CS 69001: Computing Lab I

Assignment – 4A: Introduction to SED

Introduction:

After completing the last two assignments on command shell, you are familiar with how a Unix type shell works. In this assignment you'll get more exposure on another Unix filter for file processing named ***sed***, that process a file based on regular expressions. In this assignment, we'll work with the ***sed*** command that is used to replace or delete some contents in a file based on regular expressions.

Objective:

To understand pattern matching using ***sed***.

Description:

Integrate the following things in your mini command shell.

1. **SED :**
   a. **Substitution**: **[20 Marks]**

The format for the substitute command is as follows:

```
sed
'[optional-regex/]s/regex-pattern/regex-replacement/
[flags]' file_name
```

The flags can be any of the following:

n	Replace n^{th} instance of "pattern" with "replacement"
g	Replace all instances of "pattern" with "replacement"

Regular expression (regex): A regular expression may be simple string or followed by one of several repetitions. It should support the following o**perators:**

? The preceding item is optional and matched at most once.

* The preceding item will be matched zero or more times.

+ The preceding item will be matched one or more times.

Example 1:

```
>cat file
the black cat was chased by the brown dog

>mysed  's/black/white/g' file
the white cat was chased by the brown dog
```

Example 2:

```
>cat file
the black cat was chased by the brown dog.
the black cat was not chased by the brown dog

>mysed  '/not/s/black/white/g' file
the black cat was chased by the brown dog.
the white cat was not chased by the brown dog.
```

Here, the substitution is only applied to lines matching the regular expression ‘not’. Hence it is not applied to the first line.

b. **Delete:** **[20 Marks]**

The format for the substitute command is as follows:

mysed [line-number]d

It should delete that exact line.

mysed [line-number1, line-number2]d

It should delete line-number1 and line-number2.

mysed [line-number1...line-number2]d

It should delete all the lines from line-number1 to line-number2.

mysed \[line-number1...line-number2]d

It should delete all the lines except from line-number1 to line-number2. This do not delete option should work for previously mentioned formats also.

Example 1:

>cat file
line 1 (one)
line 2 (two)
line 3 (three)
line 4 (four)

>mysed [1,2]d file
line 3 (three)
line 4 (four)

>cat file2
line 1 (one)
line 2 (two)
line 3 (three)
line 4 (four)

>mysed [3]d file2
line 1 (one)
line 2 (two)
line 4 (four)

>cat file1
line 1 (one)
line 2 (two)
line 3 (three)
line 4 (four)

> mysed [1…3]d file
line 4 (four)

>cat file3
line 1 (one)
line 2 (two)
line 3 (three)
line 4 (four)

> mysed \[1…3]d file
line 1 (one)
line 2 (two)
line 3 (three)

Assignment – 4B: Introduction to UNIX Type Signals

Objective: These assignments will give you an idea about how UNIX Signals work.

1. Write a C program to create a user-defined signal (SIGUSR1). On receiving the signal, the program should give some appropriate output to say that the signal is received properly. [output should be like “SIGUSR1 Received”]. You can test your code using the following command:
   “kill –USR1 pid”
   where *pid* is process ID of the program. **[20 Marks]**

2. Normally, on pressing “crtl + c” in terminal, it terminates the current process executing. Write a C program that will not respond on pressing “ctrl + c” but it will quit on pressing “ctrl + \”.
   [10 Marks]

3. Write a C program to do the following things:
   a. On pressing “ctrl + c” first time: create a file called “temp” with a message“Interrupt” in the same file.
   b. On pressing “crtl + c” second time: it should print “I am taking rest now”.
   c. On pressing “ctrl +c” third time: it should do the default work of “ctrl + c”.
   Note: Do not use count to achieve this. **[10 Marks]**

Deliverables:

It is mandatory that your code should follow proper indentation and commenting. The C programs with name as “myprogram_rollno.c” along with the makefile should be submitted in a compressed tar file format with the name rollno_a4.tar.gz . There will be deductions in the awarded marks, if you fail to do so. All the source c files should remain in the same directory. You should submit a report mentioning all the signals and your findings about those.

Happy Learning!