Cassandra Lundberg

System Level Programming

09/12/21

Lab Assignment# 3: Post Lab

Part2:

3) To open any file with the vi editor the vi command is used along with the file name. So the command is typed: vi RealEstate.csv

4) In vi text editor the “$” symbol denotes the line number of the last line of the text file. When the command: :$ is typed, the cursor moved to the last line of the text and at the bottom right corner of the terminal screen the line number is displayed.

5) Since the “$” denotes the last line of the text, the symbol “.” denotes the current line in which the cursor contains. Typing the command: :. displays the current line number at the bottom right of the terminal screen.

6)There are two directions in vi editor to search for specific words or strings inside of a text. To search forwards, where the cursor is toward the top of the text and the editor will search down throughout the text the “/” symbol is used where the command looks like “: /string/. However, when the cursor is at the end of the text, as in this case, it is most efficient to search backwards up the text. This is done by using the “?” symbol instead of the slash symbol. The correct command to use is: :?111 EAST?

When this is typed the line number in which 111 EASTDALE DRIVE is on is line number 7991.

Since this string 111 EAST seems to be repeated throughout the text, I also tried moving the cursor to the beginning of the text using the command :1 and then typing :/111 EAST/ to get the string when searching the text forward at line number 58. I am not sure which one is the answer, but it depends on where the cursor is and which direction the user searches either forward or backwards.

There are many ways to delete a line in the text. First I could position the cursor on the line I’d like to delete then enter the command dd . However, if a number is put before the dd then the vi editor will delete that number of lines from the text. The other way to delete the line here is to find out the line number that has “111 EAST” then enter the command :58d or :7991d depending on which answer is the correct one. In this case I’m going to assume the string on line 58 was meant to be searched and deleted but this should have been specified.

7) To move the cursor to line 50, the command is typed :50

8)To replace all comas with colons from line 50 to 54, a global search and replace operation must be done. The command should include the range of line numbers from 50 to 54 and must be global to replace all occurrences and not just the first occurrence. So the command must be typed:

:50,54s/,/:/g

9) To copy line number 50 to 54 to the end of the file the command :50,54y is used. As these 5 lines are now copied to the paste buffer, the lines can be pasted wherever they need to be. In this case the lines need to be pasted at the end of the text so the command :9315pu is used, where 9315 is the last line of the text.

10) To remove line numbers 50 to 54, the command to remove a block of lines is removed where the range from 50 to 54 is entered. The command is :50,54d

11) To enter text mode in the vi editor, one of seven keys must be pressed: i, I, a, A, o, O, and R. To insert a new text in between lines 1 and 2, there are many ways to achieve this. First the cursor must be moved to line number 1 in order for a new line to be created and entered. So the command :1 is first used. Then “o” can be entered to create a blank space where the cursor moves to in prompt of text. Then lastly, the text “Recorded in year 2008” can be typed as the vi editor is still in text entry mode. So the commands look like this:

:1

o

Recorded in year 2008

After the text is entered in between lines 1 and 2, the text entry mode of vi editor can be exited by pressing the ESC key.

12) To switch back to command mode from text editor mode of vi editor, the ESC key can be pressed:

Esc

13) To save the file and quit vi editor with the current file name, the command is typed:

:wq

Part 3:

10)

Graphical user interface

Description automatically generated

12) The file permissions look like this: -r - - r - - r - -

The first dash represents a regular file type. The first set of three symbols after the first dash, r - - represent the user’s permissions of the file Try.c. The r means the user can read the file and the other two dashes mean the user cannot write or execute on the file. The next set of three symbols, r - - represent the same thing but to the group. This means the group can read but not write or execute on the file. The last set of three symbols, r - - also mean the same thing but to the public or outside users. This represents the public can read the file but not write or execute on the file.

15) The error message is:

E45: ‘readonly’ option is set (add ! to override)

To quit the vi editor without saving the command entered should be: :q!

16) To add write permissions to all the users for the file Try.c, the command should be typed:

chmod a+w Try.c

17) The file permissions for Try.c looks like -rw-rw-rw-

This is because the file is still a regular type and adding the write permissions to all the users did not affect the read permissions that were already granted using the octal form. The first three symbols after the first dash, rw- mean the user has permission to read and write but not execute. The next three symbols, rw- mean the group has permissions to read and write but not to execute on the file. The last three symbols, rw- mean the public has permissions to read and write but not to execute.

Part 4:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Owner’s Permissions | ls  A. Read Contents | mkdir  B. Create sub-directory | cat >  C. Create file | rm  D. Remove Contents | cp  E. Copy contents from | cd  F. Enter into directory |
| --- | N | N | N | N/A | N/A | N |
| --x | N | N | N | N/A | N/A | Y |
| -w- | N | N | N | N/A | N/A | N |
| -wx | N | Y | Y | Y | Y | Y |
| r-- | N | N | N | N/A | N/A | N |
| r-x | Y | N | Y | N/A | Y | Y |
| rw- | N | N | N | N/A | N/A | N |
| rwx | Y | Y | Y | Y | Y | Y |