell script using regular expressions. The calculator, at the minimum, must be able to process addition, subtraction, multiplication, division and modulo operations. It must also have cancel and clear features.

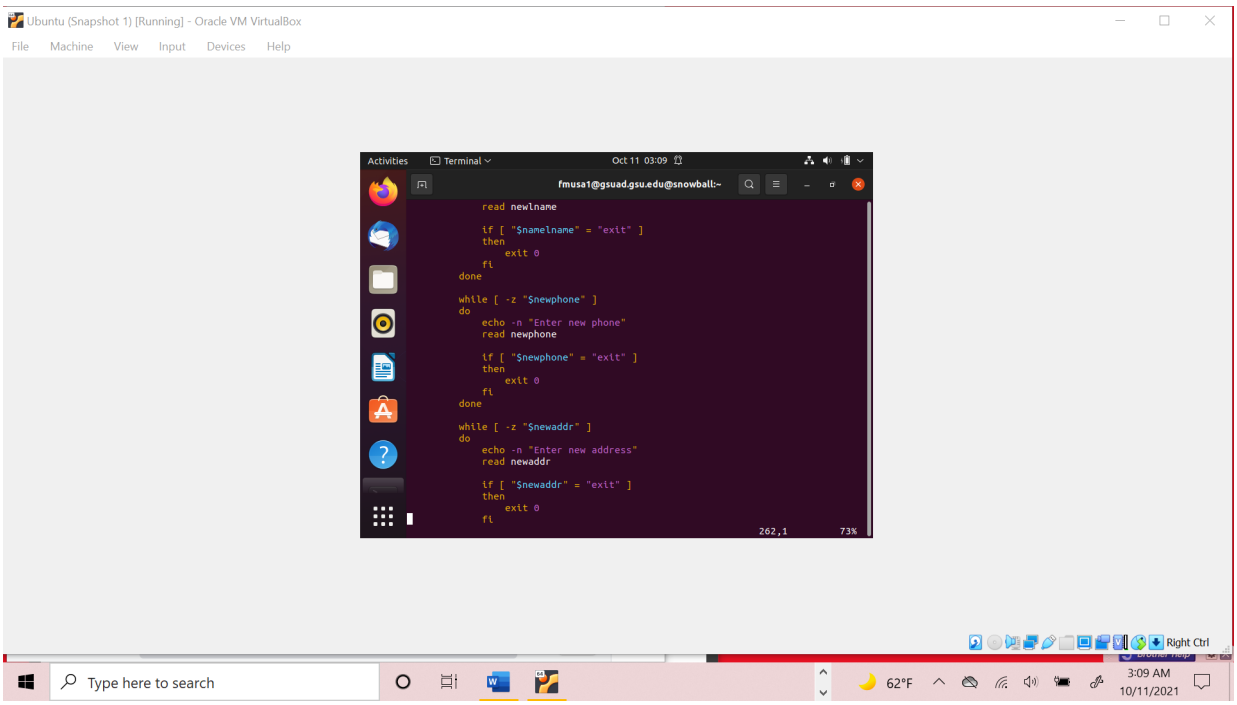
(I have been having trouble uploading the text file to this document for all my sh documents, sorry if it caused any inconvenience)

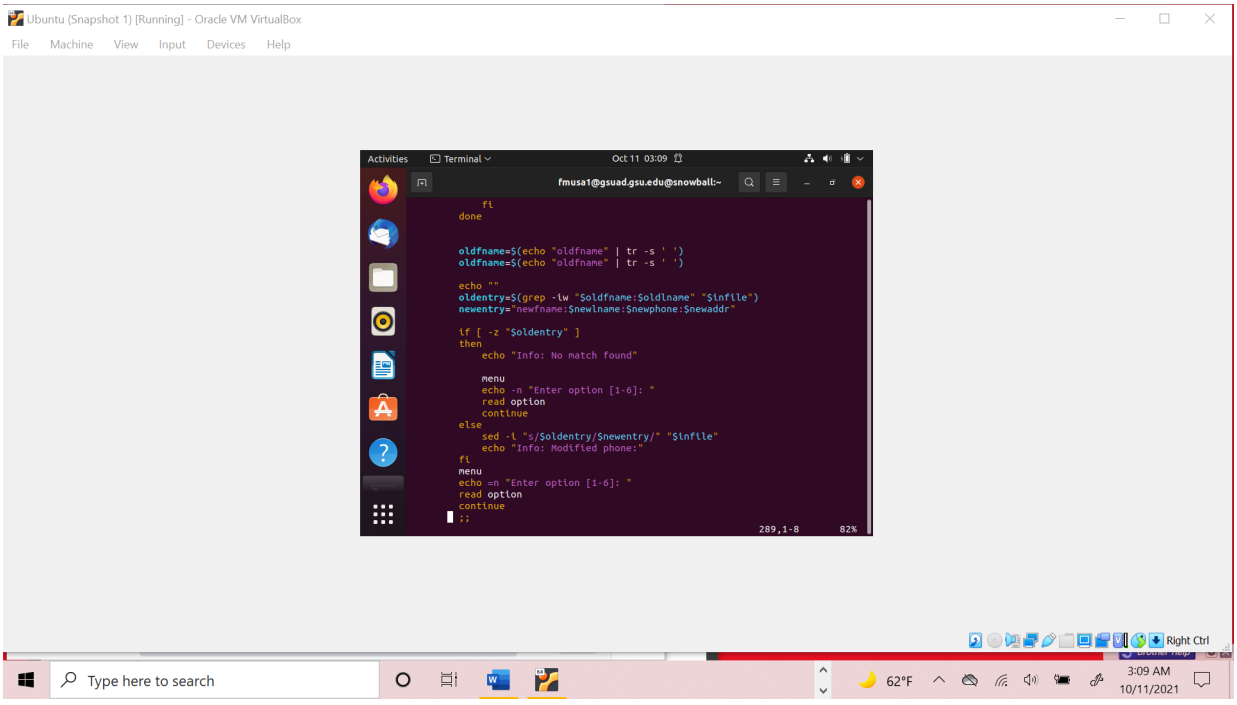


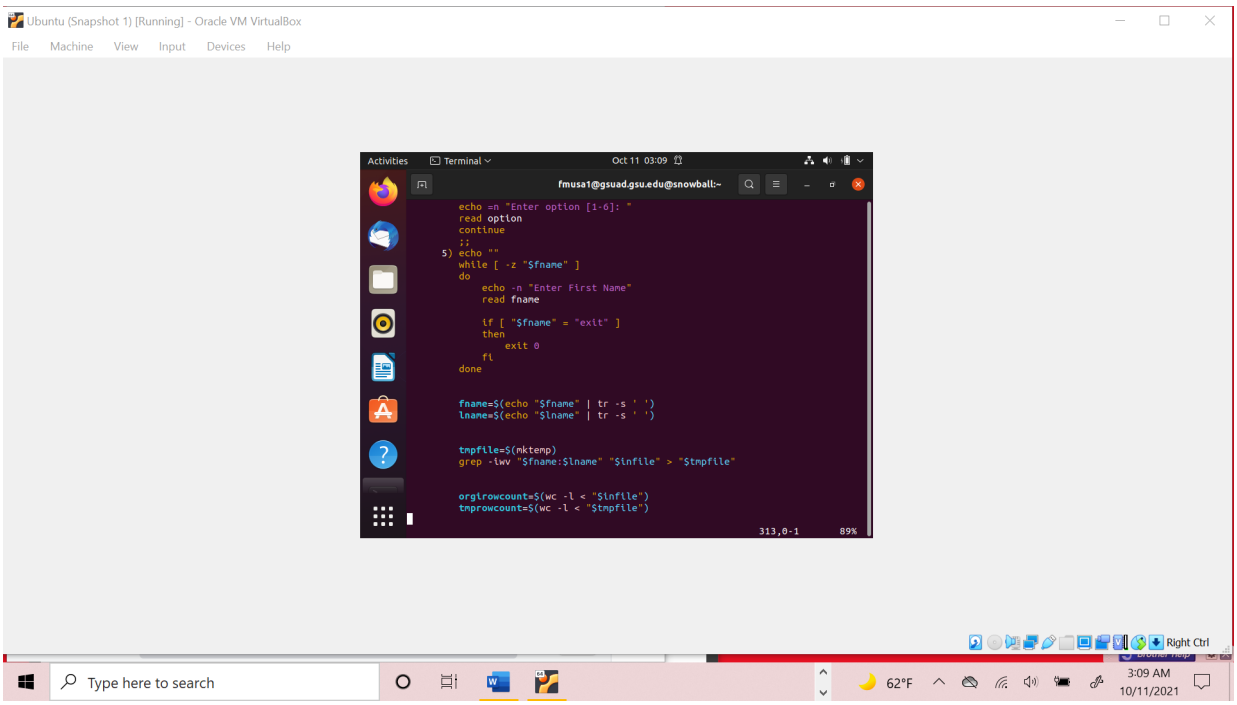

Ubuntu (Snapshot 1) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Oct 11 03:13
fmusa1@gsuad.gsu.edu@snowball:~
sum=0
l="y"
echo "enter one number"
read n1
echo "enter second number"
read n2
echo "1.addition"
echo "2.subtraction"
echo "3.multiplication"
echo "4.modulo"
echo "5.division"
echo "enter your choice"
read ch
case $ch in
1)sum=`expr $n1 + $n2`
echo "sum=" $sum;;
2) sum=`expr $n1 - $n2`
echo "sub = "$sum;;
3)sum=`expr $n1 * $n2`
echo "mul = " $sum;;
4) sum=`expr $n1 % $n2`
echo "mod = " $sum;;
5) sum= `expr $n1 / $n2`
echo "div = " $sum;;
*)echo "Invalid choice";;
esac
exit 0
"calculator.sh" 28L, 500C 4,1 All
Right Ctrl
Type here to search
62°F
3:13 AM
10/11/2021

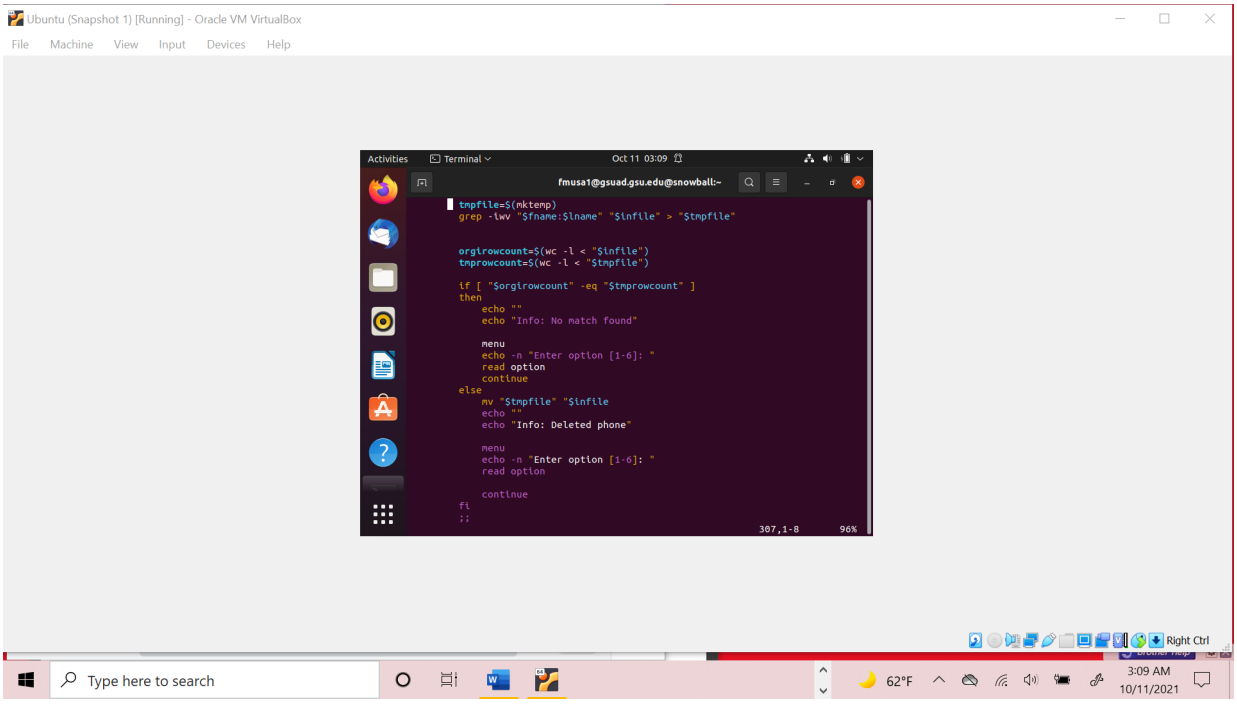
4. (20pts) Build a phone-book utility that allows you to access and modify an alphabetical list of names, addresses and telephone numbers. Use utilities such as awk and sed, to maintain and edit the file of phone-book information. The user (in this case, you) must be able to read, edit, and delete the phone book contents. The permissions for the phone book database must be such that it is inaccessible to anybody other than you (the user).

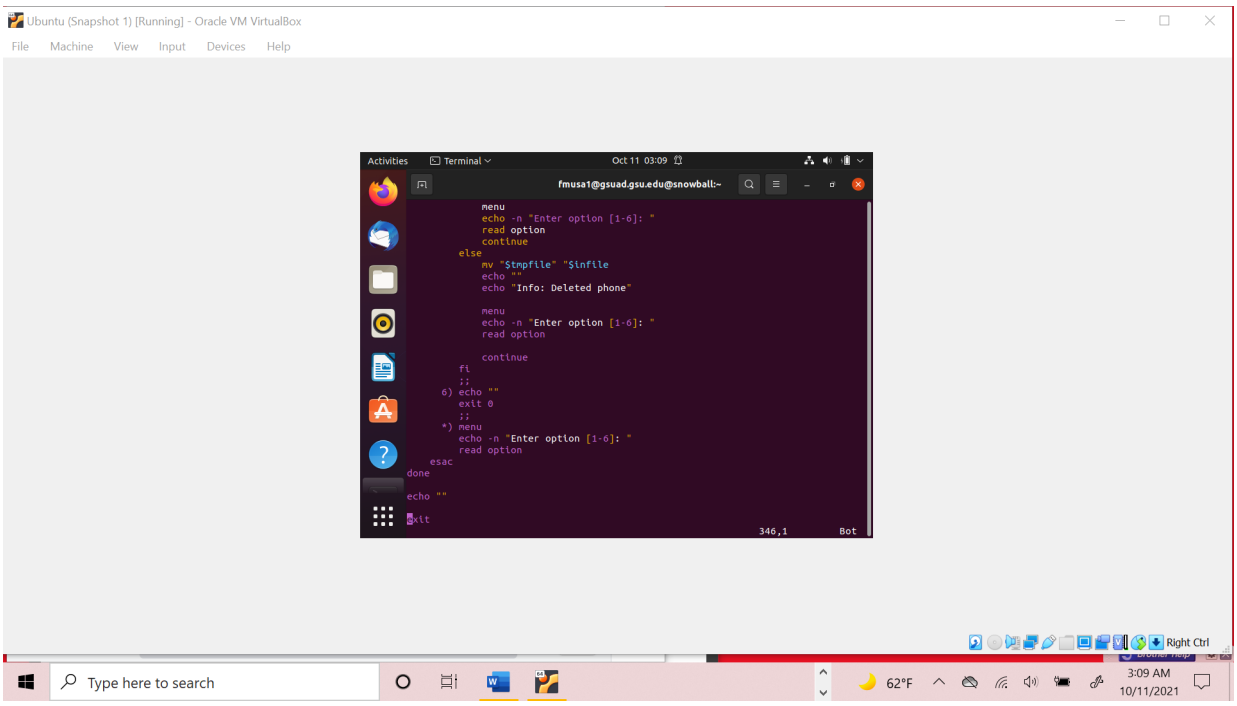


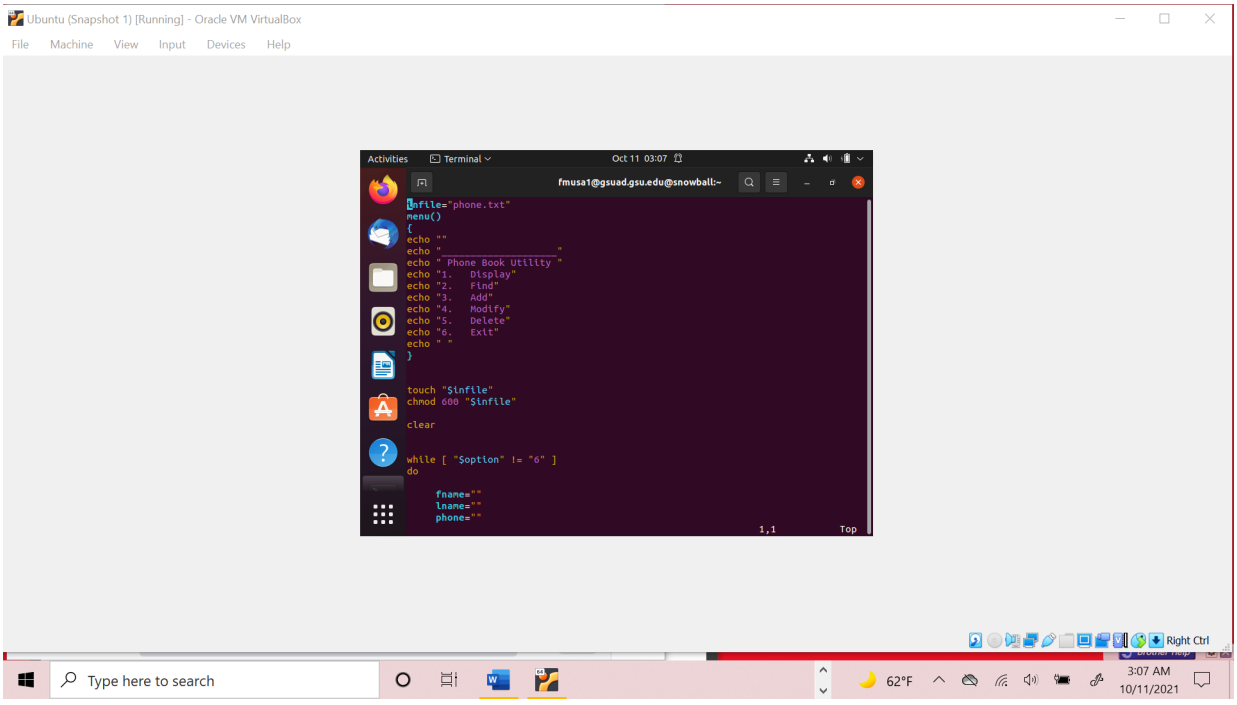

Ubuntu (Snapshot 1) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Oct 11 03:09
fmusa1@gsuad.gsu.edu@snowball:~
read newlname
if ["$namelname" = "exit"]
then
exit 0
fi
done
while [-z "$newphone"]
do
echo -n "Enter new phone"
read newphone
if ["$newphone" = "exit"]
then
exit 0
fi
done
while [-z "$newaddr"]
do
echo -n "Enter new address"
read newaddr
if ["$newaddr" = "exit"]
then
exit 0
fi
262,1 73%
Right Ctrl
Type here to search
62°F
3:09 AM
10/11/2021

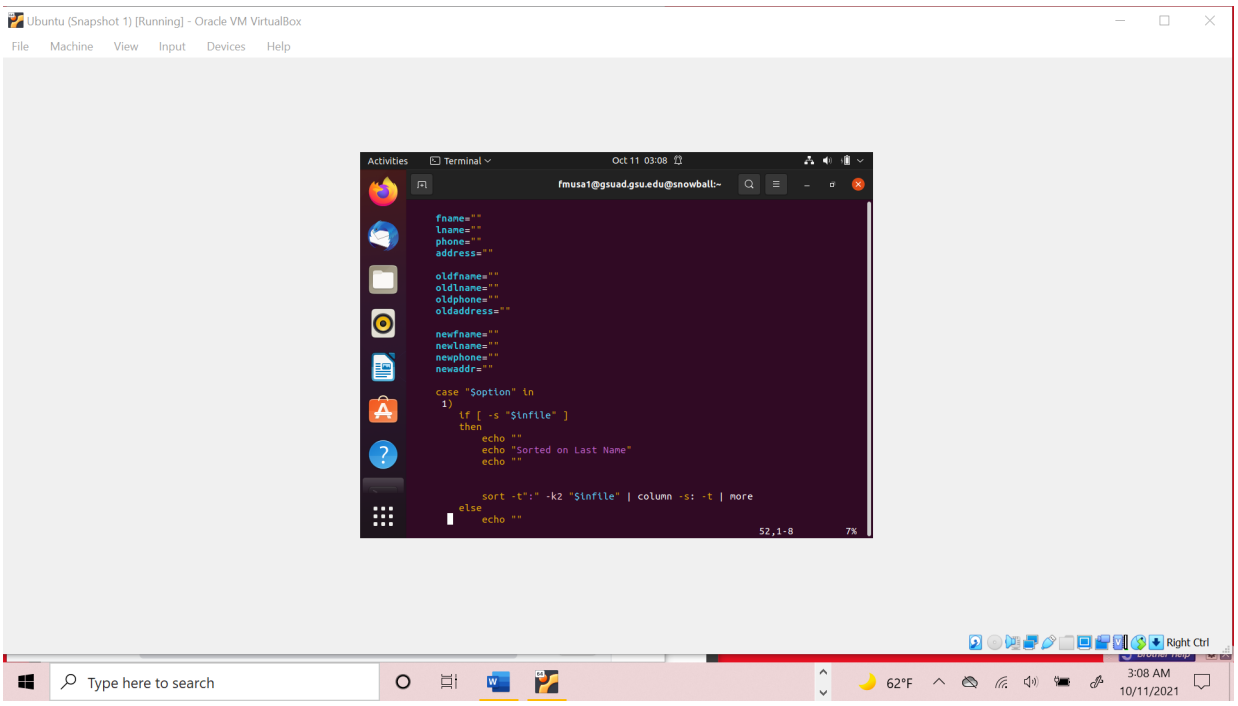

Ubuntu (Snapshot 1) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Oct 11 03:09
fmusa1@gsuad.gsu.edu@snowball:~
fi
done
oldfname=$(echo "oldfname" | tr -s ' ')
oldfname=$(echo "oldfname" | tr -s ' ')
echo ""
oldentry=$(grep -iw "$oldfname:$oldlname" "$infile")
newentry="newfname:$newlname:$newphone:$newaddr"
if [-z "$oldentry"]
then
echo "Info: No match found"
menu
echo -n "Enter option [1-6]: "
read option
continue
else
sed -i "s/$oldentry/$newentry/" "$infile"
echo "Info: Modified phone:"
fi
menu
echo -n "Enter option [1-6]: "
read option
continue
;;
289,1-8 82%
Right Ctrl
Type here to search
62°F
3:09 AM
10/11/2021

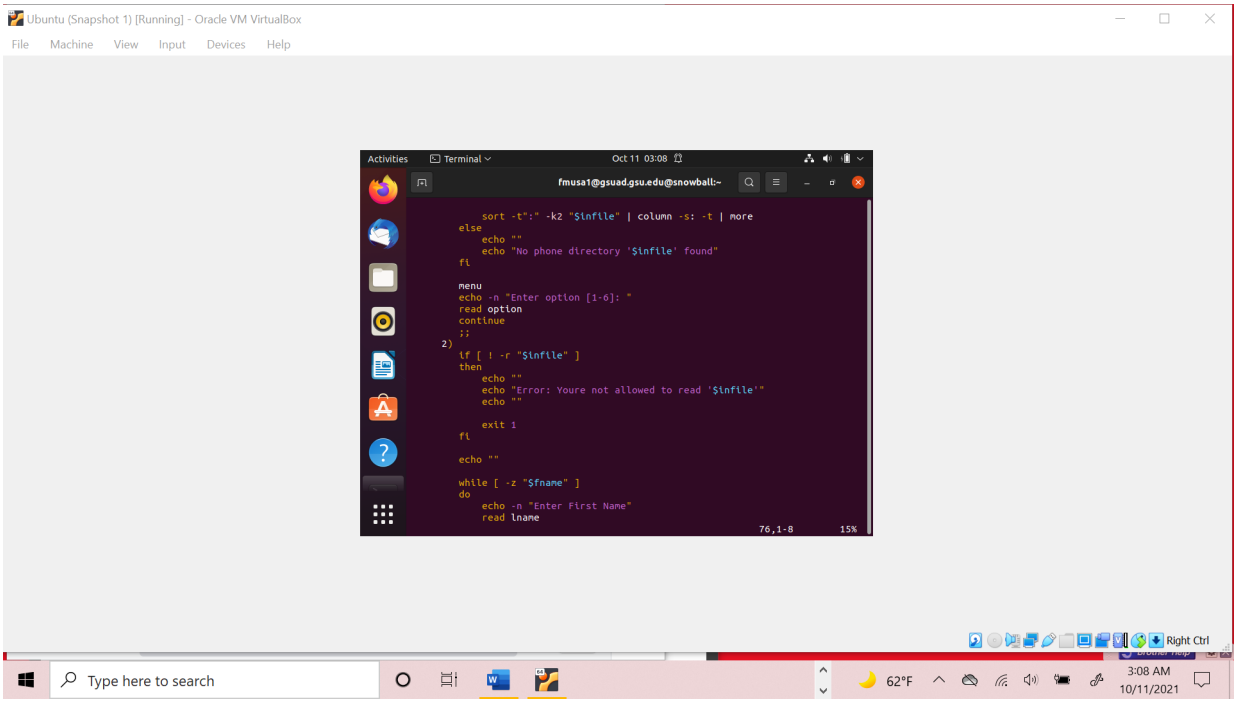

Ubuntu (Snapshot 1) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Oct 11 03:09
fmusa1@gsuad.gsu.edu@snowball:~
echo -n "Enter option [1-6]: "
read option
continue
;;
5) echo ""
while [-z "$fname"]
do
echo -n "Enter First Name"
read fname
if ["$fname" = "exit"]
then
exit 0
fi
done
fname=$(echo "$fname" | tr -s ' ')
lname=$(echo "$lname" | tr -s ' ')
tmpfile=$(mktemp)
grep -iwv "$fname:$lname" "$infile" > "$tmpfile"
orgirowcount=$(wc -l < "$infile")
tmprowcount=$(wc -l < "$tmpfile")
313,0-1 89%
Right Ctrl
Type here to search
62°F
3:09 AM
10/11/2021

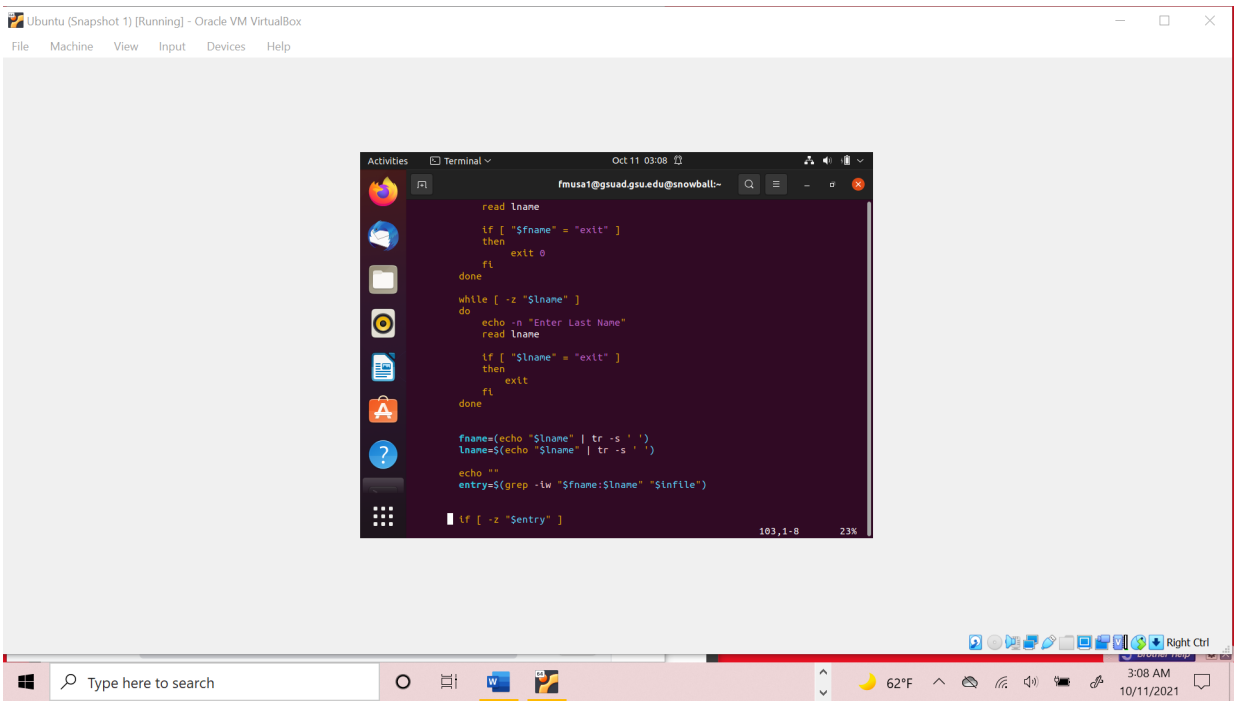

Ubuntu (Snapshot 1) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Oct 11 03:09
fmusa1@gsuad.gsu.edu@snowball:~
tmpfile=$(mktemp)
grep -iwv "$fname:$lname" "$infile" > "$tmpfile"
orgirowcount=$(wc -l < "$infile")
tmprowcount=$(wc -l < "$tmpfile")
if ["$orgirowcount" -eq "$tmprowcount"]
then
echo ""
echo "Info: No match found"
menu
echo -n "Enter option [1-6]: "
read option
continue
else
mv "$tmpfile" "$infile
echo ""
echo "Info: Deleted phone"
menu
echo -n "Enter option [1-6]: "
read option
continue
fi
;;
307,1-8 96%
Right Ctrl
Type here to search
62°F
3:09 AM
10/11/2021

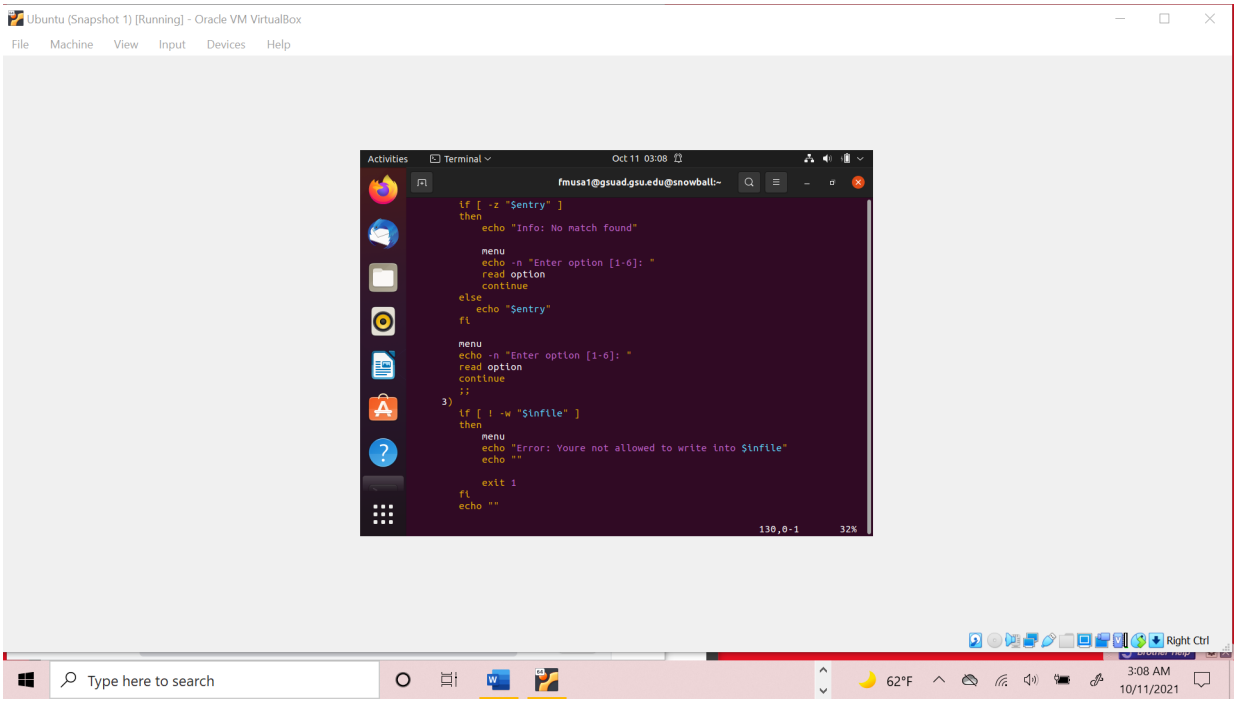

Ubuntu (Snapshot 1) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Oct 11 03:09
fmusa1@gsuad.gsu.edu@snowball:~
menu
echo -n "Enter option [1-6]: "
read option
continue
else
mv "$tmpfile" "$infile"
echo ""
echo "Info: Deleted phone"
menu
echo -n "Enter option [1-6]: "
read option
continue
fi
;;
6) echo ""
exit 0
;;
*) menu
echo -n "Enter option [1-6]: "
read option
esac
done
echo ""
exit
346,1 Bot
Right Ctrl
Type here to search
62°F
3:09 AM
10/11/2021


Ubuntu (Snapshot 1) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Oct 11 03:07
fmusa1@gsuad.gsu.edu@snowball:~
infile="phone.txt"
menu()
{
echo ""
echo "_____"
echo " Phone Book Utility "
echo "1. Display"
echo "2. Find"
echo "3. Add"
echo "4. Modify"
echo "5. Delete"
echo "6. Exit"
echo " "
}
touch "$infile"
chmod 600 "$infile"
clear
while ["$option" != "6"]
do
fname=""
lname=""
phone=""
1,1 Top
Right Ctrl
Type here to search
62°F
3:07 AM
10/11/2021


Ubuntu (Snapshot 1) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Oct 11 03:08
fmusa1@gsuad.gsu.edu@snowball:~
fname=""
lname=""
phone=""
address=""
oldfname=""
oldlname=""
oldphone=""
oldaddress=""
newfname=""
newlname=""
newphone=""
newaddr=""
case "$option" in
1)
if [-s "$infile"]
then
echo ""
echo "Sorted on Last Name"
echo ""
sort -t":" -k2 "$infile" | column -s: -t | more
else
echo ""
52,1-8 7%
Right Ctrl
Type here to search
62°F
3:08 AM
10/11/2021


Ubuntu (Snapshot 1) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Oct 11 03:08
fmusa1@gsuad.gsu.edu@snowball:~
sort -t":" -k2 "$infile" | column -s: -t | more
else
echo ""
echo "No phone directory '$infile' found"
fi
menu
echo -n "Enter option [1-6]: "
read option
continue
;;
2)
if [! -r "$infile"]
then
echo ""
echo "Error: Youre not allowed to read '$infile'"
echo ""
exit 1
fi
echo ""
while [-z "$fname"]
do
echo -n "Enter First Name"
read lname
76,1-8 15%
Right Ctrl
Type here to search
62°F
3:08 AM
10/11/2021


Ubuntu (Snapshot 1) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Oct 11 03:08
fmusa1@gsuad.gsu.edu@snowball:~
read lname
if ["$fname" = "exit"]
then
exit 0
fi
done
while [-z "$lname"]
do
echo -n "Enter Last Name"
read lname
if ["$lname" = "exit"]
then
exit
fi
done
fname=(echo "$lname" | tr -s ' ')
lname=$(echo "$lname" | tr -s ' ')
echo ""
entry=$(grep -iw "$fname:$lname" "$infile")
if [-z "$entry"]
103,1-8 23%
Right Ctrl
Type here to search
62°F
3:08 AM
10/11/2021


Ubuntu (Snapshot 1) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Oct 11 03:08
fmusa1@gsuad.gsu.edu@snowball:~
if [-z "$entry"]
then
echo "Info: No match found"
menu
echo -n "Enter option [1-6]: "
read option
continue
else
echo "$entry"
fi
menu
echo -n "Enter option [1-6]: "
read option
continue
;;
3)
if [! -w "$infile"]
then
menu
echo "Error: Youre not allowed to write into $infile"
echo ""
exit 1
fi
echo ""
130,0-1 32%
Right Ctrl
Type here to search
62°F
3:08 AM
10/11/2021

Ubuntu (Snapshot 1) [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Activities Terminal Oct 11 03:08

fmusa1@gsuad.gsu.edu@snowball:~

```
while [ -z "$fname" ]
do
    echo -n "Enter First Name"
    read fname

    if [ "$fname" = "exit" ]
    then
        exit 0
    fi
done

while [ -z "$lname" ]
do
    echo -n "Enter Last name"
    read lname

    if [ "$lname" = "exit" ]
    then
        exit 0
    fi
done

while [ -z "$phone" ]
do
    echo -n "Enter phone#"
    read phone
```

157,0-1 40%

Right Ctrl

Type here to search

62°F 3:08 AM 10/11/2021

Ubuntu (Snapshot 1) [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Activities Terminal Oct 11 03:09

fmusa1@gsuad.gsu.edu@snowball:~

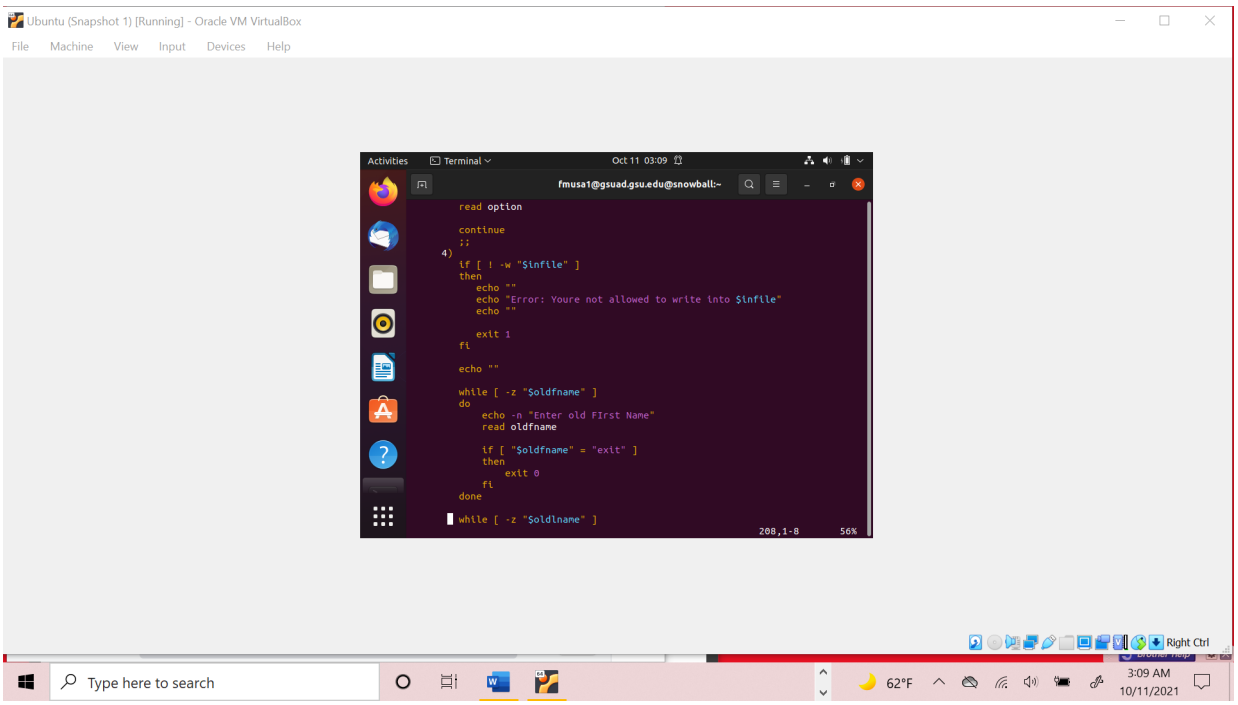
```
do
    echo -n "Enter phone#"
    read phone

    if [ "$phone" = "exit" ]
    then
        exit 0
    fi
done

while [ -z "$address" ]
do
    echo -n "Enter address"
    read address

    if [ "$address" = "exit" ]
    then
        exit 0
    fi
done

echo "$fname:$lname:$phone:$address" >> "$infile"
echo ""
echo "Info: New phone added"

menu
echo -n "Enter option [1-6]: "
read option
```

154,1 48%

Right Ctrl

Type here to search

62°F 3:09 AM 10/11/2021


Ubuntu (Snapshot 1) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Oct 11 03:09
fmusa1@gsuad.gsu.edu@snowball:~
read option
continue
;;
4)
if [! -w "$infile"]
then
echo ""
echo "Error: Youre not allowed to write into $infile"
echo ""
exit 1
fi
echo ""
while [-z "$oldfname"]
do
echo -n "Enter old FIrst Name"
read oldfname
if ["$oldfname" = "exit"]
then
exit 0
fi
done
while [-z "$oldlname"]
208,1-8 56%
Right Ctrl
Type here to search
62°F
3:09 AM
10/11/2021

5. (4 pts each) Give brief answers with examples, wherever relevant

A. What is the use of a shell?

Shell is a CLI (Command Line Interface) and it is between the users and the kernel and it interprets your commands and accepts it. A shell lets you run programs, build pipelines of processes, save output to files, and run more than one program at the same time. A shell executes all of the commands that you enter.

B. Is there any difference between the shell that you see on your PC versus that you see on the snowball server upon login. If yes, what are they? Provide screenshots for examples.

Each shell has its own programming language. I currently have a macbook which uses the Mac OS X and comes with the Bourne Again Shell (a.k.a Bash). In Linux we use the Bash or Bourne Again Shell. Theyre both using the same shell and you can compile all the same programs and utilities with no noticeable difference. but there are differences in user interfaces and with mac not having many open source applications built on open source libraries.

C. What are the elements in a computer (software and hardware) that enable the understanding and interpretation of a C program?

C is a compiled language with 2 parts compiling and linking and there is a compiler program that translates the C code inot machine instructions. C is not an interpreted language.

When you run a file the OS takes the file and puts the code and data segments into necessary locations in the memory RAM and tells the processor to run the code from a new memory location all while pulling the data from the assigned location.

D. The “printf()” C command is used for printing anything on the screen. In bash we use the command “echo ”. What is the difference (if any) in terms of how the computer interprets and executes these commands?

The prinff() command in C is uses the header library to print things on the screen for example, printf(“Hello World”) would print out Hello World. The echo command is used to output a line of text on the screen just like in C and can also be used to declare a variable. In C with printf when a function is called arguments are passed on the stack…..

E. What do these shell commands do? “ssh”, “scp” and “wget”. Describe briefly using an example that you have executed using the snowball server.

-Ssh (secure shell) is a network protocol used to securely communicate between computers, and transfers inputs from the client to the host and relays back the output

Ssh fmusa1@snowball.cs.gsu.edu

-scp (secure copy) is used to copy files between servers securely between two remote hosts and uses same security as ssh protocol.

Scp file.txt fmusa1@remotehost:/tmp/

-wget is a command-line utility used for downloading files from the web and you can download files using HTTP,HTTPS, FTP protocols and such. And by using wget it gives you a number of options that allows you to download multiple files.

wget [options] [url]