

Homework# 1  
CSC3320 System-Level Programming

Submission instructions:

1. Create a Google doc for each homework assignment 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 in your document TWO POINTS WILL BE DEDUCTED per submission.
4. Keep this page 1 intact on all your submissions. If this *submissions instructions* page is missing in your submission TWO POINTS WILL BE DEDUCTED per submission.
5. Each homework will typically have 2-3 PARTS, where each PART focuses on specific topic(s).
6. Start your responses to each PART on a new page.
7. If you are being asked to write code copy the code into a separate txt file and submit that as well.
8. If you are being asked to test code or run specific commands or scripts, provide the evidence of your outputs through a screenshot and copy the same into the document.
9. Upon completion, download a .PDF version of the document and submit the same.

Full Name: Cassandra Lily Lundberg

Campus ID: clundberg3

Panther #: 002345582

## **PART 1:**

- 1. UNIX is a base operating system in which each utility is designed to do one thing in a pipe formation to create pipelines of data. Many different designers utilized the UNIX source code to enhance it and create improvements, marketing new operating systems which stemmed from UNIX. Some of these operating systems are Apollo which derived from the 4.2 BSD series, OSF/1 which came from IBM AIX of version 2 in System V series, and Sun OS which formed from both System V series and BSD series. Linux on the other hand is a completely different operating system which behaves like UNIX and was created by many different programmers but shares no direct common code with UNIX.**
- 2. The pipeline mechanism in UNIX is a notion that a program should do one thing and do it well where complex tasks should be done by all the programs together. The designers of UNIX created a pipe in UNIX which allows a user to specify the output of one process as the input of another process. As the processes' outputs are connected to other processes' inputs, the programs create a pipeline feeding information from one process to another. For example in the command: (grep 'POINTS' homework\_instruction.txt | wc -l ) the grep command finds all the lines with the string 'POINTS' and utilizes the output as an input for the second part of the command (wc -l). Now the first command's output is the second command's input and the lines containing 'POINTS' are counted to display the total number of lines in the text homework\_instructions.txt that contain the string 'POINTS'.**
- 3. In a Linux system, the /bin is a sub directory root that contains the binary executable files that must be available in order to obtain minimal functionality such as booting and repairing a system. The /dev subdirectory is a location of special or device files such as a speaker device where any data written to the file will be directed to the speaker. The /boot subdirectory contains everything required for the boot process such as storing data that is used before the kernel begins executing user-mode programs. The /usr subdirectory is the default location of the user home directories which consists of several other subdirectories containing various commands and data files. The /etc subdirectory is a central location for all configuration files and the central nerve of Linux. The /mnt subdirectory is intended for temporary mount points for mounting storage devices such as USB or floppy disks. The /sbin subdirectory contains executable programs such as administrative tools that should only be**

made available to the root. The /var subdirectory contains files that the system writes with data during the course of its operation.

4. In a UNIX system, multitasking means multiple processes running simultaneously. As most operating systems can do this, UNIX incorporates multi-user ,multi-tasking where multiple users run multiple processes simultaneously. This can be done by dividing the processor's time between tasks into slices so it looks as though everything is being multitasked as multiple users access the same computer at the same time through accounts.
5. The permission setting -rwxr-xr-x means that the file is a regular file type and the user can read, write, and execute, the group users can read and execute, and the public can read and execute. Both the group users and the public users cannot write, only the owner is allowed all three permissions. To change a file to this setting using the octal representation the command needed to be entered is: `chmod 755 fileName`.
6. In a directory, if the read permission has been allowed, the user may see the list of file names contained in that directory. If a write permission has been allowed, the user may add files or remove files from the specific directory. If a execute permission is allowed, the user may access any files in the directory or any of its subdirectories.

## Part II-a:

7. Example: abbababababababa, aba, abbbbba, aa, The matched string should begin and end with 'a' then the next character should be any character in the list 'a' and 'b', matching 0 or more times.
8. Example: a, abc, The matched string should begin with 'a' then followed by 'bc' 0 or 1 times.
9. Example: wiiiiiiiiiii, wind, kind, The first character should be any character followed by any character in the list 'i', 'n', or 'd', matching 0 or more times.
10. Example: hello, hi, The matched string should first be any character in the range from 'a' to 'z' between 1 or more times, followed by any character in the range from 'a' to 'z'.
11. Example: et, eit, eight, aut, aught, The matched string should match one or more occurrences of any character from 'a' to 'z' followed by a group of expressions from 'a' to 'z' between 1 and many number of times.
12. Example: atc, aobcbcbc, apccc, asbb, The matches string matches the beginning character as 'a' followed by any character followed by any character in the list 'b' and 'c' between one and many number of times.
13. Example: at1, ar2, ay3, ao4, ap5, ax6, az0, aw7, aq8, av9, The matches string matches the first character as 'a' followed by any character, followed by any number from '0' to '9'.
14. Example: goodbye!, hello!, what?, yes., The matched string matches any character in the range from 'a' to 'z' between one and many number of times followed by the character '.' or '?' or '!'.
15. Example: 'store. Then', 'really? So', 'end! Lastly', The matched string matches any character in the range from 'a' to 'z' between one and many number of times followed by the character '.' or '?' or '!', then followed by any white space character 0 or many times, followed by any character from uppercase 'A' to uppercase 'Z'. This extended expression basically finds any sentence with a sentence after it.
16. Example: very very cool bad weather, very cool good weather, The matched string matches the word 'very' one or many times followed by the word 'cool' one or none times, followed by either the string 'good' or the string 'bad, then followed by the string 'weather'.
17. Example: -12, 12, -123456789, The matched string matched one

or none of the character '-' followed by any character between '0' and '9' between one and many times. This extended expression matches a number regardless of the sign either positive or negative.

18. Example: -0.9, 0.9, 1.99999, -1, -2.345, This matched string matches one or none of the character '-' followed by any character between '0' and '9' between zero and many times, followed by a '.' period zero or one time, followed by any character from '0' to '9' between zero and many times. This extended expression basically matches negative and positive decimal numbers.

**Part II-b:**

19. `(^http:\\\\)?.(.edu)$`

20. `\+[0-9]+`

21. `((\\/.*)\\/*)`

22. `(^[a-z]*-?)[a-z]*[0-9]*-*){1,10}`

23.

Phone number in format: 9999999999

`[0-9]{10}$`

Phone number in format: 999-999-9999

`[0-9]{3}-[0-9]{3}-[0-9]{4}$`

Phone number in format: (999)-999-9999

`\([0-9]{3}\)-[0-9]{3}-[0-9]{4}`

### **Part III:**

#### **Step#1:**

First I created a specific directory, just for my case to keep all my homework files for the semester and called it "Homework\_Files". I did this by going into my home directory (`cd ~`) then typing (`mkdir Homework_Files`) then I went into that directory for the assignment. I typed (`cd Homework_Files`) then I created the assigned file "homeworks" in that directory (`mkdir homeworks`). Next I went into that directory (`cd homeworks`) to create my `homework_instructions.txt` file using vi editor.

**File name and Directory name:**

```
[clundberg3@gsuad.gsu.edu@snowball homeworks]$ pwd
/home/clundberg3/Homework_Files/homeworks
[clundberg3@gsuad.gsu.edu@snowball homeworks]$ ls
homework_instructions.txt
[clundberg3@gsuad.gsu.edu@snowball homeworks]$
```

#### **Step#2:**

Now I entered the vi editor just by typing (`vi`). Then I pressed the lowercase (`i`) to enter text entry mode and begin typing. Once my text was complete I pressed (`ESC`) to go back into command mode and entered (`:w homework_instructions.txt`) to save and name the file.

## Vi Editor contents:

```
cclundberg3@gsuad.gsu.edu@snowball:~/Homework_Files/homeworks
```

Submission Instructions:

1. Create a Google doc for each homework assignment 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 in your document TWO POINTS WILL BE DEDUCTED per submission.
4. Keep this page 1 intact on all submissions. If this submission instruction page is missing in your submission TWO POINTS WILL BE DEDUCTED per submission.
5. Each homework will typically have 2-3 PARTS, where each PART focuses on specific topic(s).
6. Start your responses to each PART on a new page.
7. If you are being asked to write code copy the code into a specific txt file and submit that as well.
8. If you are being asked to test code or run specific commands or scripts, provide the evidence of your outputs through a screenshot and copy the same into the document.
9. Upon completion, download a .PDF version of the document and submit the same.

Full Name: Cassandra Lily Lundberg

Campus ID: clundberg3

Panther # 002345582

~  
~  
~  
~  
~  
~  
~  
~

"homework instructions.txt" 16L, 1091C 1,1 All

### Step#3:

Now to change the permissions of the file I used the `chmod` command and typed (`chmod 744 homework_instructions.txt`) to change the permissions. The octal number 744 represents the owner has all 3 permissions allowed, read, write, and execute. The group users and public however only have read so the two 4's represent their permissions. Then I entered the command (`ls -l homework_instructions.txt`) to check the permissions were `-rwxr--r--`

### Permissions of File:

```
[clundberg3@gsuad.gsu.edu@snowball homeworks]$ ls -l homework_instructions.txt
-rwxr--r--. 1 clundberg3@gsuad.gsu.edu clundberg3@gsuad.gsu.edu 1091 Sep 18 17:2
2 homework_instructions.txt
[clundberg3@gsuad.gsu.edu@snowball homeworks]$
```



#### Step#4:

Lastly I used grep to search for the string "POINTS" in the text file using the command (grep 'POINTS' homework\_instructions.txt). This found the string in the vi text file twice.

Searching for 'POINTS':

```
clundberg3@gsuad.gsu.edu@snowball:~/Homework_Files/homeworks
[clundberg3@gsuad.gsu.edu@snowball Homework_Files]$ cd homeworks
[clundberg3@gsuad.gsu.edu@snowball homeworks]$ ls
homework_instructions.txt1
[clundberg3@gsuad.gsu.edu@snowball homeworks]$ vi homework_instructions.txt1
[clundberg3@gsuad.gsu.edu@snowball homeworks]$ vi homework_instructions.txt1
[clundberg3@gsuad.gsu.edu@snowball homeworks]$ ls
homework_instructions.txt homework_instructions.txt1
[clundberg3@gsuad.gsu.edu@snowball homeworks]$ vi homework_instructions.txt
[clundberg3@gsuad.gsu.edu@snowball homeworks]$ rm homework_instructions.txt1
[clundberg3@gsuad.gsu.edu@snowball homeworks]$ ls
homework_instructions.txt
[clundberg3@gsuad.gsu.edu@snowball homeworks]$ chmod 744 homework_instructions.t
xt
[clundberg3@gsuad.gsu.edu@snowball homeworks]$ ls -l homework_instructions.txt
-rwxr--r--. 1 clundberg3@gsuad.gsu.edu clundberg3@gsuad.gsu.edu 1091 Sep 18 17:2
2 homework_instructions.txt
[clundberg3@gsuad.gsu.edu@snowball homeworks]$ pwd
/home/clundberg3/Homework_Files/homeworks
[clundberg3@gsuad.gsu.edu@snowball homeworks]$ ls
homework_instructions.txt
[clundberg3@gsuad.gsu.edu@snowball homeworks]$ vi homework_instructions.txt
[clundberg3@gsuad.gsu.edu@snowball homeworks]$ ls -l homework_instructions.txt
-rwxr--r--. 1 clundberg3@gsuad.gsu.edu clundberg3@gsuad.gsu.edu 1091 Sep 18 17:2
2 homework_instructions.txt
[clundberg3@gsuad.gsu.edu@snowball homeworks]$ vi homework_instructions.txt
[clundberg3@gsuad.gsu.edu@snowball homeworks]$ grep 'POINTS' homework_instructio
ns.txt
      3. Fill in your name, Campus ID and panther # in the fields provided. If
this information is missing in your document TWO POINTS WILL BE DEDUCTED per su
bmission.
      4. Keep this page 1 intact on all submissions. If this submissions instr
uction page is missing in your submission TWO POINTS WILL BE DEDUCTED per submis
sion.
[clundberg3@gsuad.gsu.edu@snowball homeworks]$
```