**CSc 3320: Systems Programming**

Spring 2021

Homework

# 3: Total points 100

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: Noah Platt

Campus ID: Nplatt1

Panther #: 002-31-0477

**PART I**

|  |  |  |  |
| --- | --- | --- | --- |
| 1. | grep | The command searches for patterns in file based on basic regular expressions. For example, searching for patterns matching ‘[a-m]nd’. |  |
| 2. | egrep | Searches for patterns in file based on extended regular expressions. For example, searching for patterns matching ‘off|will’. |  |
| 3. | fgrep | Searches for patterns in file based on fixed strings. For example, searching for patterns matching the text string ‘too.’. |  |
| 4. | uniq | Displays file with all adjacent repeated lines collapsed into a single line. For example, collapsing repeated lines of ‘dolphin elephant’ and ‘pig pig’ into single lines. |  |
| 5. | sort | Sorts lines in files based on criteria. For example, sorting a list of dates in alphabetical order. |  |
| 6. | cmp | Tests whether two files are the same. For example, testing whether two file lady1 and lady2 are the same. |  |
| 7. | diff | Compares two files and outputs edits needed to change the first file to the second file. For example, finding the changes needed to make lady1 turn into lady2. |  |
| 8. | find | Locate files based on pattern and perform actions on the located files. For example, printing al c source files in the current and subdirectories. |  |
| 9. | cpio | Creates cpio format files, which can backup small subdirectories. For example, backing up all sh files in the current directory. |  |
| 10. | tar | Designed to create archive on magnetic tape. Creates tar file archive. For example, creating a tar archive of all sh files in current directory. |  |
| 11. | dump | Creates a archive files on multiple backup systems, standard being the /dev/rmt0 directory. For example, creating a backup of all sh files. |  |
| 12. | crontab | Schedules a series of jobs to be executed periodically. For example, setting a greeting every Monday morning and send email. |  |
| 13. | cron | Same as crontab. Type of file crontab reads to create schedules. |  |
| 14. | at | Schedules one-time execution of commands or scripts. For example, setting the system to print “2 minutes have passed” in 2 minutes. |  |
| 15. | awk | Scans lines of the input and performs action on all lines that meet criteria. For example, printing the first word of every line in a file. |  |
| 16. | ln | Creates links between files. For example, creating file sleep.txt that opens file grepfile. |  |
| 17. | whoami | Displays the name of the owner of the shell. |  |
| 18. | su | Create a subshell owned by a different user. For example, entering the root shell, which I am unable to do as I do not have the superuser password. |  |
| 19. | biff | Enables or disables instant mail notification. |  |
| 20. | compress | Compress files into a compact format. For example, compressing hello.sh. |  |
| 21. | uncompress | Uncompress previously compressed files. For example, uncompressing hello.sh. |  |
| 22. | gzip | Compress files into a more compact size using unrestricted algorithm. For example, compressing hello.sh. |  |
| 23. | gunzip | Uncompress previously compressed files using gzip. For example, uncompressing hello.sh. |  |
| 24. | crypt | Creates an encoded version of a text file. For example, encrypting hello.sh. |  |
| 25. | sed | Scan a file and edits all lines that match a particular condition. For example, replacing all spaces with tabs. |  |
| 26. | tr | Maps all the characters of one set to another set. For example, turning all lowercase letters to uppercase letters. |  |
| 27. | ul | Transforms input with underline characters so that it can be displayed correctly. For example, getting rid of underlines in the text in man who. |  |
| 28. | od | Displays contents of a non-text file in various formats. For example, displaying /bin/od in octal. |  |
| 29. | mount | Lists connected devices, and can add device files into root hierarchy. For example, mounting /dev/dsk2 to the /usr directory. |  |
| 20. | umount | Unmounts device files. For example, unmounting /dev/dsk2. |  |
| 31. | time | Reports execution time of commands. For example, seeing the time it takes to list the files in the directory. |  |
| 32. | perl | Allows the use of the perl script. For example, using perl to print “hello world.” |  |

**Part II**

|  |  |  |  |
| --- | --- | --- | --- |
| 1. | chsh | Changes the default shell. For example, change to the Bourne shell. |  |
| 2. | echo | Displays arguments to standard output. For example, outputting “hello world”. |  |
| 3. | ps | Generates process status information. |  |
| 4. | sleep | Sleeps command line for an amount of time. For example, sleeping for 10 seconds. |  |
| 5. | kill | Terminates a process before it finishes. For example, printing a list of valid kill signals. |  |
| 6. | nohup | Executes a command and makes it immune to hangup and terminate commands. For example, making sleep command immune. |  |

**Part III**

|  |  |  |  |
| --- | --- | --- | --- |
| 1. | expr | Evaluates an expression and sends result to standard output. For example, computing (4 \* 5)/3. |  |
| 2. | test | Tests if a given expression is true, and if so evaluates to 0. For example, testing to see if hello.sh exists. |  |