CSc 3320: Systems Programming

Fall 2021

EXAM 1 (Midterm)

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: Aleeza Iftikhar

Campus ID: aiftikhar2

Panther #: 002473603

Questions 1-5 are 20pts each

1. (20 pts) Pick any of your 10 favorite 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

Creating mandatabase.txt file

```
[aiftikhar2@gsuad.gsu.edu@snowball midterm_redo]$ ls
q1  q2  q3  q4
[aiftikhar2@gsuad.gsu.edu@snowball midterm_redo]$ cd q1
[aiftikhar2@gsuad.gsu.edu@snowball q1]$ touch mandatabase.txt
[aiftikhar2@gsuad.gsu.edu@snowball q1]$ man ls >> mandatabase.txt
[aiftikhar2@gsuad.gsu.edu@snowball q1]$ man mv >> mandatabase.txt
[aiftikhar2@gsuad.gsu.edu@snowball q1]$ man sed >> mandatabase.txt
[aiftikhar2@gsuad.gsu.edu@snowball q1]$ man pwd >> mandatabase.txt
[aiftikhar2@gsuad.gsu.edu@snowball q1]$ man rm >> mandatabase.txt
[aiftikhar2@gsuad.gsu.edu@snowball q1]$ man cp >> mandatabase.txt
[aiftikhar2@gsuad.gsu.edu@snowball q1]$ man mkdir >> mandatabase.txt
[aiftikhar2@gsuad.gsu.edu@snowball q1]$ man rmdir >> mandatabase.txt
[aiftikhar2@gsuad.gsu.edu@snowball q1]$ man touch >> mandatabase.txt
[aiftikhar2@gsuad.gsu.edu@snowball q1]$ man  tr >> mandatabase.txt
[aiftikhar2@gsuad.gsu.edu@snowball q1]$
```

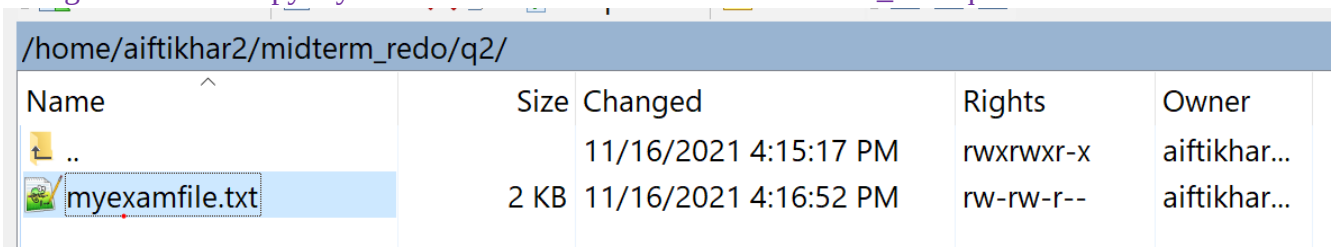
Running helpme.sh

```
[aiftikhar2@gsuad.gsu.edu@snowball q1]$ ls
helpme.sh  mandatabase.txt
[aiftikhar2@gsuad.gsu.edu@snowball q1]$ ./helpme.sh
Enter a command: ls
LS(1)                            User Commands                           LS(1)



NAME
       ls - list directory contents

SYNOPSIS
       ls [OPTION]... [FILE]...

DESCRIPTION
       List  information  about  the FILEs (the current directory by default).
       Sort entries alphabetically if none of -cftuvSUX nor --sort  is  speci□
       fied.
```

2. (10pts each) On your computer open your favorite 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.

Using WinSCP to copy myexamfile.txt to /home/aiftikhar2/midterm_redo/q2:



Running script.sh:

```
[aiftikhar2@gsuad.gsu.edu@snowball ~]$ cd midterm_redo
[aiftikhar2@gsuad.gsu.edu@snowball midterm_redo]$ cd q2
[aiftikhar2@gsuad.gsu.edu@snowball q2]$ ls
myexamfile.txt  script.sh
[aiftikhar2@gsuad.gsu.edu@snowball q2]$ ./script.sh
Total number of statments found: 9
Statment_no      no_of_words     no_of_letters
1                3               23
2                6               31
3                29              181
4                13              73
5                34              203
6                38              233
7                26              162
8                27              179
9                34              231
[aiftikhar2@gsuad.gsu.edu@snowball q2]$
```

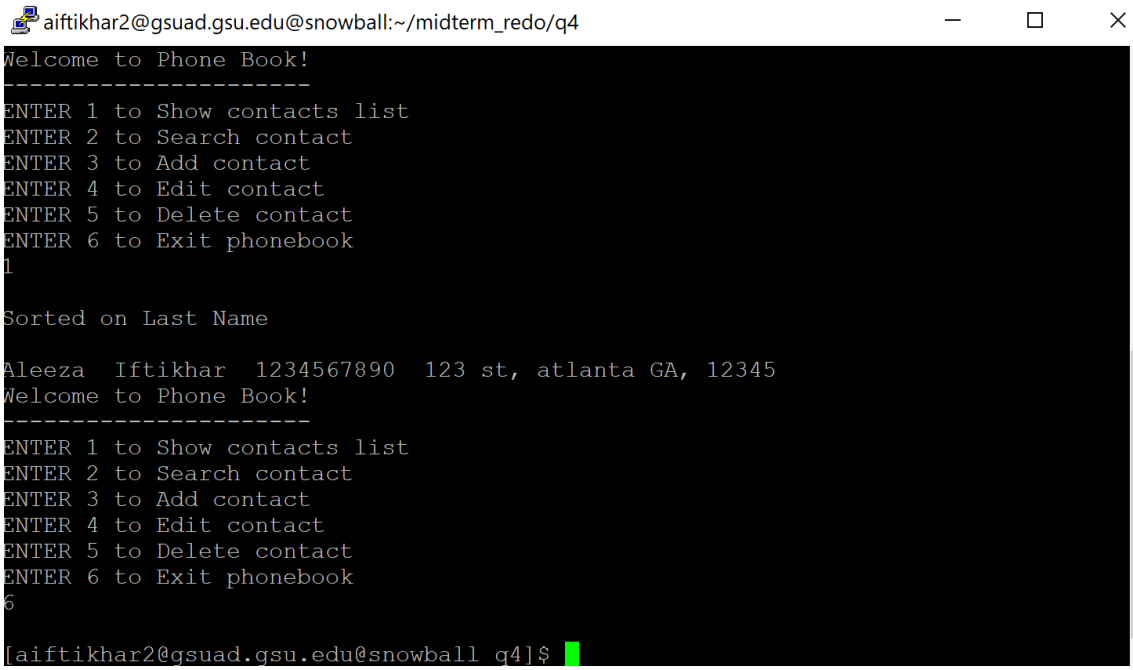
3. (20pts) Design a calculator using a shell script using regular expressions. The calculator, at the minimum, must be able to process addition, subtraction, multiplication, division and modulo operations. It must also have cancelled and clear features.

Running calculator.sh

```
[aiftikhar2@gsuad.gsu.edu@snowball midterm_redo]$ cd q3
[aiftikhar2@gsuad.gsu.edu@snowball q3]$ ls
calculator.sh
[aiftikhar2@gsuad.gsu.edu@snowball q3]$ ./calculator.sh

Enter num OR operator(+,-,*,/,%) OR 'C' for Cancel OR 'CE' for clear:
2


Enter operator(+,-,*,/,%) OR 'C' for Cancel OR 'CE' for clear:
+


Enter num OR 'C' for Cancel OR 'CE' for clear:
3

num1 = 2 ; num2 = 3 ; operator = +
result = 5

Enter num OR operator(+,-,*,/,%) OR 'C' for Cancel OR 'CE' for clear:
C

-------Cancelling the calculator...bye--------
[aiftikhar2@gsuad.gsu.edu@snowball q3]$
```

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).

```
[aiftikhar2@gsuad.gsu.edu@snowball ~]$ cd midterm_redo/
[aiftikhar2@gsuad.gsu.edu@snowball midterm_redo]$ cd q4
[aiftikhar2@gsuad.gsu.edu@snowball q4]$ ls
phonebook.sh
[aiftikhar2@gsuad.gsu.edu@snowball q4]$ ./phonebook.sh
Welcome to Phone Book!
----------------------
ENTER 1 to Show contacts list
ENTER 2 to Search contact
ENTER 3 to Add contact
ENTER 4 to Edit contact
ENTER 5 to Delete contact
ENTER 6 to Exit phonebook
```



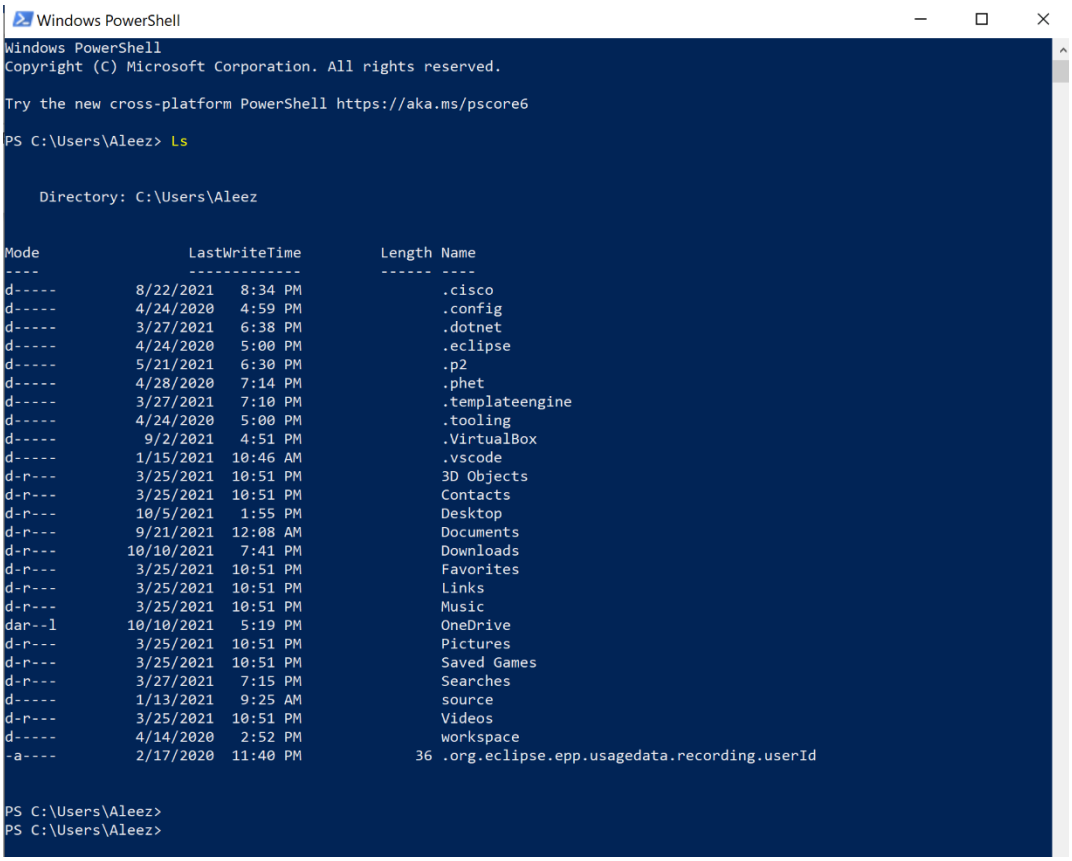
5. (4 pts each) Give brief answers with examples, wherever relevant:

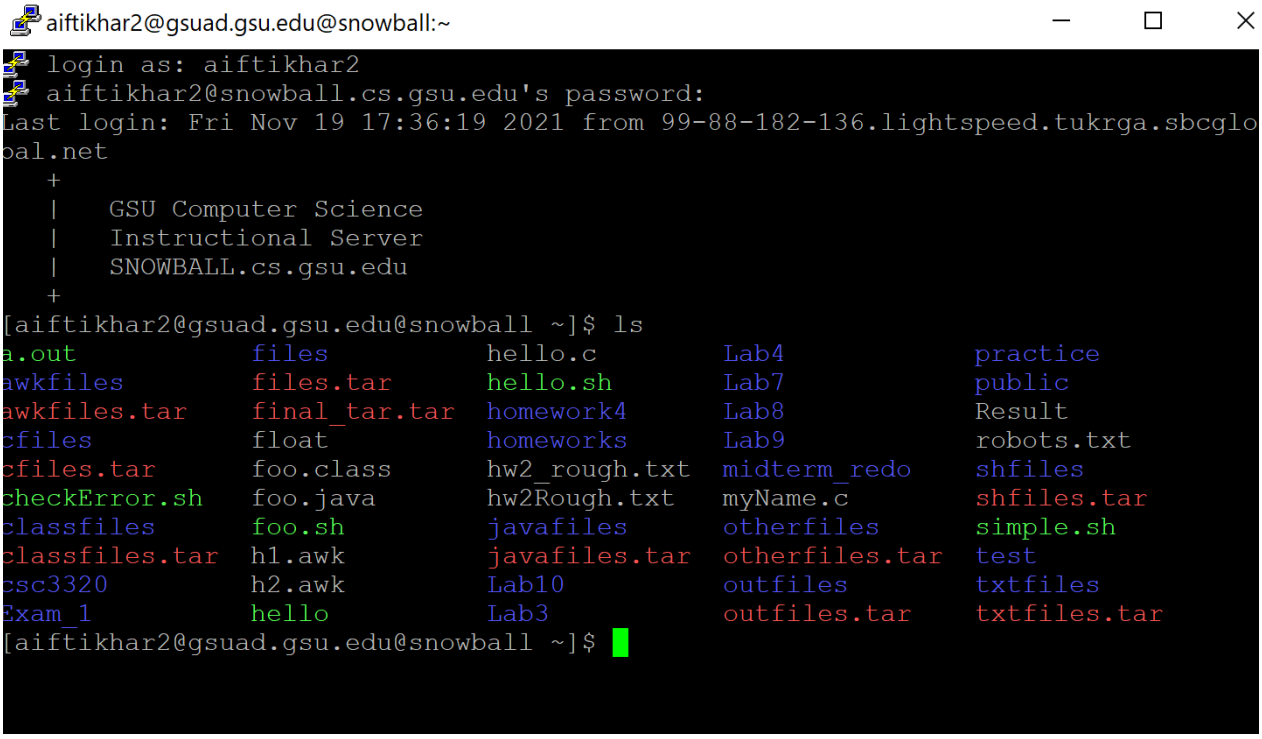
A. What is the use of a shell?

Shell is command line virtual interface between user and operating system. It reads commands from users, interprets them, and runs/executes programs accordingly. In addition, it also performs background processing.

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.

- Operating System: On my PC, I have power shell which is shell for Windows operating system. Whereas on snowball server, I see bash shell which is shell for Linux operating system.
- User Interface: Bash shell has text based CLI whereas PowerShell has graphical command line interface. (Point-and-click)
- Commands: ls in bash works same as Ls in PowerShell


Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\Users\Aleez> Ls
Directory: C:\Users\Aleez
Mode LastWriteTime Length Name
d----- 8/22/2021 8:34 PM .cisco
d----- 4/24/2020 4:59 PM .config
d----- 3/27/2021 6:38 PM .dotnet
d----- 4/24/2020 5:00 PM .eclipse
d----- 5/21/2021 6:30 PM .p2
d----- 4/28/2020 7:14 PM .phet
d----- 3/27/2021 7:10 PM .templateengine
d----- 4/24/2020 5:00 PM .tooling
d----- 9/2/2021 4:51 PM .VirtualBox
d----- 1/15/2021 10:46 AM .vscode
d-r--- 3/25/2021 10:51 PM 3D Objects
d-r--- 3/25/2021 10:51 PM Contacts
d-r--- 10/5/2021 1:55 PM Desktop
d-r--- 9/21/2021 12:08 AM Documents
d-r--- 10/10/2021 7:41 PM Downloads
d-r--- 3/25/2021 10:51 PM Favorites
d-r--- 3/25/2021 10:51 PM Links
d-r--- 3/25/2021 10:51 PM Music
dar--l 10/10/2021 5:19 PM OneDrive
d-r--- 3/25/2021 10:51 PM Pictures
d-r--- 3/25/2021 10:51 PM Saved Games
d-r--- 3/27/2021 7:15 PM Searches
d----- 1/13/2021 9:25 AM source
d-r--- 3/25/2021 10:51 PM Videos
d----- 4/14/2020 2:52 PM workspace
-a---- 2/17/2020 11:40 PM 36 .org.eclipse.epp.usagedata.recording.userId
PS C:\Users\Aleez>
PS C:\Users\Aleez>


aiftikhar2@gsuad.gsu.edu@snowball:~
login as: aiftikhar2
aiftikhar2@snowball.cs.gsu.edu's password:
Last login: Fri Nov 19 17:36:19 2021 from 99-88-182-136.lightspeed.tukrga.sbcglobal.net
GSU Computer Science
Instructional Server
SNOWBALL.cs.gsu.edu
[aiftikhar2@gsuad.gsu.edu@snowball ~]$ ls
a.out files hello.c Lab4 practice
awkfiles files.tar hello.sh Lab7 public
awkfiles.tar final_tar.tar homework4 Lab8 Result
cfiles float homeworks Lab9 robots.txt
cfiles.tar foo.class hw2_rough.txt midterm_redo shfiles
checkError.sh foo.java hw2Rough.txt myName.c shfiles.tar
classfiles foo.sh javafiles otherfiles simple.sh
classfiles.tar h1.awk javafiles.tar otherfiles.tar test
csc3320 h2.awk Lab10 outfiles txtfiles
Exam_1 hello Lab3 outfiles.tar txtfiles.tar
[aiftikhar2@gsuad.gsu.edu@snowball ~]$

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

- Preprocessor (directives commands), Compiler (translator of code), Linker (combines code with other instructions to create an executable programs), are main components in interpreting C program. In addition, other components include operating system, IO routines, assemblers, and interpreters, libraries.

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?

- printf command does not change line at the end of output, whereas echo displays a new line character at end by default. Echo always exits with 0 status whereas printf gives error upon failure to execute, hence it has more control over output.

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: allows the system to form an encrypted secure connection with the host machine/server.

```
[aiftikhar2@gsuad.gsu.edu@snowball ~]$ ssh
usage: ssh [-1246AaCfGgKkMNnqsTtVvXxYy] [-b bind_address] [-c cipher_spec]
           [-D [bind_address:]port] [-E log_file] [-e escape_char]
           [-F configfile] [-I pkcs11] [-i identity_file]
           [-J [user@]host[:port]] [-L address] [-l login_name] [-m mac_spec]
           [-O ctl_cmd] [-o option] [-p port] [-Q query_option] [-R address]
           [-S ctl_path] [-W host:port] [-w local_tun[:remote_tun]]
           [user@]hostname [command]
[aiftikhar2@gsuad.gsu.edu@snowball ~]$ ssh aiftikhar2@snowball.cs.gsu.edu
Last login: Fri Nov 19 16:38:57 2021 from 99-88-182-136.lightspeed.tukrga.sbcglo
bal.net
   +
   |   GSU Computer Science
   |   Instructional Server
   |   SNOWBALL.cs.gsu.edu
   +
[aiftikhar2@gsuad.gsu.edu@snowball ~]$
```

- Scp: transfers files between local and remote server or b/w two remote servers.

```
[aiftikhar2@gsuad.gsu.edu@snowball ~]$ scp
usage: scp [-12346BCpqrv] [-c cipher] [-F ssh_config] [-i identity_file]
           [-l limit] [-o ssh_option] [-P port] [-S program]
           [[user@]host1:]file1 ... [[user@]host2:]file2
[aiftikhar2@gsuad.gsu.edu@snowball ~]$ scp bbello1@snowball.cs.gsu.edu:\Public\R
ealEstate.csv aiftikhar2@snowball.cs.gsu.edu:\test
bbello1@snowball.cs.gsu.edu's password:
```

- Wget: retrieves content from web servers.

```
[aiftikhar2@gsuad.gsu.edu@snowball ~]$ wget https://www.google.com/robots.txt
--2021-11-19 17:51:43--  https://www.google.com/robots.txt
Resolving www.google.com (www.google.com)... 172.217.10.164, 2607:f8b0:4002:c06:
:6a, 2607:f8b0:4002:c06::69, ...
Connecting to www.google.com (www.google.com)|172.217.10.164|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 7165 (7.0K) [text/plain]
Saving to: 'robots.txt'

100%[======================================>] 7,165       --.-K/s   in 0s

2021-11-19 17:51:43 (53.7 MB/s) - 'robots.txt' saved [7165/7165]

[aiftikhar2@gsuad.gsu.edu@snowball ~]$
```