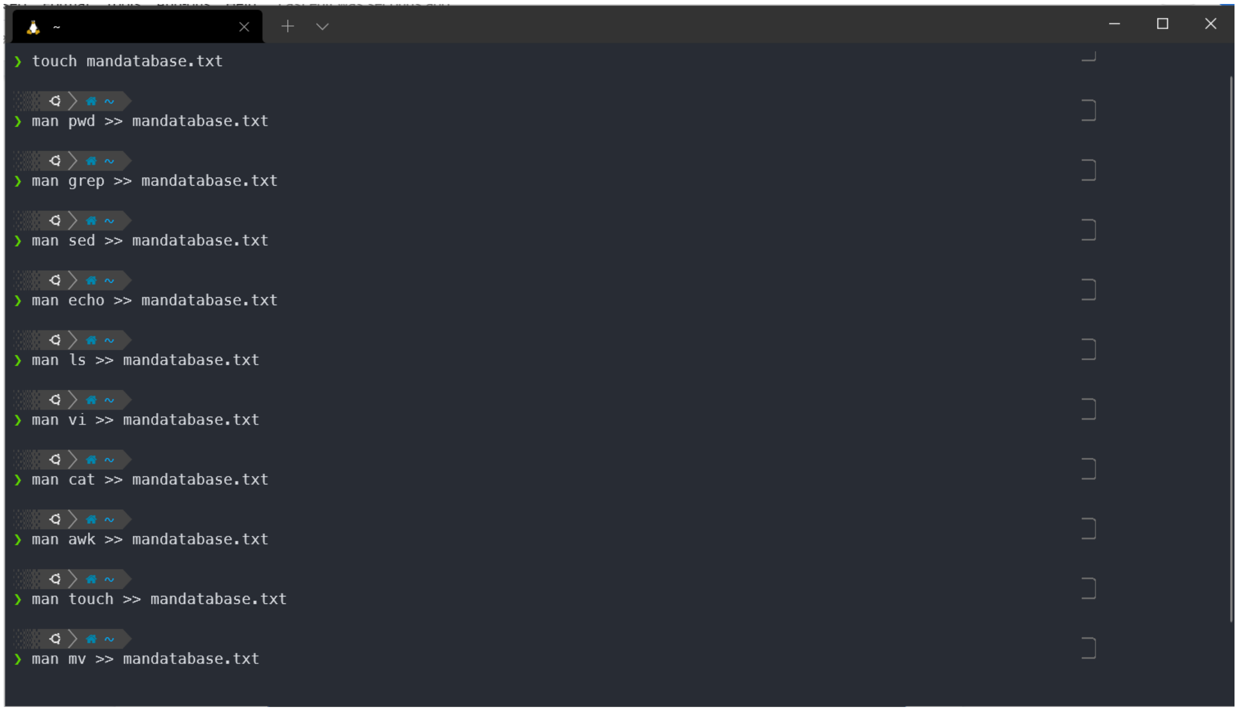
Homework 4 - Midterm Redo
Paul Ofremu Jr.

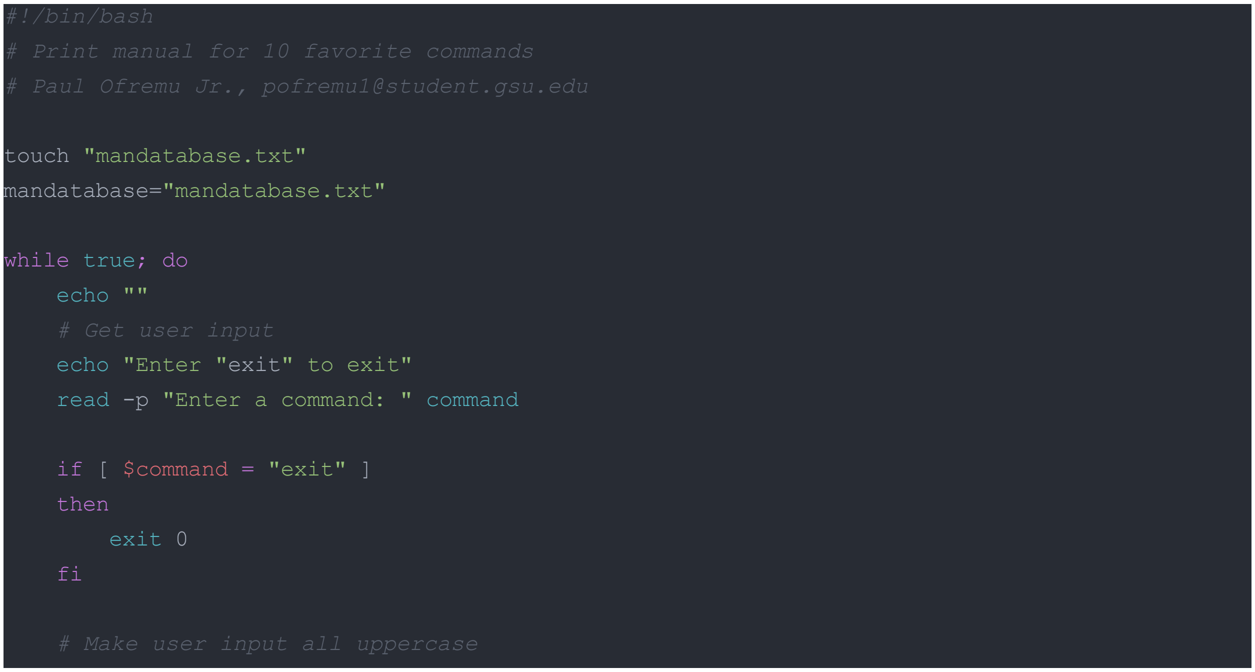
Q1)

Enter favorite commands into mandatabase.txt (File provided):



```
❯ touch mandatabase.txt

❯ man pwd >> mandatabase.txt

❯ man grep >> mandatabase.txt

❯ man sed >> mandatabase.txt

❯ man echo >> mandatabase.txt

❯ man ls >> mandatabase.txt

❯ man vi >> mandatabase.txt

❯ man cat >> mandatabase.txt

❯ man awk >> mandatabase.txt

❯ man touch >> mandatabase.txt

❯ man mv >> mandatabase.txt
```

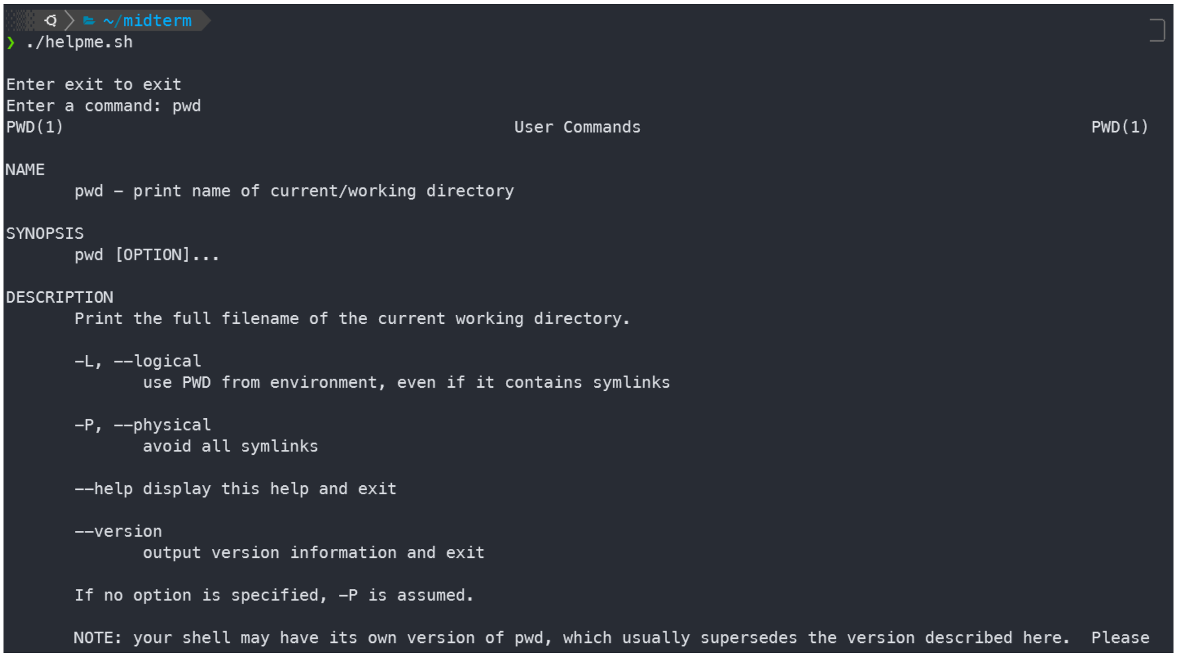
Code:



```
#!/bin/bash
# Print manual for 10 favorite commands
# Paul Ofremu Jr., pofremu1@student.gsu.edu

touch "mandatabase.txt"
mandatabase="mandatabase.txt"

while true; do
    echo ""
    # Get user input
    echo "Enter "exit" to exit"
    read -p "Enter a command: " command

    if [ $command = "exit" ]
    then
        exit 0
    fi

    # Make user input all uppercase
    command=$(echo "$command" | tr [a-z] [A-Z])

    # Check for the commandname(1) at the beginning and end of a line
    if [ ! -z "$(sed -n "/^$command(1)/,/$command(1)$/p" "$mandatabase")" ]
    then
        # Print everything in between first line and last line of command
        sed -n "/^$command(1)/,/$command(1)$/p" "$mandatabase" | more
    else
        # Command is not found
        echo "Sorry , I cannot help you"
    fi
done
```

Instructions: To run **helpme.sh**, user must be granted execution permission. Enter this command: **chmod u+x helpme.sh**. Run file by entering: **./helpme.sh**

Screenshot:



```
~/midterm
❯ ./helpme.sh

Enter exit to exit
Enter a command: pwd
PWD(1)                                                  User Commands                                                 PWD(1)

NAME
       pwd - print name of current/working directory

SYNOPSIS
       pwd [OPTION]...

DESCRIPTION
       Print the full filename of the current working directory.

       -L, --logical
              use PWD from environment, even if it contains symlinks

       -P, --physical
              avoid all symlinks

       --help display this help and exit

       --version
              output version information and exit

       If no option is specified, -P is assumed.

       NOTE: your shell may have its own version of pwd, which usually supersedes the version described here.  Please
```

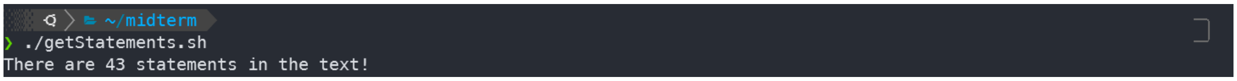
Q2)

a)
Code:

```
#!/bin/bash
# Count # of Statements
# Paul Ofremu Jr., pofremu1@student.gsu.edu

# Input file
myexamfile="myexamfile.txt"

# Get count of statements
count=$(grep -oc ".*\." "$myexamfile")
echo "There are $count statements in the text!"
echo ""
```

Instructions: **myexamfile.txt** contains the text from a wikipedia page.To run **getStatements.sh**, user must be granted execution permission. Enter this command: **chmod u+x getStatements.sh**. Run file by entering: **./getStatements.sh**

Screenshot:



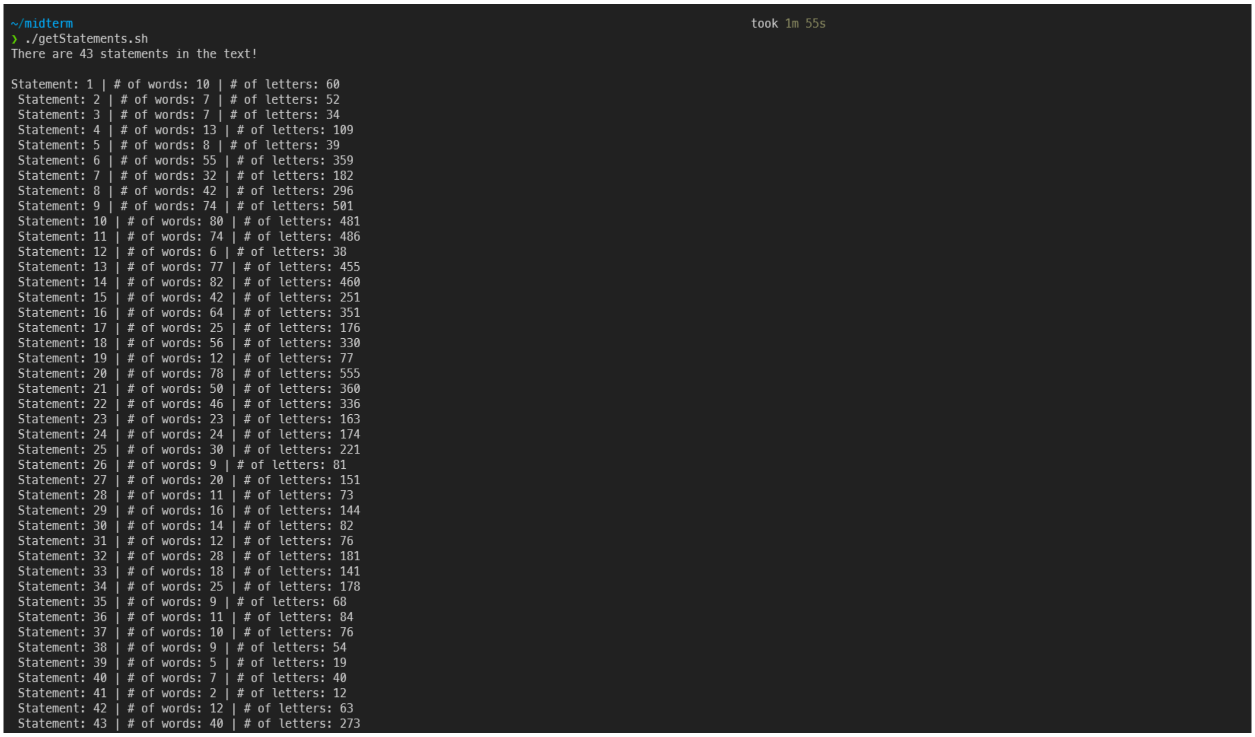
```
~/midterm
❯ ./getStatements.sh
There are 43 statements in the text!
```

b)
Code:

```
#!/bin/bash
# Count # of Statements
# Paul Ofremu Jr., pofremu1@student.gsu.edu

# Input file
myexamfile="myexamfile.txt"

# Get count of statements
count=$(grep -oc ".*\." "$myexamfile")
echo "There are $count statements in the text!"
echo ""

statements=$(grep -o ".*\." "$myexamfile")

# Get number of words and letters for each statements
echo $(grep -o ".*\." "$myexamfile" | awk '{print "Statement: " NR " | # of words: " NF " | # of letters: " length " \\n"}')
```

Instructions: **myexamfile.txt** contains the text from a wikipedia page.To run **getStatements.sh**, user must be granted execution permission. Enter this command: **chmod u+x getStatements.sh**. Run file by entering: **./getStatements.sh**

Screenshot:



```
~/midterm                                                                                                                          took 1m 55s
❯ ./getStatements.sh
There are 43 statements in the text!

Statement: 1 | # of words: 10 | # of letters: 60
 Statement: 2 | # of words: 7 | # of letters: 52
 Statement: 3 | # of words: 7 | # of letters: 34
 Statement: 4 | # of words: 13 | # of letters: 109
 Statement: 5 | # of words: 8 | # of letters: 39
 Statement: 6 | # of words: 55 | # of letters: 359
 Statement: 7 | # of words: 32 | # of letters: 182
 Statement: 8 | # of words: 42 | # of letters: 296
 Statement: 9 | # of words: 74 | # of letters: 501
 Statement: 10 | # of words: 80 | # of letters: 481
 Statement: 11 | # of words: 74 | # of letters: 486
 Statement: 12 | # of words: 6 | # of letters: 38
 Statement: 13 | # of words: 77 | # of letters: 455
 Statement: 14 | # of words: 82 | # of letters: 460
 Statement: 15 | # of words: 42 | # of letters: 251
 Statement: 16 | # of words: 64 | # of letters: 351
 Statement: 17 | # of words: 25 | # of letters: 176
 Statement: 18 | # of words: 56 | # of letters: 330
 Statement: 19 | # of words: 12 | # of letters: 77
 Statement: 20 | # of words: 78 | # of letters: 555
 Statement: 21 | # of words: 50 | # of letters: 360
 Statement: 22 | # of words: 46 | # of letters: 336
 Statement: 23 | # of words: 23 | # of letters: 163
 Statement: 24 | # of words: 24 | # of letters: 174
 Statement: 25 | # of words: 30 | # of letters: 221
 Statement: 26 | # of words: 9 | # of letters: 81
 Statement: 27 | # of words: 20 | # of letters: 151
 Statement: 28 | # of words: 11 | # of letters: 73
 Statement: 29 | # of words: 16 | # of letters: 144
 Statement: 30 | # of words: 14 | # of letters: 82
 Statement: 31 | # of words: 12 | # of letters: 76
 Statement: 32 | # of words: 28 | # of letters: 181
 Statement: 33 | # of words: 18 | # of letters: 141
 Statement: 34 | # of words: 25 | # of letters: 178
 Statement: 35 | # of words: 9 | # of letters: 68
 Statement: 36 | # of words: 11 | # of letters: 84
 Statement: 37 | # of words: 10 | # of letters: 76
 Statement: 38 | # of words: 9 | # of letters: 54
 Statement: 39 | # of words: 5 | # of letters: 19
 Statement: 40 | # of words: 7 | # of letters: 40
 Statement: 41 | # of words: 2 | # of letters: 12
 Statement: 42 | # of words: 12 | # of letters: 63
 Statement: 43 | # of words: 40 | # of letters: 273
```

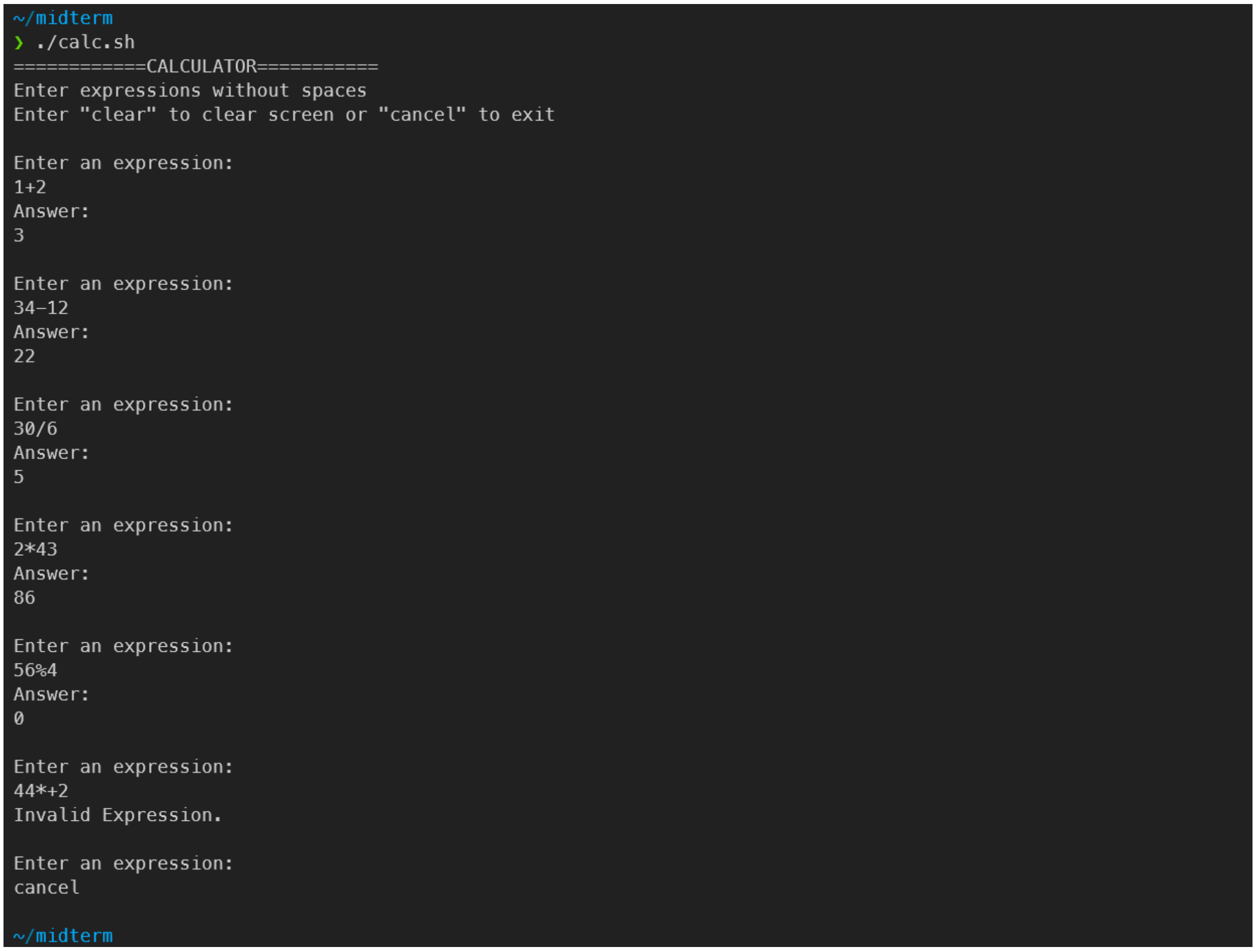
Q3)

Code:

```
#!/bin/bash
# Calculator
# Paul Ofremu Jr., pofremu1@student.gsu.edu

menu ()
{
echo "============CALCULATOR==========="
echo "Enter expressions without spaces"
echo "Enter \"clear\" to clear screen or \"cancel\" to exit"
echo ""
}

menu
while true; do
    # Get user input
    echo "Enter an expression:"
    read expr

    #
    case "$expr" in
        "cancel")
            exit
            ;;

        "clear")
            clear
            menu
            ;;

        *)
            # If input matches regex for valid expression evaluate with bc
            [[ $expr =~ ([0-9]+)([/%*+-])([0-9]+) ]] && echo "Answer: " && echo "$expr" | bc
|| echo "Invalid Expression."
            echo ""
            ;;
    esac
done
```

Instructions: Enter this command: **chmod u+x calc.sh** to provide user execution permission. Run file by entering: **./calc.sh**. Enter a valid expression and press “enter”.

Screenshot:



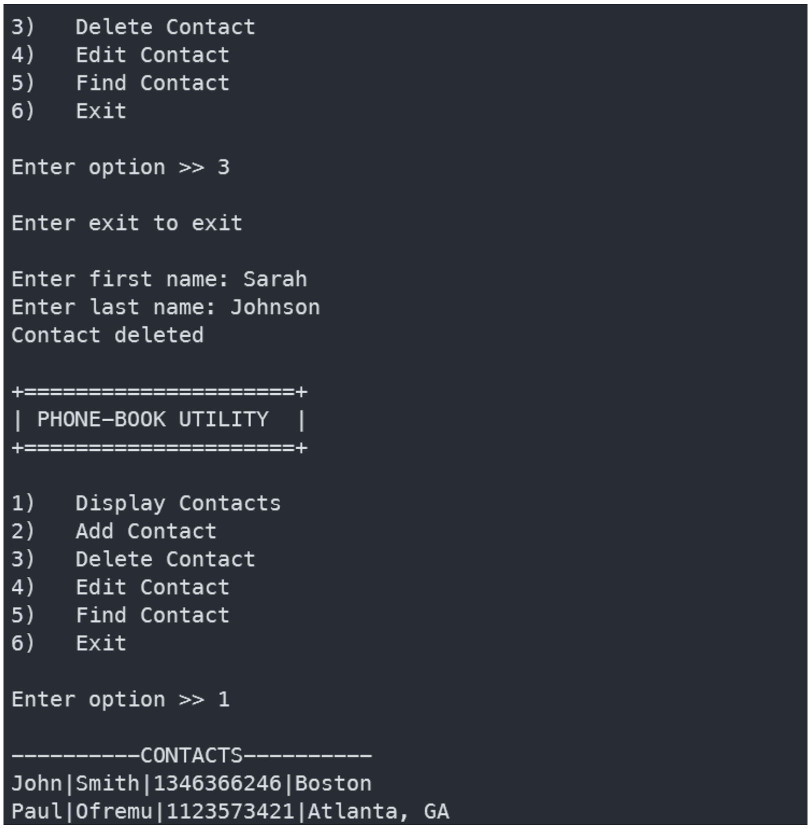
```
~/midterm
❯ ./calc.sh
============CALCULATOR===========
Enter expressions without spaces
Enter "clear" to clear screen or "cancel" to exit

Enter an expression:
1+2
Answer:
3

Enter an expression:
34-12
Answer:
22

Enter an expression:
30/6
Answer:
5

Enter an expression:
2*43
Answer:
86

Enter an expression:
56%4
Answer:
0

Enter an expression:
44*+2
Invalid Expression.

Enter an expression:
cancel

~/midterm
```

Q4)

Code: File provided (phone.sh)

Instructions: The user must be provided read, write and permission. Enter this command: **chmod 700 phone.sh**. Run file by entering: **./phone.sh**.

Screenshot:

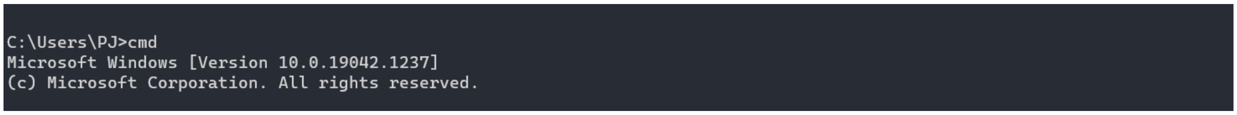


```
3)   Delete Contact
4)   Edit Contact
5)   Find Contact
6)   Exit

Enter option >> 3

Enter exit to exit

Enter first name: Sarah
Enter last name: Johnson
Contact deleted

+=====================+
| PHONE-BOOK UTILITY  |
+=====================+

1)   Display Contacts
2)   Add Contact
3)   Delete Contact
4)   Edit Contact
5)   Find Contact
6)   Exit

Enter option >> 1

----------CONTACTS----------
John|Smith|1346366246|Boston
Paul|Ofremu|1123573421|Atlanta, GA
```

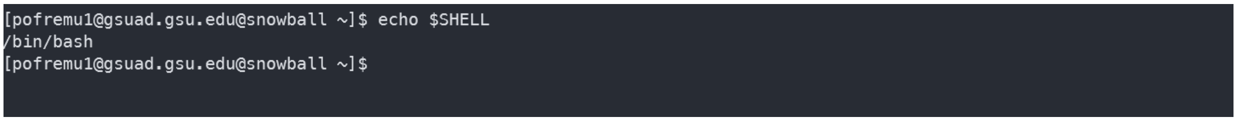
Q5)

A) A shell is an interface between a user and the operating system. A shell gives the user the ability to take advantage of the basic operations of the operating system such as file management, process management and batch processing.

B) The shell on my PC is a command-line shell while the shell on the snowball server is a bash shell.



```
C:\Users\PJ>cmd
Microsoft Windows [Version 10.0.19042.1237]
(c) Microsoft Corporation. All rights reserved.
```

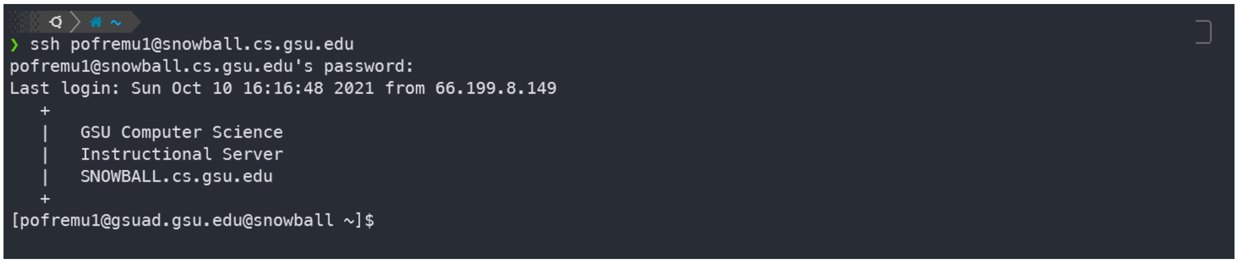


```
[pofremu1@gsuad.gsu.edu@snowball ~]$ echo $SHELL
/bin/bash
[pofremu1@gsuad.gsu.edu@snowball ~]$
```

C) C is a compiled language so the source code is handed to a compiler which converts the source code into object code, which is machine language. This is then passed to linker to add any additional code needed for the code to execute. The output is code that the computer can understand and be executed by the CPU

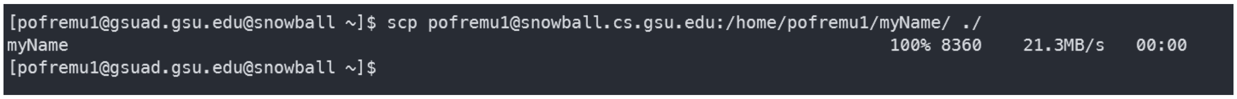
D) The "echo" command takes one argument and prints that argument to the screen followed by a new line. The "printf()" command in C takes multiple arguments, the first one is the string format to print to screen, and the remaining arguments are the arguments to replace the specifiers.

E) The "ssh" command allows you to securely connect to a remote computer or server and gain access to its files, the terminal and other applications. Ex: Connecting to the GSU Snowball server through ssh:



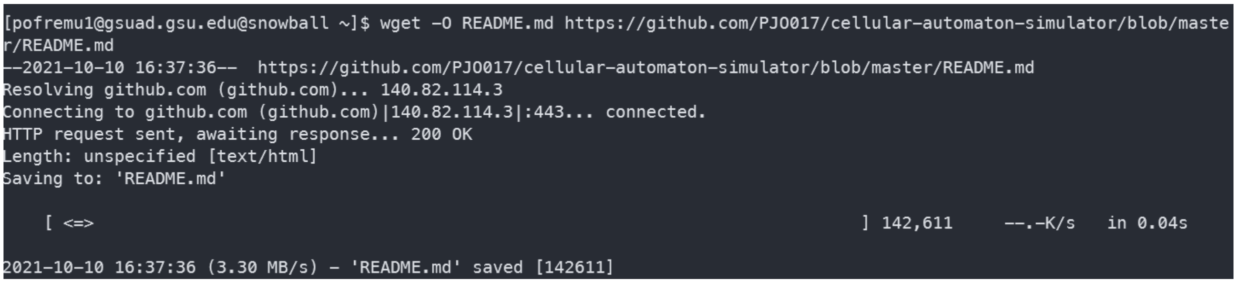
```
❯ ssh pofremu1@snowball.cs.gsu.edu
pofremu1@snowball.cs.gsu.edu's password:
Last login: Sun Oct 10 16:16:48 2021 from 66.199.8.149
    +
    |   GSU Computer Science
    |   Instructional Server
    |   SNOWBALL.cs.gsu.edu
    +
[pofremu1@gsuad.gsu.edu@snowball ~]$
```

The "scp" command allows you to copy files and directories between servers or computers. (Local host to remote or remote to another remote computer). Ex: Copying myName directory to local machine:



```
[pofremu1@gsuad.gsu.edu@snowball ~]$ scp pofremu1@snowball.cs.gsu.edu:/home/pofremu1/myName/ ./
myName                                                                          100% 8360    21.3MB/s   00:00
[pofremu1@gsuad.gsu.edu@snowball ~]$
```

The “wget” command is used to download files from the web. You can download files using the HTTP, HTTPS, and FTP protocols. Ex: Download Readme.md file from my github repo:



```
[pofremu1@gsuad.gsu.edu@snowball ~]$ wget -O README.md https://github.com/PJO017/cellular-automaton-simulator/blob/maste
r/README.md
--2021-10-10 16:37:36--  https://github.com/PJO017/cellular-automaton-simulator/blob/master/README.md
Resolving github.com (github.com)... 140.82.114.3
Connecting to github.com (github.com)|140.82.114.3|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/html]
Saving to: 'README.md'

    [ <=>                                                                            ] 142,611     --.-K/s   in 0.04s

2021-10-10 16:37:36 (3.30 MB/s) - 'README.md' saved [142611]
```