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: Vivian Do

Campus ID: vdo10

Panther #: 002486640

Oct 11 2021

Questions 1-5 are 20pts each

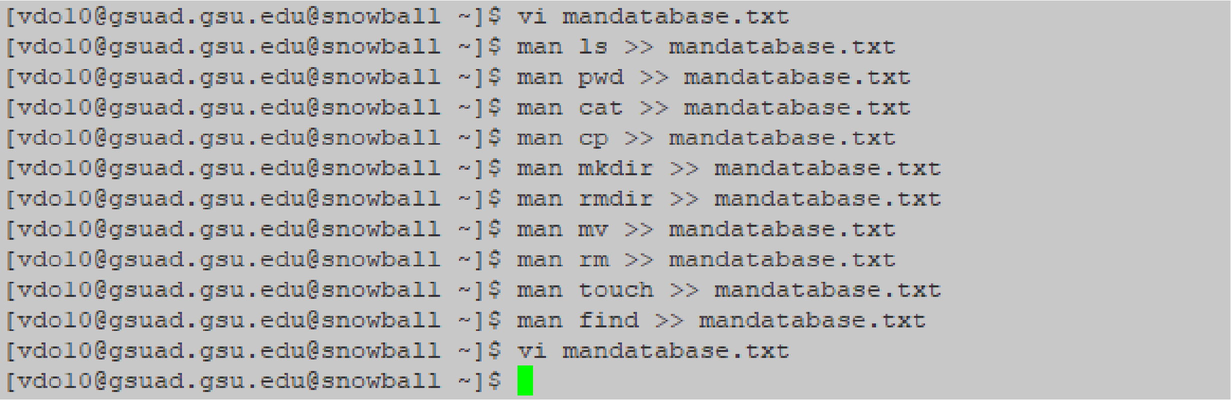
1. (20 pts) Pick any of your 10 favourite unix commands. For each command run the *man* command and copy the text that is printed into a mandatabase.txt. Write a shell script *helpme.sh* that will ask the user to type in a command and then print the manual's text associated with that corresponding command.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

   sorry, I cannot help you

   Files: mandatabase.txt, helpme.sh

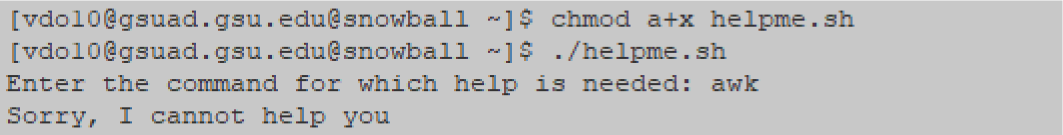
   My 10 Commands in mandatabase.txt:

   ls   pwd   cat   cp   mkdir   rmdir   mv   rm   touch   find



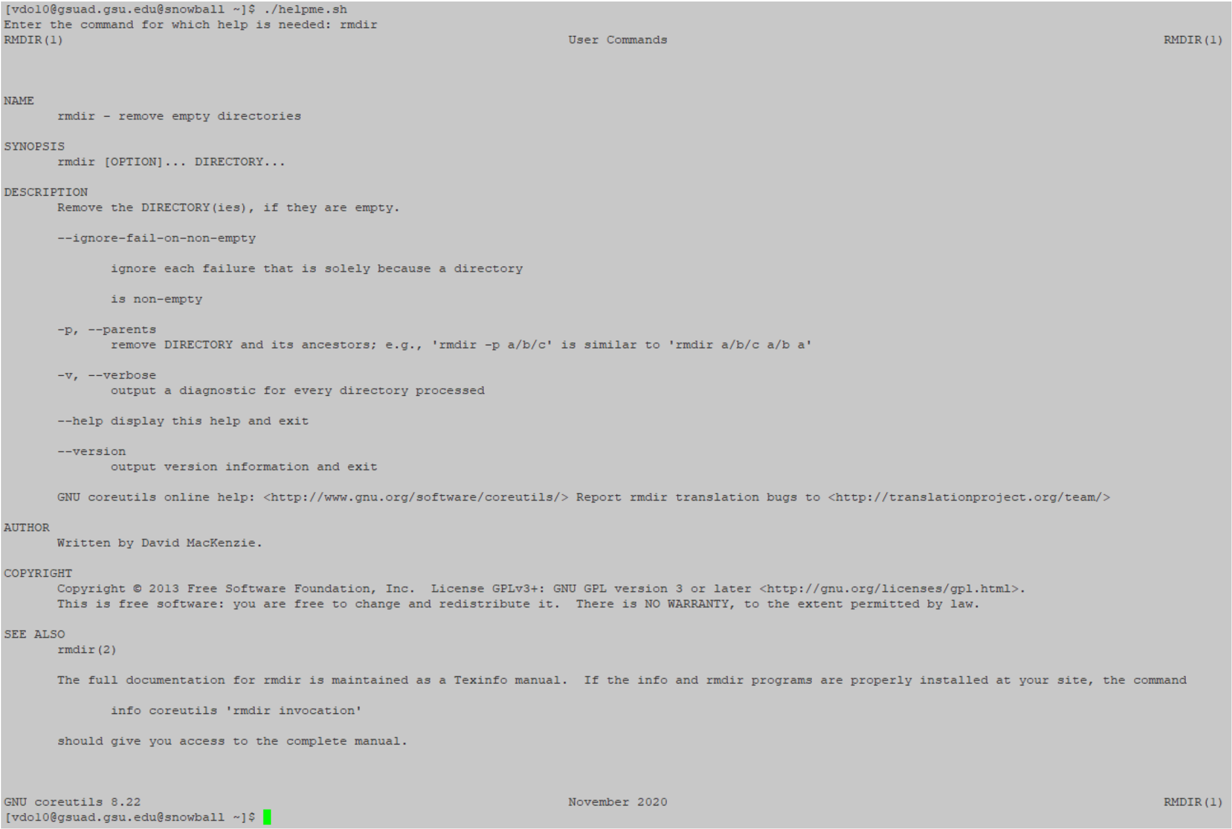
```
[vdo10@gsuad.gsu.edu@snowball ~]$ vi mandatabase.txt
[vdo10@gsuad.gsu.edu@snowball ~]$ man ls >> mandatabase.txt
[vdo10@gsuad.gsu.edu@snowball ~]$ man pwd >> mandatabase.txt
[vdo10@gsuad.gsu.edu@snowball ~]$ man cat >> mandatabase.txt
[vdo10@gsuad.gsu.edu@snowball ~]$ man cp >> mandatabase.txt
[vdo10@gsuad.gsu.edu@snowball ~]$ man mkdir >> mandatabase.txt
[vdo10@gsuad.gsu.edu@snowball ~]$ man rmdir >> mandatabase.txt
[vdo10@gsuad.gsu.edu@snowball ~]$ man mv >> mandatabase.txt
[vdo10@gsuad.gsu.edu@snowball ~]$ man rm >> mandatabase.txt
[vdo10@gsuad.gsu.edu@snowball ~]$ man touch >> mandatabase.txt
[vdo10@gsuad.gsu.edu@snowball ~]$ man find >> mandatabase.txt
[vdo10@gsuad.gsu.edu@snowball ~]$ vi mandatabase.txt
[vdo10@gsuad.gsu.edu@snowball ~]$
```

putting my 10 commands in mandatabase.txt by using man (command) >> mandatabase.txt



```
[vdo10@gsuad.gsu.edu@snowball ~]$ chmod a+x helpme.sh
[vdo10@gsuad.gsu.edu@snowball ~]$ ./helpme.sh
Enter the command for which help is needed: awk
Sorry, I cannot help you
```

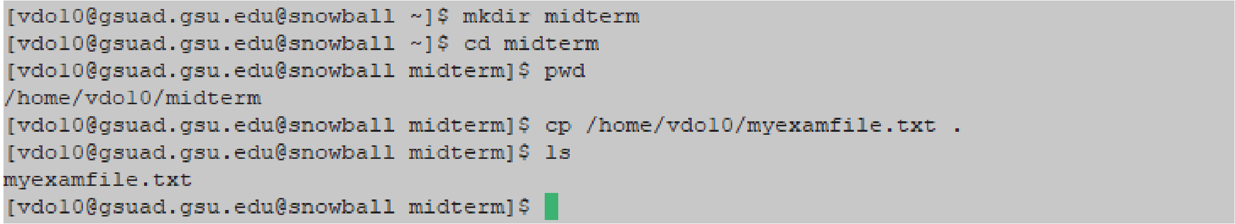
using the command chmod a+x to enable the command ./helpme.sh
checking to see if the code in the file helpme.sh works



```
[vdo10@gsuad.gsu.edu@snowball ~]$ ./helpme.sh
Enter the command for which help is needed: rmdir
RMDIR(1)                                                                  User Commands                                                                 RMDIR(1)



NAME
       rmdir - remove empty directories

SYNOPSIS
       rmdir [OPTION]... DIRECTORY...

DESCRIPTION
       Remove the DIRECTORY(ies), if they are empty.

       --ignore-fail-on-non-empty

              ignore each failure that is solely because a directory

              is non-empty

       -p, --parents
              remove DIRECTORY and its ancestors; e.g., 'rmdir -p a/b/c' is similar to 'rmdir a/b/c a/b a'

       -v, --verbose
              output a diagnostic for every directory processed

       --help display this help and exit

       --version
              output version information and exit

       GNU coreutils online help: <http://www.gnu.org/software/coreutils/> Report rmdir translation bugs to <http://translationproject.org/team/>

AUTHOR
       Written by David MacKenzie.

COPYRIGHT
       Copyright © 2013 Free Software Foundation, Inc.  License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>.
       This is free software: you are free to change and redistribute it.  There is NO WARRANTY, to the extent permitted by law.

SEE ALSO
       rmdir(2)

       The full documentation for rmdir is maintained as a Texinfo manual.  If the info and rmdir programs are properly installed at your site, the command

              info coreutils 'rmdir invocation'

       should give you access to the complete manual.



GNU coreutils 8.22                                                        November 2020                                                                 RMDIR(1)
[vdo10@gsuad.gsu.edu@snowball ~]$
```

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).

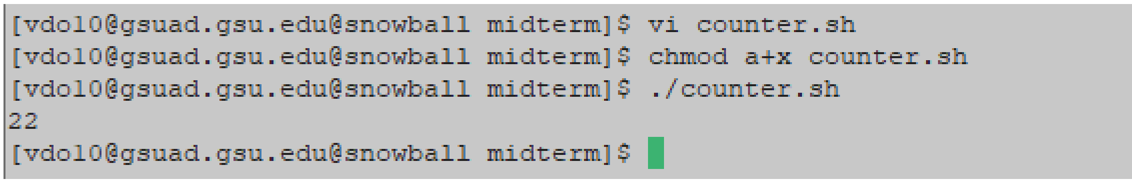
Files: myexamfile.txt, counter.sh
creating a new directory using mkdir command
copying the myexamfile.txt into midterm directory using cp command
using cd to go to midterm directory



```
[vdol0@gsuad.gsu.edu@snowball ~]$ mkdir midterm
[vdol0@gsuad.gsu.edu@snowball ~]$ cd midterm
[vdol0@gsuad.gsu.edu@snowball midterm]$ pwd
/home/vdol0/midterm
[vdol0@gsuad.gsu.edu@snowball midterm]$ cp /home/vdol0/myexamfile.txt .
[vdol0@gsuad.gsu.edu@snowball midterm]$ ls
myexamfile.txt
[vdol0@gsuad.gsu.edu@snowball midterm]$
```

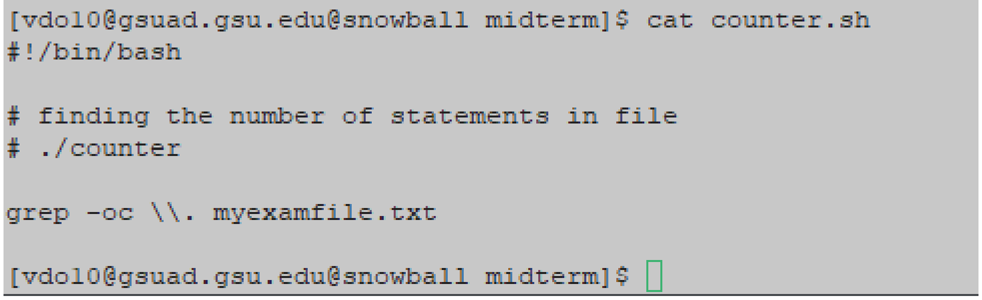
a. Write a shell script that will find the number of statements in the text. A statement is defined as the collection of text between two periods (full-stops).

creating a shell script file to count the number of statements in myexamfile.txt using vi editor
using the chmod a+x command to enable the ./counter.sh command
checking to see if ./counter.sh works



```
[vdol0@gsuad.gsu.edu@snowball midterm]$ vi counter.sh
[vdol0@gsuad.gsu.edu@snowball midterm]$ chmod a+x counter.sh
[vdol0@gsuad.gsu.edu@snowball midterm]$ ./counter.sh
22
[vdol0@gsuad.gsu.edu@snowball midterm]$
```

Since the original number of sentences for myexamfile.txt is ~22, this shell script works!

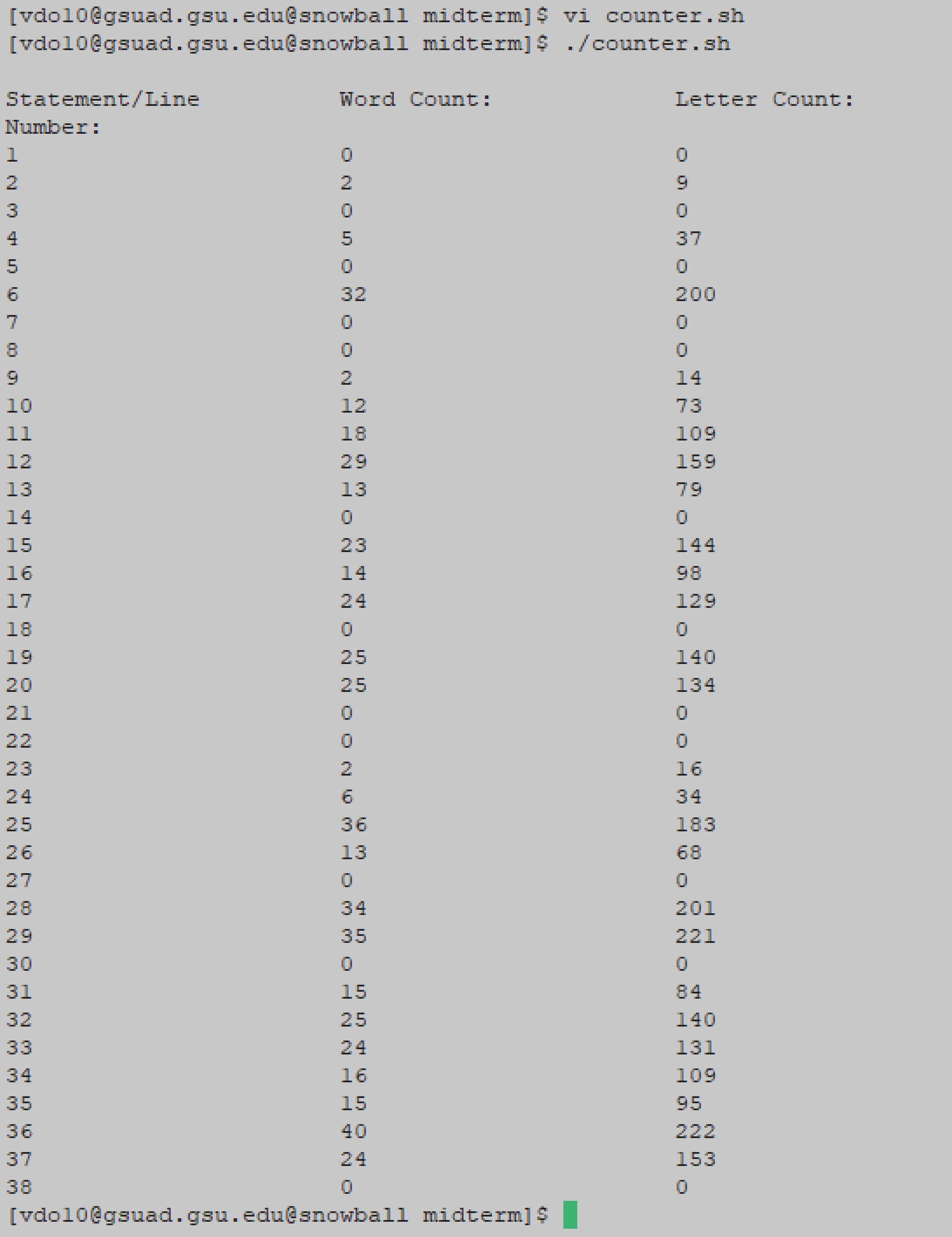


```
[vdol0@gsuad.gsu.edu@snowball midterm]$ cat counter.sh
#!/bin/bash

# finding the number of statements in file
# ./counter

grep -oc \\. myexamfile.txt

[vdol0@gsuad.gsu.edu@snowball midterm]$ 
```

^ counter.sh file for part a

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

going back into the counter.sh file using vi editor again



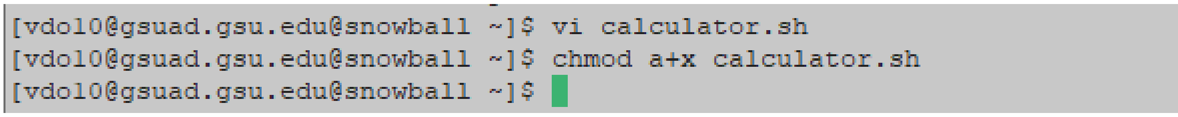
```
[vdol0@gsuad.gsu.edu@snowball midterm]$ vi counter.sh
[vdol0@gsuad.gsu.edu@snowball midterm]$ ./counter.sh

Statement/Line          Word Count:             Letter Count:
Number:
1                       0                       0
2                       2                       9
3                       0                       0
4                       5                       37
5                       0                       0
6                       32                      200
7                       0                       0
8                       0                       0
9                       2                       14
10                      12                      73
11                      18                      109
12                      29                      159
13                      13                      79
14                      0                       0
15                      23                      144
16                      14                      98
17                      24                      129
18                      0                       0
19                      25                      140
20                      25                      134
21                      0                       0
22                      0                       0
23                      2                       16
24                      6                       34
25                      36                      183
26                      13                      68
27                      0                       0
28                      34                      201
29                      35                      221
30                      0                       0
31                      15                      84
32                      25                      140
33                      24                      131
34                      16                      109
35                      15                      95
36                      40                      222
37                      24                      153
38                      0                       0
[vdol0@gsuad.gsu.edu@snowball midterm]$
```

^ results of my tabular

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.

Files: calculator.sh

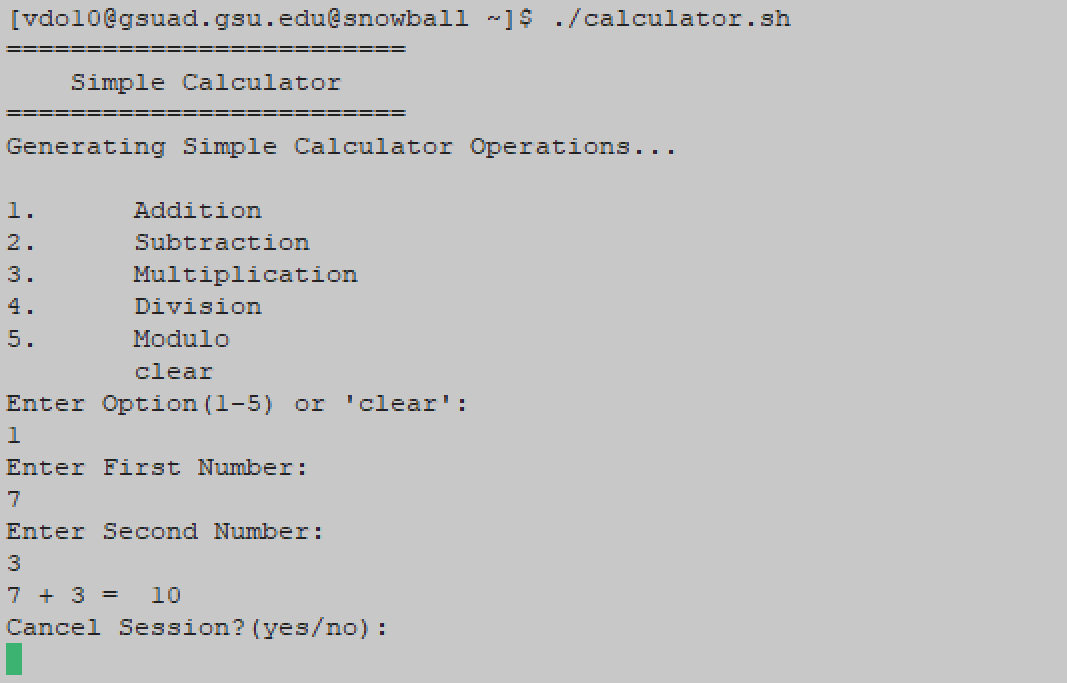
creating a shell script for my simple calculator using vi editor
using the chmod a+x command to enable the ./calculator.sh command



```
[vdol0@gsuad.gsu.edu@snowball ~]$ vi calculator.sh
[vdol0@gsuad.gsu.edu@snowball ~]$ chmod a+x calculator.sh
[vdol0@gsuad.gsu.edu@snowball ~]$
```

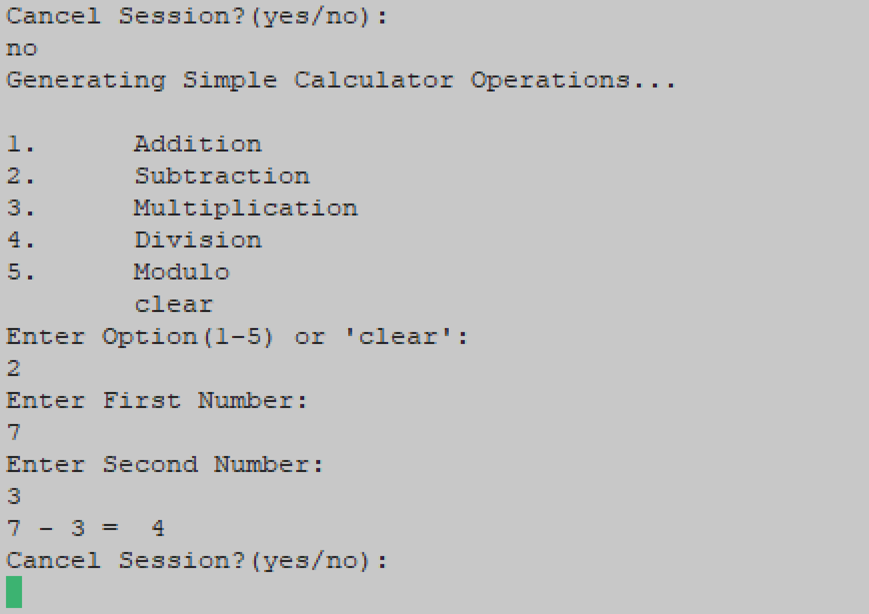
testing each option to see if there's any problems with the script

- addition (7 + 3)



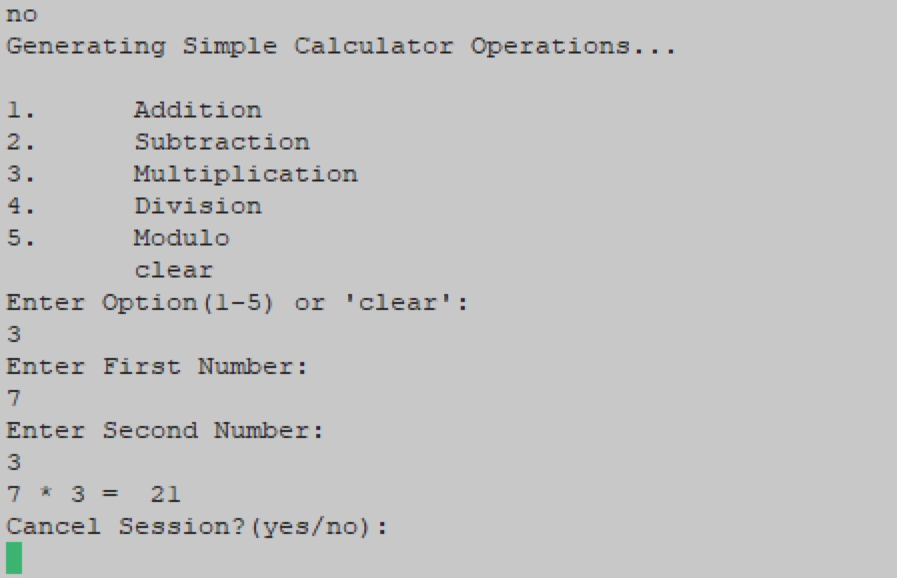
```
[vdol0@gsuad.gsu.edu@snowball ~]$ ./calculator.sh
=========================
    Simple Calculator
=========================
Generating Simple Calculator Operations...

1.      Addition
2.      Subtraction
3.      Multiplication
4.      Division
5.      Modulo
        clear
Enter Option(1-5) or 'clear':
1
Enter First Number:
7
Enter Second Number:
3
7 + 3 =  10
Cancel Session?(yes/no):
```

- subtraction (7 - 3)



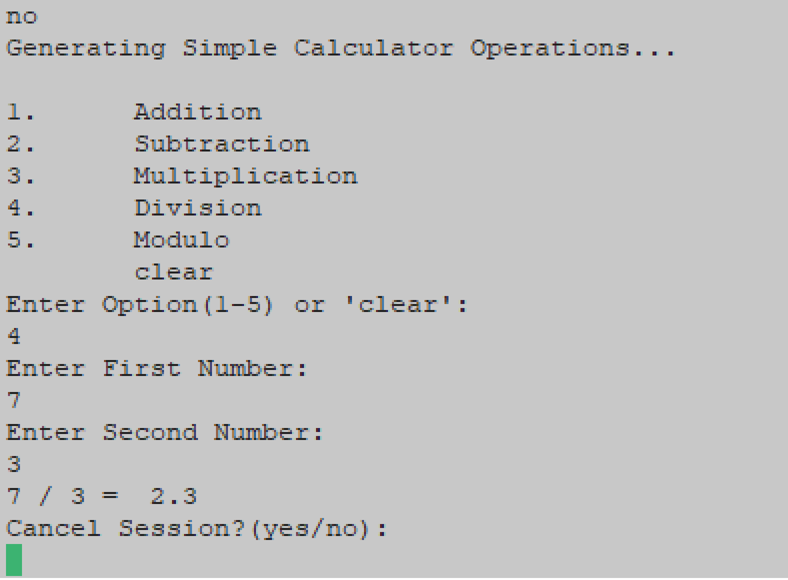
```
Cancel Session?(yes/no):
no
Generating Simple Calculator Operations...

1.      Addition
2.      Subtraction
3.      Multiplication
4.      Division
5.      Modulo
        clear
Enter Option(1-5) or 'clear':
2
Enter First Number:
7
Enter Second Number:
3
7 - 3 =  4
Cancel Session?(yes/no):
```

- multiplication (7 * 3)



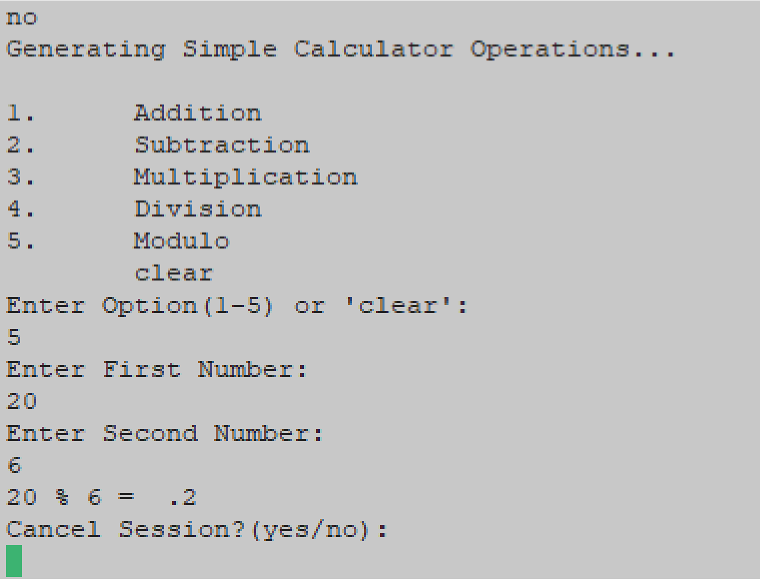
```
no
Generating Simple Calculator Operations...

1.      Addition
2.      Subtraction
3.      Multiplication
4.      Division
5.      Modulo
        clear
Enter Option(1-5) or 'clear':
3
Enter First Number:
7
Enter Second Number:
3
7 * 3 =  21
Cancel Session?(yes/no):
```

- division (7 / 3)



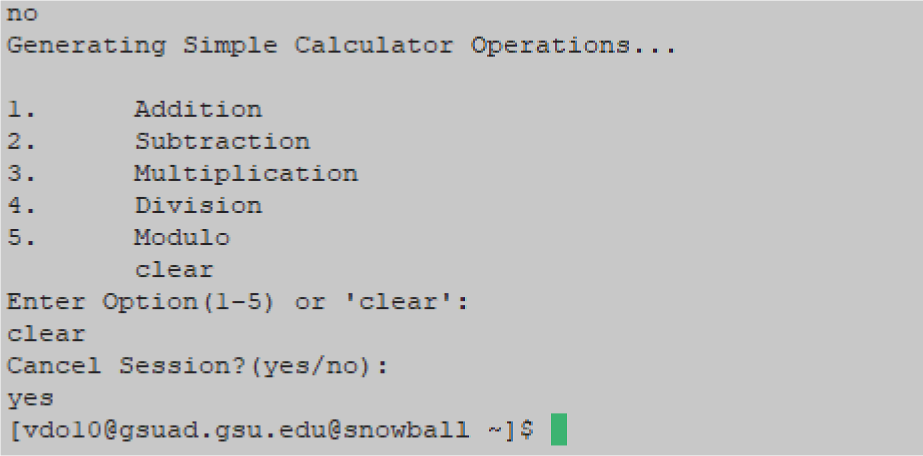
```
no
Generating Simple Calculator Operations...

1.       Addition
2.       Subtraction
3.       Multiplication
4.       Division
5.       Modulo
         clear
Enter Option(1-5) or 'clear':
4
Enter First Number:
7
Enter Second Number:
3
7 / 3 =  2.3
Cancel Session?(yes/no):
```

- modulo (20 % 6)



```
no
Generating Simple Calculator Operations...

1.       Addition
2.       Subtraction
3.       Multiplication
4.       Division
5.       Modulo
         clear
Enter Option(1-5) or 'clear':
5
Enter First Number:
20
Enter Second Number:
6
20 % 6 =  .2
Cancel Session?(yes/no):
```

- clear (this will just clean up the whole space)



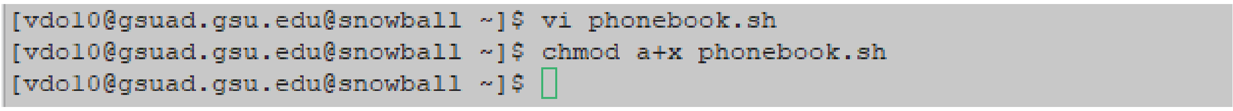
```
no
Generating Simple Calculator Operations...

1.      Addition
2.      Subtraction
3.      Multiplication
4.      Division
5.      Modulo
        clear
Enter Option(1-5) or 'clear':
clear
Cancel Session?(yes/no):
yes
[vdo10@gsuad.gsu.edu@snowball ~]$
```

* Note: At the end of every operation, there is always a prompt asking the user if they wish to stop using the calculator or to continue using the calculator. To cancel, option is “yes”; to continue, option is “no”.

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).

Files: phone.sh, phonebook.txt

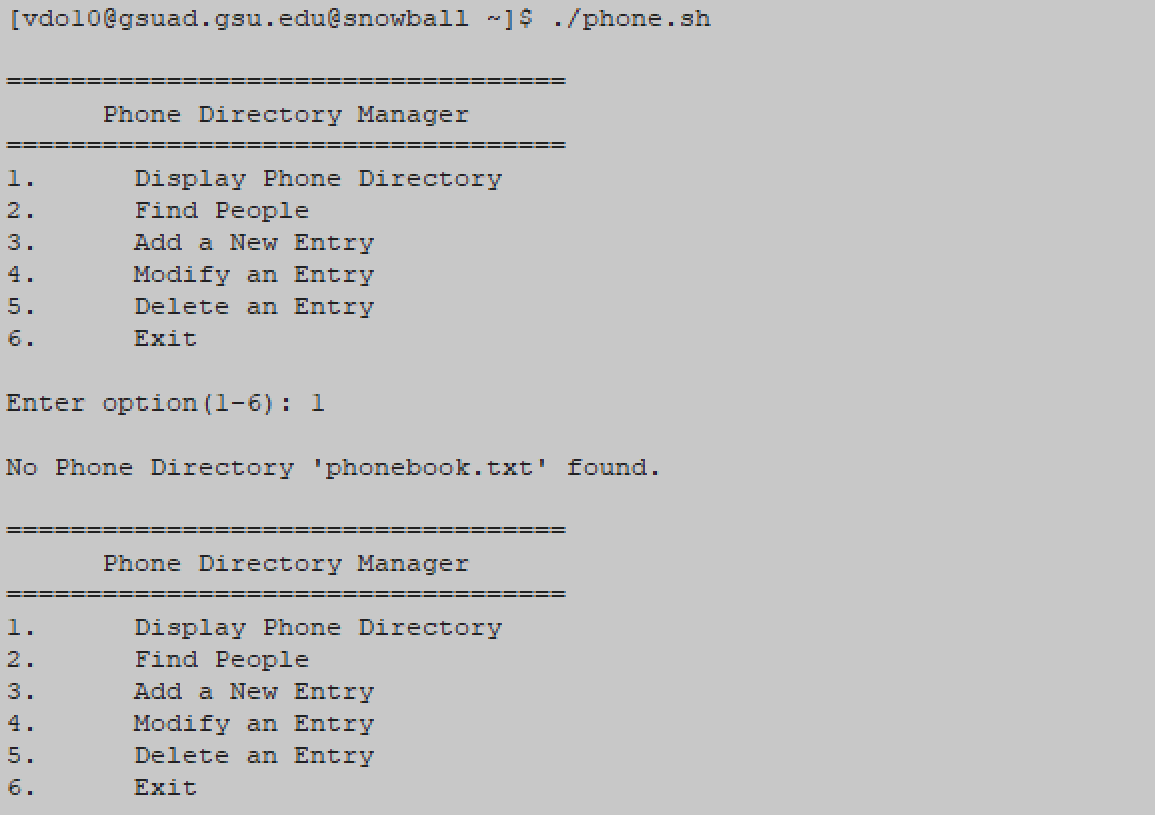
creating a phone book utility using vi editor
using the chmod a+x command to enable the ./phonebook.sh command



```
[vdol0@gsuad.gsu.edu@snowball ~]$ vi phonebook.sh
[vdol0@gsuad.gsu.edu@snowball ~]$ chmod a+x phonebook.sh
[vdol0@gsuad.gsu.edu@snowball ~]$ 
```

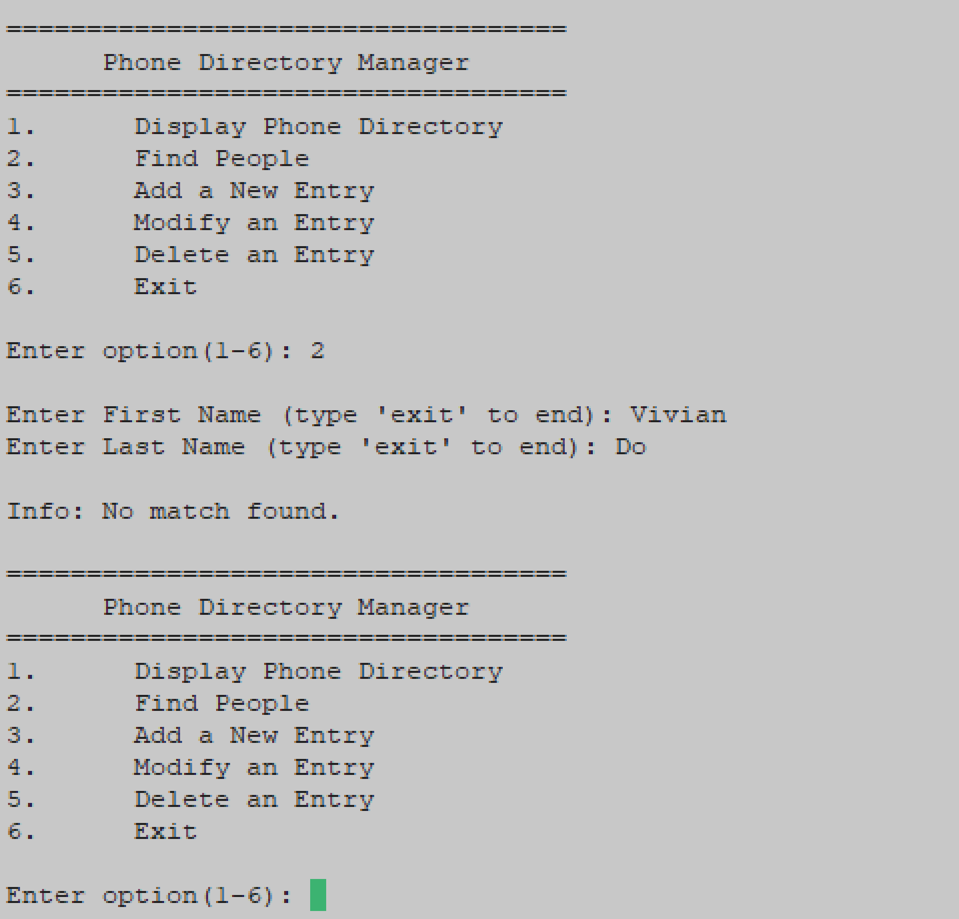
testing each option to see if there’s any problems with the script

- after each entry of a new name, the phonebook.txt will constantly get updated as many times as you make the updates through the phonebook.sh. The phonebook.txt will be created once the first entry has been made.



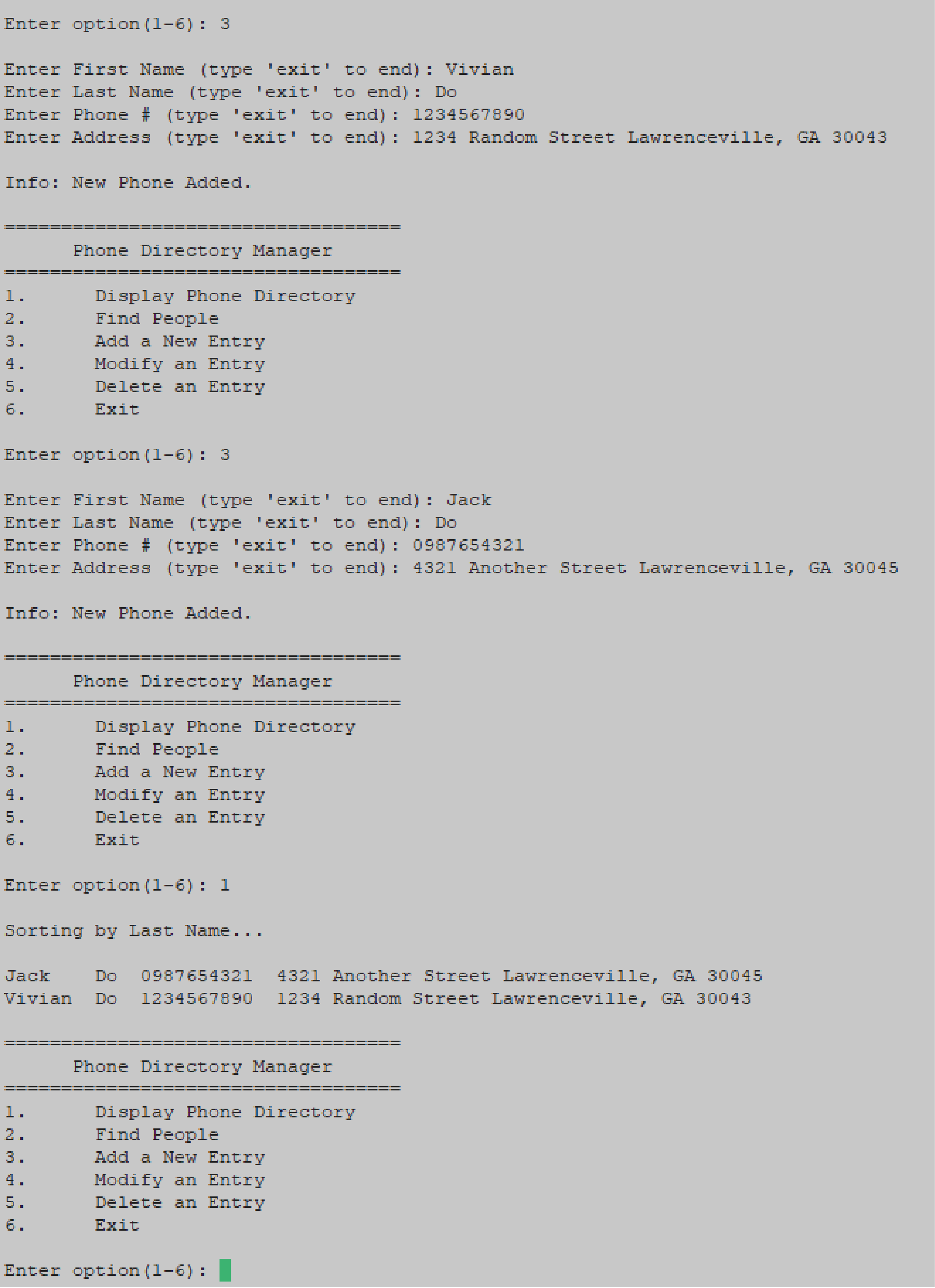
```
[vdol0@gsuad.gsu.edu@snowball ~]$ ./phone.sh

===================================
      Phone Directory Manager
===================================
1.      Display Phone Directory
2.      Find People
3.      Add a New Entry
4.      Modify an Entry
5.      Delete an Entry
6.      Exit

Enter option(1-6): 1

No Phone Directory 'phonebook.txt' found.

===================================
      Phone Directory Manager
===================================
1.      Display Phone Directory
2.      Find People
3.      Add a New Entry
4.      Modify an Entry
5.      Delete an Entry
6.      Exit
```

^ empty phonebook directory since it’s my first time using it



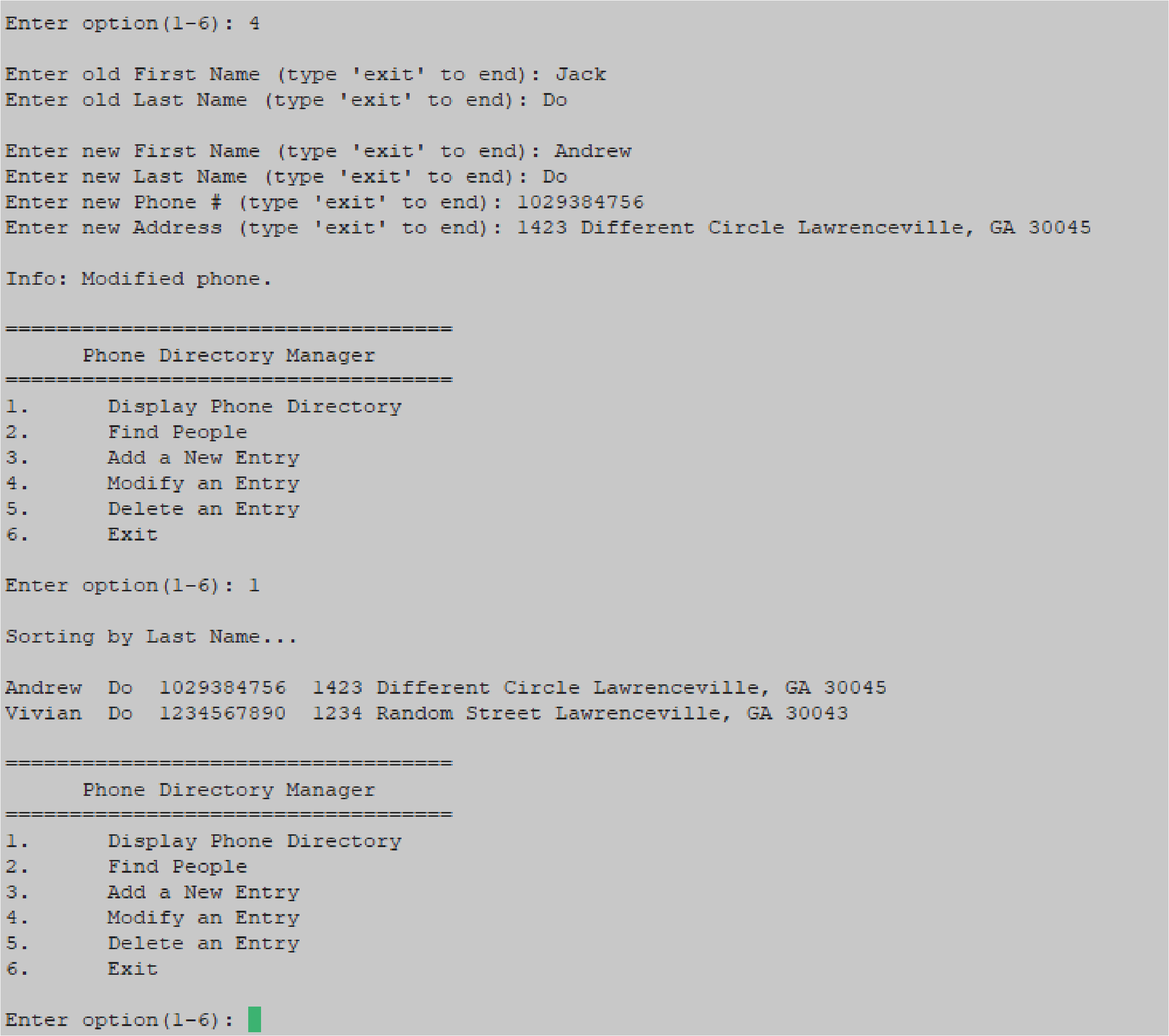
```
===================================
      Phone Directory Manager
===================================
1.      Display Phone Directory
2.      Find People
3.      Add a New Entry
4.      Modify an Entry
5.      Delete an Entry
6.      Exit

Enter option(1-6): 2

Enter First Name (type 'exit' to end): Vivian
Enter Last Name (type 'exit' to end): Do

Info: No match found.

===================================
      Phone Directory Manager
===================================
1.      Display Phone Directory
2.      Find People
3.      Add a New Entry
4.      Modify an Entry
5.      Delete an Entry
6.      Exit

Enter option(1-6): █
```

^ test-searching myself up in the phonebook for the first time



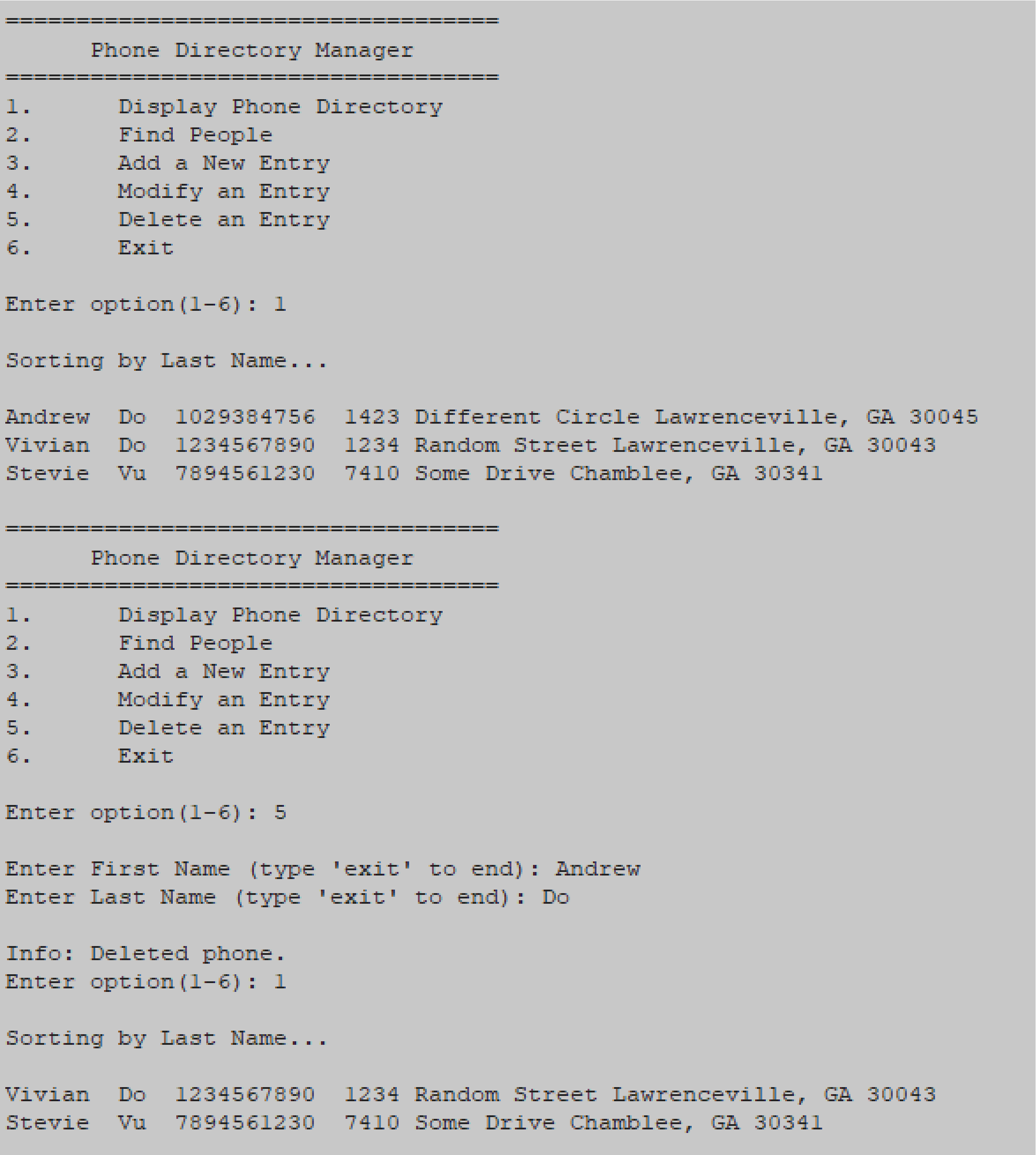
```
Enter option(1-6): 3

Enter First Name (type 'exit' to end): Vivian
Enter Last Name (type 'exit' to end): Do
Enter Phone # (type 'exit' to end): 1234567890
Enter Address (type 'exit' to end): 1234 Random Street Lawrenceville, GA 30043

Info: New Phone Added.

===================================
      Phone Directory Manager
===================================
1.      Display Phone Directory
2.      Find People
3.      Add a New Entry
4.      Modify an Entry
5.      Delete an Entry
6.      Exit

Enter option(1-6): 3

Enter First Name (type 'exit' to end): Jack
Enter Last Name (type 'exit' to end): Do
Enter Phone # (type 'exit' to end): 0987654321
Enter Address (type 'exit' to end): 4321 Another Street Lawrenceville, GA 30045

Info: New Phone Added.

===================================
      Phone Directory Manager
===================================
1.      Display Phone Directory
2.      Find People
3.      Add a New Entry
4.      Modify an Entry
5.      Delete an Entry
6.      Exit

Enter option(1-6): 1

Sorting by Last Name...

Jack    Do  0987654321  4321 Another Street Lawrenceville, GA 30045
Vivian  Do  1234567890  1234 Random Street Lawrenceville, GA 30043

===================================
      Phone Directory Manager
===================================
1.      Display Phone Directory
2.      Find People
3.      Add a New Entry
4.      Modify an Entry
5.      Delete an Entry
6.      Exit

Enter option(1-6):
```

^ adding myself and my cousin to the phonebook with fake phone numbers and addresses



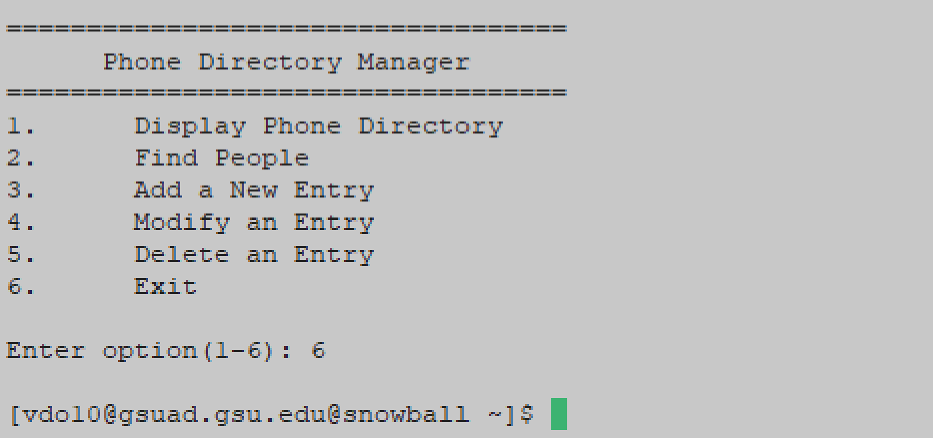
```
Enter option(1-6): 4

Enter old First Name (type 'exit' to end): Jack
Enter old Last Name (type 'exit' to end): Do

Enter new First Name (type 'exit' to end): Andrew
Enter new Last Name (type 'exit' to end): Do
Enter new Phone # (type 'exit' to end): 1029384756
Enter new Address (type 'exit' to end): 1423 Different Circle Lawrenceville, GA 30045

Info: Modified phone.

===================================
      Phone Directory Manager
===================================
1.      Display Phone Directory
2.      Find People
3.      Add a New Entry
4.      Modify an Entry
5.      Delete an Entry
6.      Exit

Enter option(1-6): 1

Sorting by Last Name...

Andrew  Do  1029384756  1423 Different Circle Lawrenceville, GA 30045
Vivian  Do  1234567890  1234 Random Street Lawrenceville, GA 30043

===================================
      Phone Directory Manager
===================================
1.      Display Phone Directory
2.      Find People
3.      Add a New Entry
4.      Modify an Entry
5.      Delete an Entry
6.      Exit

Enter option(1-6):
```

^ modified Jack to Andrew in the phonebook



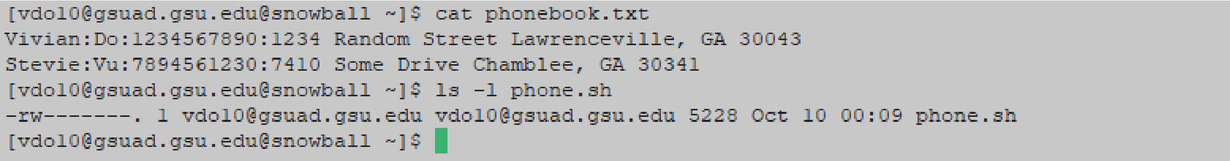
```
===================================
      Phone Directory Manager
===================================
1.      Display Phone Directory
2.      Find People
3.      Add a New Entry
4.      Modify an Entry
5.      Delete an Entry
6.      Exit

Enter option(1-6): 1

Sorting by Last Name...

Andrew  Do  1029384756  1423 Different Circle Lawrenceville, GA 30045
Vivian  Do  1234567890  1234 Random Street Lawrenceville, GA 30043
Stevie  Vu  7894561230  7410 Some Drive Chamblee, GA 30341

===================================
      Phone Directory Manager
===================================
1.      Display Phone Directory
2.      Find People
3.      Add a New Entry
4.      Modify an Entry
5.      Delete an Entry
6.      Exit

Enter option(1-6): 5

Enter First Name (type 'exit' to end): Andrew
Enter Last Name (type 'exit' to end): Do

Info: Deleted phone.
Enter option(1-6): 1

Sorting by Last Name...

Vivian  Do  1234567890  1234 Random Street Lawrenceville, GA 30043
Stevie  Vu  7894561230  7410 Some Drive Chamblee, GA 30341
```

^ added one more person to the list and sorted the list by last names; by pressing option 5, someone was deleted from the list



```
===================================
      Phone Directory Manager
===================================
1.      Display Phone Directory
2.      Find People
3.      Add a New Entry
4.      Modify an Entry
5.      Delete an Entry
6.      Exit

Enter option(1-6): 6

[vdol0@gsuad.gsu.edu@snowball ~]$
```

^ exiting phonebook once finished



```
[vdol0@gsuad.gsu.edu@snowball ~]$ cat phonebook.txt
Vivian:Do:1234567890:1234 Random Street Lawrenceville, GA 30043
Stevie:Vu:7894561230:7410 Some Drive Chamblee, GA 30341
[vdol0@gsuad.gsu.edu@snowball ~]$ ls -l phone.sh
-rw-------. 1 vdol0@gsuad.gsu.edu vdol0@gsuad.gsu.edu 5228 Oct 10 00:09 phone.sh
[vdol0@gsuad.gsu.edu@snowball ~]$
```

^ using cat to see if the contents in the phonebook has saved and using ls -l to check the read and write permissions for phone.sh

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

A. What is the use of a shell?

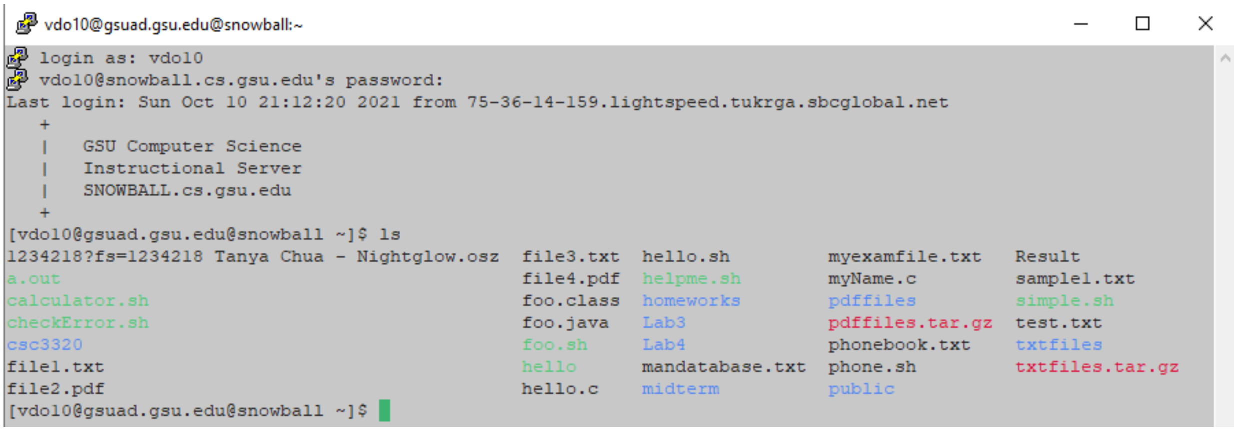
The shell is used to interpret commands and act as a medium between the user and the operating system. It takes the command from a program and interacts with the system to execute all the commands in that program.

- Example: PowerShell, sh, csh

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.

Yes, the shell I see on my PC is different from the shell I see on the snowball server upon login. The snowball server stops all the commands running in the system between data sources, whereas the shell in my PC won't stop these commands. The Snowball Shell Source (SSH) is used to secure connections.

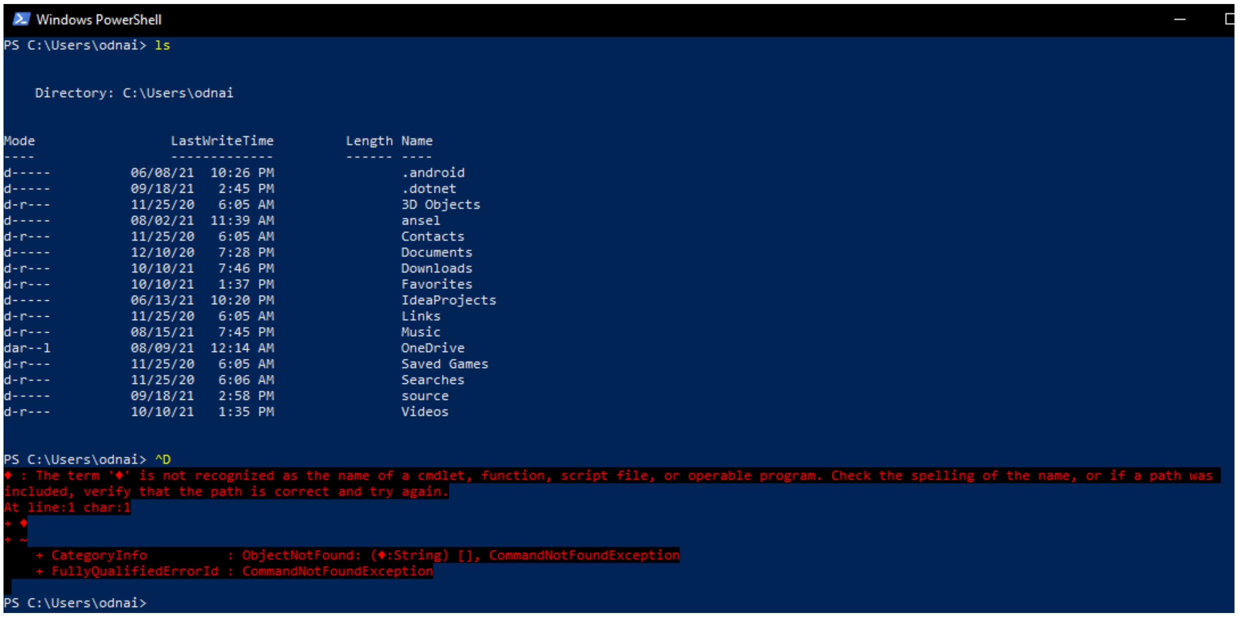
- When trying to exit the shell with ctrl D, the shell on the Snowball Server automatically closes since it is a recognizable command.



```
login as: vdo10
vdo10@snowball.cs.gsu.edu's password:
Last login: Sun Oct 10 21:12:20 2021 from 75-36-14-159.lightspeed.tukrga.sbcglobal.net
   +
   |   GSU Computer Science
   |   Instructional Server
   |   SNOWBALL.cs.gsu.edu
   +
[vdo10@gsuad.gsu.edu@snowball ~]$ ls
1234218?fs=1234218 Tanya Chua - Nightglow.osz  file3.txt  hello.sh         myexamfile.txt   Result
a.out                                          file4.pdf  helpme.sh        myName.c         sample1.txt
calculator.sh                                  foo.class  homeworks        pdffiles         simple.sh
checkError.sh                                  foo.java   Lab3             pdffiles.tar.gz  test.txt
csc3320                                        foo.sh     Lab4             phonebook.txt    txtfiles
file1.txt                                      hello      mandatabase.txt  phone.sh         txtfiles.tar.gz
file2.pdf                                      hello.c    midterm          public
[vdo10@gsuad.gsu.edu@snowball ~]$
```

^ connects to snowball server through my campus ID

- When trying to exit the shell with ctrl D, the shell on my own PC has an error message and won't exit since it is not a recognizable command. I'd have to manually click on the 'x' if I wanted to exit the shell. The home directory layout is also different from putty's.



```
PS C:\Users\odnai> ls


    Directory: C:\Users\odnai


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        06/08/21  10:26 PM                .android
d-----        09/18/21   2:45 PM                .dotnet
d-r---        11/25/20   6:05 AM                3D Objects
d-----        08/02/21  11:39 AM                ansel
d-r---        11/25/20   6:05 AM                Contacts
d-----        12/10/20   7:28 PM                Documents
d-r---        10/10/21   7:46 PM                Downloads
d-r---        10/10/21   1:37 PM                Favorites
d-----        06/13/21  10:20 PM                IdeaProjects
d-r---        11/25/20   6:05 AM                Links
d-r---        08/15/21   7:45 PM                Music
dar--l        08/09/21  12:14 AM                OneDrive
d-r---        11/25/20   6:05 AM                Saved Games
d-r---        11/25/20   6:06 AM                Searches
d-----        09/18/21   2:58 PM                source
d-r---        10/10/21   1:35 PM                Videos


PS C:\Users\odnai> ^D
♦ : The term '♦' is not recognized as the name of a cmdlet, function, script file, or operable program. Check the spelling of the name, or if a path was
included, verify that the path is correct and try again.
At line:1 char:1
+ ♦
+ ~
    + CategoryInfo          : ObjectNotFound: (♦:String) [], CommandNotFoundException
    + FullyQualifiedErrorId : CommandNotFoundException

PS C:\Users\odnai>
```

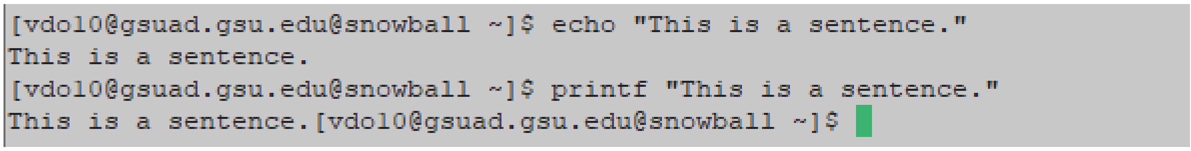
C. What are the elements in a computer (software and hardware) that enable the understanding and interpretation of a C program?

For hardware, the central processing unit (CPU) is able to understand and interpret software instructions in general. This applies to compiling C programs to be interpreted and executed.
For softwares, putty and Ubuntu are two examples of softwares that are able to compile C programs.

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 commands *printf()* and *echo* are two built-in commands. *printf()* usually gives an exit status of non-zero, whereas *echo* always ends with a zero exit status. *printf()* is slower than *echo* when their commands are executed.

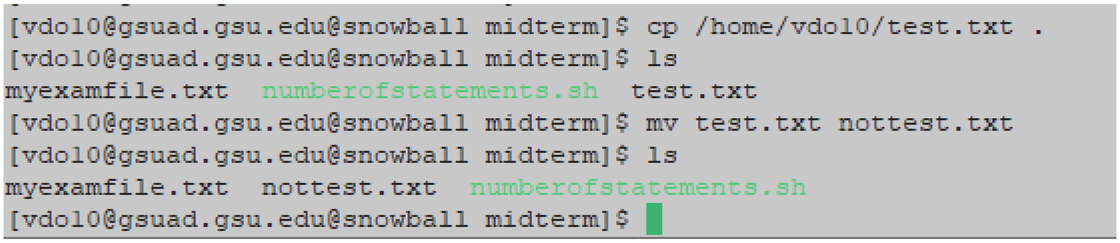
- Example:
    - *echo* has a default newline character
    - *printf* has to manually add a newline character



```
[vdo10@gsuad.gsu.edu@snowball ~]$ echo "This is a sentence."
This is a sentence.
[vdo10@gsuad.gsu.edu@snowball ~]$ printf "This is a sentence."
This is a sentence.[vdo10@gsuad.gsu.edu@snowball ~]$
```

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

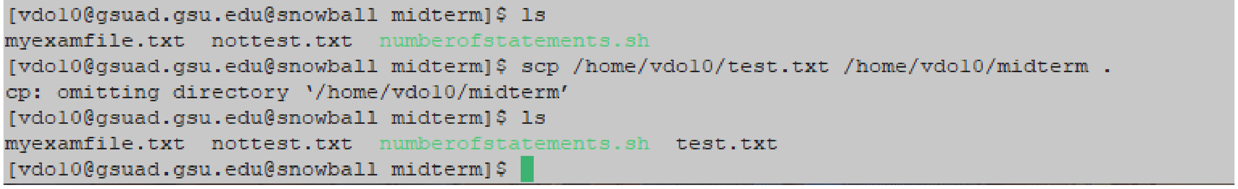
ssh: a connection between two systems used to copy, manage, or move files



```
[vdo10@gsuad.gsu.edu@snowball midterm]$ cp /home/vdo10/test.txt .
[vdo10@gsuad.gsu.edu@snowball midterm]$ ls
myexamfile.txt  numberofstatements.sh  test.txt
[vdo10@gsuad.gsu.edu@snowball midterm]$ mv test.txt nottest.txt
[vdo10@gsuad.gsu.edu@snowball midterm]$ ls
myexamfile.txt  nottest.txt  numberofstatements.sh
[vdo10@gsuad.gsu.edu@snowball midterm]$
```

- I made a test.txt file. I used *cp* to copy the file into a directory I had. I used *mv* to change the name of the test.txt file into nottest.txt. It is now a new file.

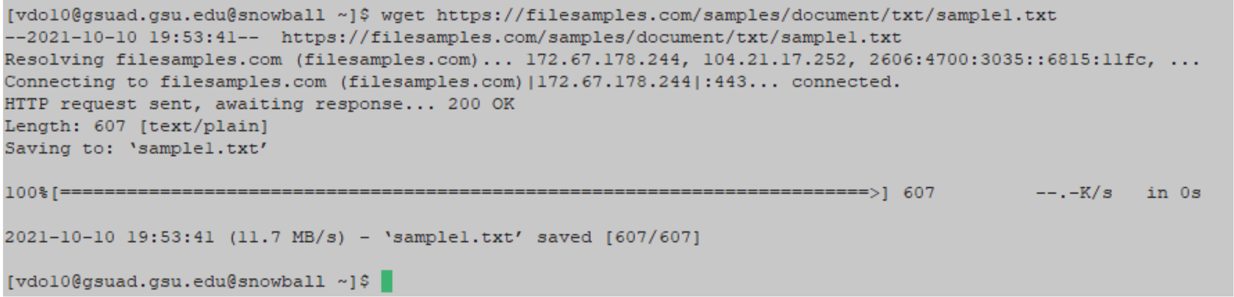
scp: a tool used by the SSH network protocol that copies a file on a remote server to the computer



```
[vdo10@gsuad.gsu.edu@snowball midterm]$ ls
myexamfile.txt  nottest.txt  numberofstatements.sh
[vdo10@gsuad.gsu.edu@snowball midterm]$ scp /home/vdo10/test.txt /home/vdo10/midterm .
cp: omitting directory ‘/home/vdo10/midterm’
[vdo10@gsuad.gsu.edu@snowball midterm]$ ls
myexamfile.txt  nottest.txt  numberofstatements.sh  test.txt
[vdo10@gsuad.gsu.edu@snowball midterm]$
```

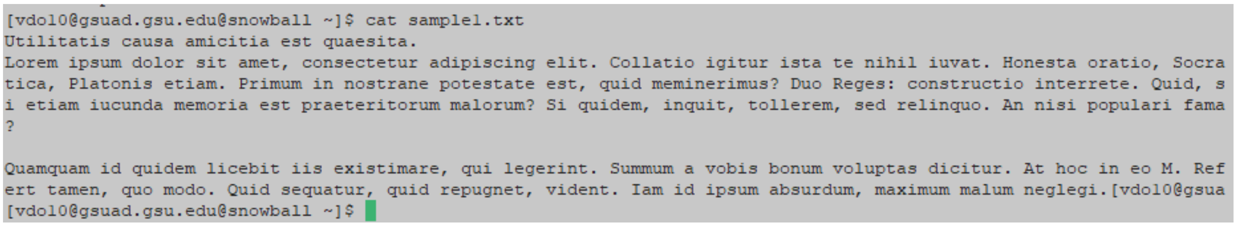
- Using the same file as above, I copied the file from the home directory to the midterm directory using *scp /home/vdo10/test.txt /home/vdo10/midterm .*
- This copies the text.file into the midterm directory.

wget: a non-active network downloader that is used to download files using the pasted URL from the server



```
[vdol0@gsuad.gsu.edu@snowball ~]$ wget https://filesamples.com/samples/document/txt/sample1.txt
--2021-10-10 19:53:41--  https://filesamples.com/samples/document/txt/sample1.txt
Resolving filesamples.com (filesamples.com)... 172.67.178.244, 104.21.17.252, 2606:4700:3035::6815:11fc, ...
Connecting to filesamples.com (filesamples.com)|172.67.178.244|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 607 [text/plain]
Saving to: ‘sample1.txt’

100%[=========================================================================>] 607         --.-K/s   in 0s

2021-10-10 19:53:41 (11.7 MB/s) - ‘sample1.txt’ saved [607/607]

[vdol0@gsuad.gsu.edu@snowball ~]$ 
```

- Using a sample file I found from the internet, I downloaded the sample1.txt file into my main directory using *wget*.



```
[vdol0@gsuad.gsu.edu@snowball ~]$ cat sample1.txt
Utilitatis causa amicitia est quaesita.
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Collatio igitur ista te nihil iuvat. Honesta oratio, Socra
tica, Platonis etiam. Primum in nostrane potestate est, quid meminerimus? Duo Reges: constructio interrete. Quid, s
i etiam iucunda memoria est praeteritorum malorum? Si quidem, inquit, tollerem, sed relinquo. An nisi populari fama
?

Quamquam id quidem licebit iis existimare, qui legerint. Summum a vobis bonum voluptas dicitur. At hoc in eo M. Ref
ert tamen, quo modo. Quid sequatur, quid repugnet, vident. Iam id ipsum absurdum, maximum malum neglegi.[vdol0@gsua
[vdol0@gsuad.gsu.edu@snowball ~]$ 
```

^ the contents inside the sample1.txt file