CSc 3320: Systems Programming

Fall 2021

Homework

1: 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: Paul Ofremu Jr.

Campus ID: pofremu1

Panther #: 002513676

PART 1

1) Unix refers to the original operating system. Linux is but a flavor of the Unix operating system. Linux doesn't share any of the code of the original and is open-source.
2) The pipe mechanism in Unix, allows the output of one process to be used as the input for another process. Ex: The command ***who sort.*** The **who** utility outputs an unsorted list and that is used as the input for the **sort** utility that takes a list as an input and outputs the sorted list.
3) :
    a) /bin - holds the binaries that run on the machine
    b) /dev - holds file related to the hardware
    c) /boot - where the bootloader and kernel are stored
    d) /usr - holds user applications and tools
    e) /etc - holds startup scripts and config files
    f) /mnt - holds temporary mounting files
    g) /sbin - holds binaries reserved for the system admin
    h) /var - holds variable data and cached data
4) Multitasking is the ability to run multiple programs at one time. Multi-user is allowing more than a single user to work at any given time.
5) -rwxr-xr-x : This term gives the user read, write, and execution permissions, the groups read and execution permissions, and others read and execution permissions. The unix command to set these permissions is **chmod 755**
6) Read for directories is being able to list its files. Write for directories is being able to add or remove files. Execute for directories is being able to have access to any of its files.

Part II-a

Regular Expression
Find outcomes for each given basic/extended regular expression (maybe multiple correct answers)

Points per question: 2.5

Example:
'ab+a'（extended regex）

***Answer**: aba , abba ; Pattern : The matched string should begin and end with 'a' and 'b' occurs at least once between leading and ending 'a')*

Note: 7) to 10) are basic regexes; Note: 11) to 18) are extended regexes.

7) 'a[ab]*a' ***Answer:*** abba , aaaa , ababa; Pattern: String begins and ends with 'a' and any number of 'a' or 'b' occurs in between.

8) 'a(bc)?' ***Answer:*** abc; Pattern: String matches 'abc'

9) '.[ind]*' ***Answer:*** snid , 1innd , Tdi; Pattern: String begins with any character is followed by any number of 'i', 'n', or 'd'

10) '[a-z]+[a-z]' ***Answer:*** eufnla , fskjn , pefjm; Pattern: String begins with any number of characters in range [a-z] followed by any character in range [a-z]

11) '[a-z] (\+[a-z])+' ***Answer:*** a+c+w, i+w+v+s, e+e; Pattern:

String begins with any character in range [a-z] and followed by

any number of ('+' followed by any character in range [a-z])

12) 'a.[bc]+' ***Answer:*** abc , a1ccb , a*cb; Pattern: String begins

with an 'a' followed by any character then followed by 1 or more

'b' or 'c'

13) 'a.[0-9]' ***Answer:*** ae2 , a34 , aa0; Pattern: String begins with

an 'a' followed by any character then followed by a number in

range [0-9]

14) '[a-z]+[\.\?!]' ***Answer:*** abc.?! , efio.?! , w.?!; Pattern: String

begins with one or more letters in range [a-z] then followed by

the characters '.?!'

15) '[a-z]+[\.\?!]\s*[A-Z]' ***Answer:*** are.?! T, fkdr.?! O, q.?! Q

Pattern: String begins with one or more letters in range [a-z] then

followed by '.?!' then followed by 0 or more spaces and ends with

a capital letter in range [A-Z]

16) '(very)+(cool)?(good|bad) weather' ***Answer:*** very cool good

weather, very bad weather, very cool bad weather; Pattern:

String begins with 'very ' then may or may not be followed by

'cool ' then followed by either 'good ' or 'bad ' then followed by

' weather'

17) '-?[0-9]+' ***Answer:*** -2312, 5466, -1; Pattern: String may or may not begin with '-' the followed by one or more numbers in range [0-9]

18) '-?[0-9]*\.?[0-9]*' ***Answer:*** -594.459, 3838 , -.0; Pattern: String may or may not begin with '-' followed by 0 or more numbers in range [0-9] then may or may not be followed by a '.' followed by 0 or more numbers in range [0-9]

Part II-b

Regular Expression
Write down the extended regular expression for following questions.
E.g. Social security number in the format of 999-99-9999. Answer: [0-9]{3}-[0-9]{2}-[0-9]{4}

Points per question: 5

19) Valid URL beginning with "http://" and ending with ".edu" (e.g. http://cs.gsu.edu, http://gsu.edu)

'(http:\/\/).+(\.edu)'

20) Non-negative integers. (e.g. 0, +1, 3320) '\+?[0-9]+'

21) A valid absolute pathname in Unix (e.g. /home/ylong4, /test/try.c)
'(\/[^*]*)+'

22) Identifiers which can be between 1 and 10 characters long, must start with a letter or an underscore. The following characters can be letters or underscores or digits. (e.g. number, _name1, isOK). '[a-zA-Z_]{1}[a-zA-Z0-9_]{0,9}'

23) Phone number in any of the following format: 9999999999,999-999-9999, (999)-999-9999. (Note: all of these formats should be matched by a single regular expression) '\(?[0-9]{3}\)?-?[0-9]{3}-?[0-9]{4}'

Part III

Programming
Points per question: 15

24. Create a file named homework_instructions.txt using VI editor and type in it all the submission instructions from page1 of this document. Save the file in a directory named *homeworks* that you would have created. Set the permissions for this file such that only you can edit the file while anybody can only read. Find and list (on the command prompt) all the statements that contain the word POINTS. Submit your answer as a description of what you did in a sequential manner (e.g. Step1 … Step 2… and so on..). Add a screenshot to your answer as a proof of evidence.

1. mkdir homeworks && cd homeworks
2. touch homework_instructions.txt && vi homework_instructions.txt
3. Press 'i' to enter insert mode
4. Paste instructions
5. Press 'esc' to exit insert mode
6. Press ':' and enter 'wq!' to save and quit

7. grep ‘POINTS’ homework_instructions.txt