

## 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: Nafisa Nawal

Campus ID: nnawal2

Panther #: 002398884

### 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*

**Answer:**

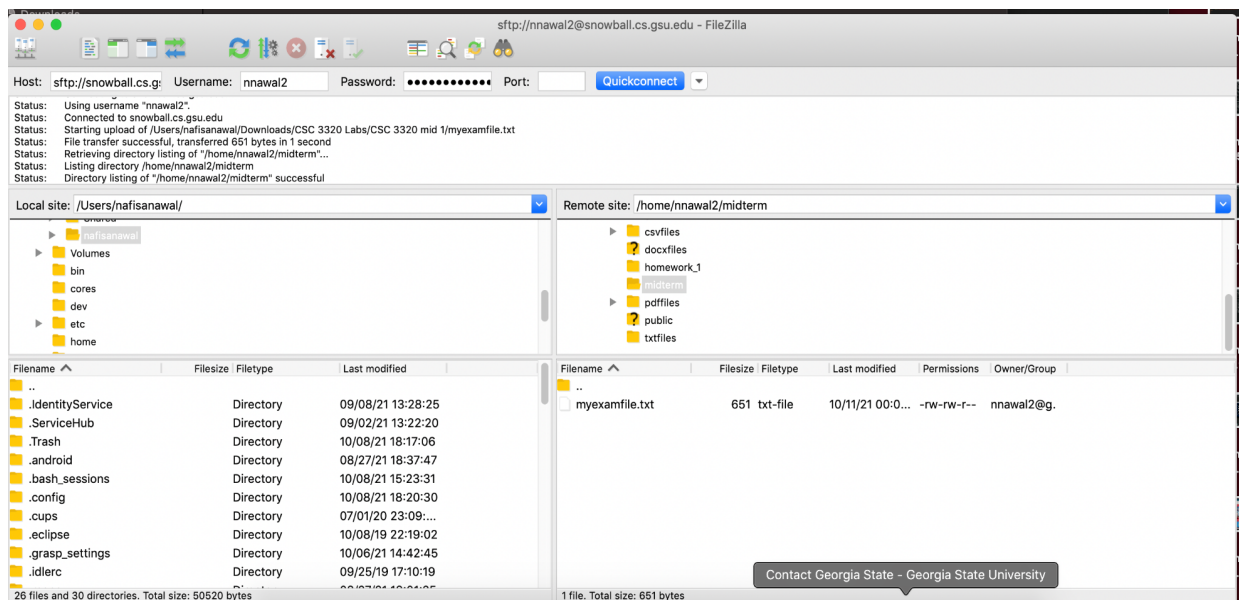
```
nnawal2@gsuad.gsu.edu@snowball:~ — ssh nnawal2@snowball.cs.gsu.edu —...
[[nnawal2@gsuad.gsu.edu@snowball ~]$ touch mandatabase.txt
[[nnawal2@gsuad.gsu.edu@snowball ~]$ man ls >> mandatabase.txt
[[nnawal2@gsuad.gsu.edu@snowball ~]$ man ls-l >> mandatabase.txt
No manual entry for ls-l
[[nnawal2@gsuad.gsu.edu@snowball ~]$ man ls -l >> mandatabase.txt
No manual entry for -l
[[nnawal2@gsuad.gsu.edu@snowball ~]$ man ls -l >> mandatabase.txt
No manual entry for -l
[[nnawal2@gsuad.gsu.edu@snowball ~]$ man chmod >> mandatabase.txt
[[nnawal2@gsuad.gsu.edu@snowball ~]$ man pwd >> mandatabase.txt
[[nnawal2@gsuad.gsu.edu@snowball ~]$ man wc >> mandatabase.txt
[[nnawal2@gsuad.gsu.edu@snowball ~]$ man mkdir >> mandatabase.txt
[[nnawal2@gsuad.gsu.edu@snowball ~]$ man cd >> mandatabase.txt
[[nnawal2@gsuad.gsu.edu@snowball ~]$ man grep >> mandatabase.txt
[[nnawal2@gsuad.gsu.edu@snowball ~]$ man passwd >> mandatabase.txt
[[nnawal2@gsuad.gsu.edu@snowball ~]$ man mv >> mandatabase.txt
[[nnawal2@gsuad.gsu.edu@snowball ~]$ man rm >> mandatabase.txt
[[nnawal2@gsuad.gsu.edu@snowball ~]$
```

```
nnawal2@gsuad.gsu.edu@snowball:~ — ssh nnawal2@snowball.cs.gsu.edu —...  
[nnawal2@gsuad.gsu.edu@snowball ~]$ ./helpme.sh  
Write a command please:  
bhg  
Sorry, I cannot help you  
[nnawal2@gsuad.gsu.edu@snowball ~]$
```

```
nnawal2@gsuad.gsu.edu@snowball:~ — ssh nnawal2@snowball.cs.gsu.edu —...
    POSIXLY_CORRECT also disables _N_GNU_nonoption_argv_flags_,
    This task is achieved through calls to the Linux-PAM and Libuser API.
    passwd through the calls to the pam_cracklib PAM module will try
    GNU coreutils online help: <http://www.gnu.org/software/coreutils/>
    info coreutils 'mv invocation'
GNU coreutils 8.22 November 2020 MV(1)
    GNU coreutils online help: <http://www.gnu.org/software/coreutils/>
    info coreutils 'rm invocation'
GNU coreutils 8.22 November 2020 RM(1)
[[nnawal2@gsuad.gsu.edu@snowball ~]$ ./helpme.sh
Write a command please:
[pwd
    pwd, read, readonly, return, set, shift, shopt, source, suspend, test,
    pwd [-LP]
    pwd - print name of current/working directory
    pwd [OPTION]...
    NOTE: your shell may have its own version of pwd, which usually super-
    Report pwd translation bugs to <http://translationproject.org/team/>
    The full documentation for pwd is maintained as a Texinfo manual. If
    the info and pwd programs are properly installed at your site, the com-
    info coreutils 'pwd invocation'
    pwd, read, readonly, return, set, shift, shopt, source, suspend, test,
    pwd [-LP]
[[nnawal2@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).
- 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.

### Answer:



```
nafisanawal — nnawal2@gsuad.gsu.edu@snowball:~/midterm — ssh nnawal2@snowball.cs.gsu.edu — 70x22
[[nnawal2@gsuad.gsu.edu@snowball ~]$ cd midterm
[[nnawal2@gsuad.gsu.edu@snowball midterm]$ ls
myexamfile.txt  statement_search.sh
[[nnawal2@gsuad.gsu.edu@snowball midterm]$ ./statement_search.sh
6
[[nnawal2@gsuad.gsu.edu@snowball midterm]$
```

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.

### Answer:

```

nafisanawal — nnawal2@gsuad.gsu.edu@snowball:~ — ssh nnawal2@snowball.cs.gsu.edu — 100×36
[[nnawal2@gsuad.gsu.edu@snowball ~]$ vi calculator.sh
[[nnawal2@gsuad.gsu.edu@snowball ~]$
[[nnawal2@gsuad.gsu.edu@snowball ~]$ ./calculator.sh
Welcome to simple calculator
Please enter two numbers each followed by a space:
5 2
Please enter the operation:
division
2.5

Would you like to continue (yes/no)?yes
Please enter two numbers each followed by a space:
5 2
Please enter the operation:
addition
7

Would you like to continue (yes/no)?yes
Please enter two numbers each followed by a space:
5 2
Please enter the operation:
subtraction
3

Would you like to continue (yes/no)?yes
Please enter two numbers each followed by a space:
5 2
Please enter the operation:
multiplication
10

Would you like to continue (yes/no)?yes
Please enter two numbers each followed by a space:
5 2
Please enter the operation:
modulo
1

Would you like to continue (yes/no)?no
[[nnawal2@gsuad.gsu.edu@snowball ~]$
```



```

nafisanawal — nnawal2@gsuad.gsu.edu@snowball:~ — ssh nnawal2@snowball.cs.gsu.edu — 90x32
echo "Welcome to simple calculator"

while [[ "$REPLY" != "no" ]]
do
    echo "Please enter two numbers each followed by a space: "
    read num1 num2
    echo "Please enter the operation: "
    read op
    case $op in
        addition)
            c=$((num1+num2))
            ;;
        subtraction)
            c=$((num1-num2))
            ;;
        multiplication)
            c=$((num1*num2))
            ;;
        division)
            #c=$((num1/num2))
            echo "scale=1; $num1 / $num2" | bc
            ;;
        modulo)
            c=$((num1%num2))
            ;;
        esac
    echo "$c"

    read -p "Would you like to continue (yes/no)?"
done
-- INSERT --
```



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).

**Answer:**

```
nafisanawal — nnawal2@gsuad.gsu.edu@snowball:~ — ssh nnawal2@snowball.cs.gsu.edu — 100x49
[nnawal2@gsuad.gsu.edu@snowball ~]$ ls -l phonebook_utility.sh
-r--w---x. 1 nnawal2@gsuad.gsu.edu nnawal2@gsuad.gsu.edu 754 Oct 11 11:41 phonebook_utility.sh
[nnawal2@gsuad.gsu.edu@snowball ~]$ ./phonebook_utility.sh
-bash: ./phonebook_utility.sh: Permission denied
[nnawal2@gsuad.gsu.edu@snowball ~]$ chmod 700 phonebook_utility.sh
[nnawal2@gsuad.gsu.edu@snowball ~]$ ls -l phonebook_utility.sh
-rwx-----. 1 nnawal2@gsuad.gsu.edu nnawal2@gsuad.gsu.edu 754 Oct 11 11:41 phonebook_utility.sh
[nnawal2@gsuad.gsu.edu@snowball ~]$ ./phonebook_utility.sh
add, find, delete, exit:
[add
Please enter Name, Address, & Phone Number to make an entry in the Phonebook
Please enter name of the person to make an entry: Nafisa Nawal
Address: 3354 Indian Rd
Phone Number: 4036483893
add, find, delete, exit:
[add
Please enter Name, Address, & Phone Number to make an entry in the Phonebook
Please enter name of the person to make an entry: Mosla Uddin
Address: 6677 Bridge Dr
Phone Number: 40664773
add, find, delete, exit:
[find
Name of the person for information to be found: Mosla
Name; Address; Phone number
Mosla Uddin ; 6677 Bridge Dr ; 40664773
add, find, delete, exit:
[find
Name of the person for information to be found: Nafisa
Name; Address; Phone number
Nafisa Koilja ; Heart ; 143
Nafisa Heart ; Heart ; 143
Nafisa Nawal ; 3354 Indian Rd ; 4036483893
```

5. (4 pts each) Give brief answers with examples, wherever relevant
- A. What is the use of a shell?
  - 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.
  - C. What are the elements in a computer (software and hardware) that enable the understanding and interpretation of a C program?
  - 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?
  - E. What do these shell commands do? “ssh”, “scp” and “wget”. Describe briefly using an example that you have executed using the snowball server.

**Answer:**

- A. Shell, also called a command interpreter, is a UNIX term for an operating system's command line interface. It is a program whose primary purpose is to read commands and run other programs. It reads and executes the commands

that a user inputs. Some examples of shells are MS-DOS Shell (command.com), csh, ksh, PowerShell, sh, and tcsh.

B. Yes, the snowball server stops all the commands running the system between the data source whereas the shell in PC's won't stop. And to secure the Snowball secure shell is used ssh that is used for connections,

C. Some basic elements of programming are input, output, looping and conditionals, mathematical operations and variables and data structure. It is interpreted by a compiler of the C language. The machine code performs the output.

D. These two are built in commands in which echo always ends with the status zero, whereas printf gives an exit if non-zero. The printf is slower compared to echo. Example: echo has a default new line character, but in printf we must add manually.

E. The ssh command provides a secure encrypted connection between two computers to communicate and share data. This connection can also be used for terminal access, file transfers, and for tunneling other applications.

The Secure copy protocol (SCP) securely transfers and copies files or directories between a local and a remote system or between two remote systems. The scp command uses ssh

for data transfer.

The `wget` command is used for downloading files from the Internet. Wget is the non-interactive network downloader which is used to download files from the server even when the user has not logged on to the system and it can work in the background.