Stream editor sed

⊙ Type	📔 Lecture
🗓 Date	@February 20, 2022
☰ Lecture #	1
🔗 Lecture URL	https://youtu.be/uCJjZ3cbaAw
🔗 Notion URL	https://21f1003586.notion.site/Stream-editor-sed-4d1c90712d98497a9bd56a4dd192e134
# Week #	7

`sed`

It is a language for processing text streams

Introduction

- It is a programming language
- `sed` is an abbreviation of **s**tream **ed**itor
- It is a part of POSIX
- `sed` precedes `awk`

Execution Model

- Input stream is a set of lines

- Each line is a sequence of characters

- Two data buffers are maintained → active **pattern** space and auxiliary **hold** space

- For each line of input, an execution cycle is performed loading the line into the pattern space

- During each cycle, all the statements in the script are executed in the sequence for matching address pattern for actions specified with the options provided

Usage

- Single line at the command line

  `sed -e 's/hello/world/g' input.txt`

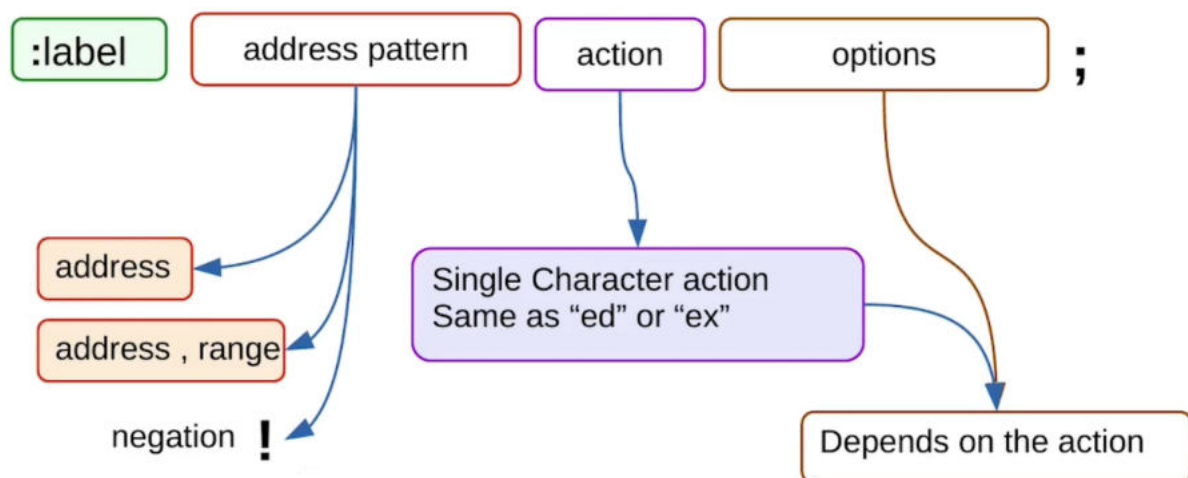- Script interpreted by sed

  `sed -f ./myscript.sed input.txt`

  Where `myscript.sed` could be ...

  ```
  #!/usr/bin/sed -f
  2,8s/hello/world/g
  ```
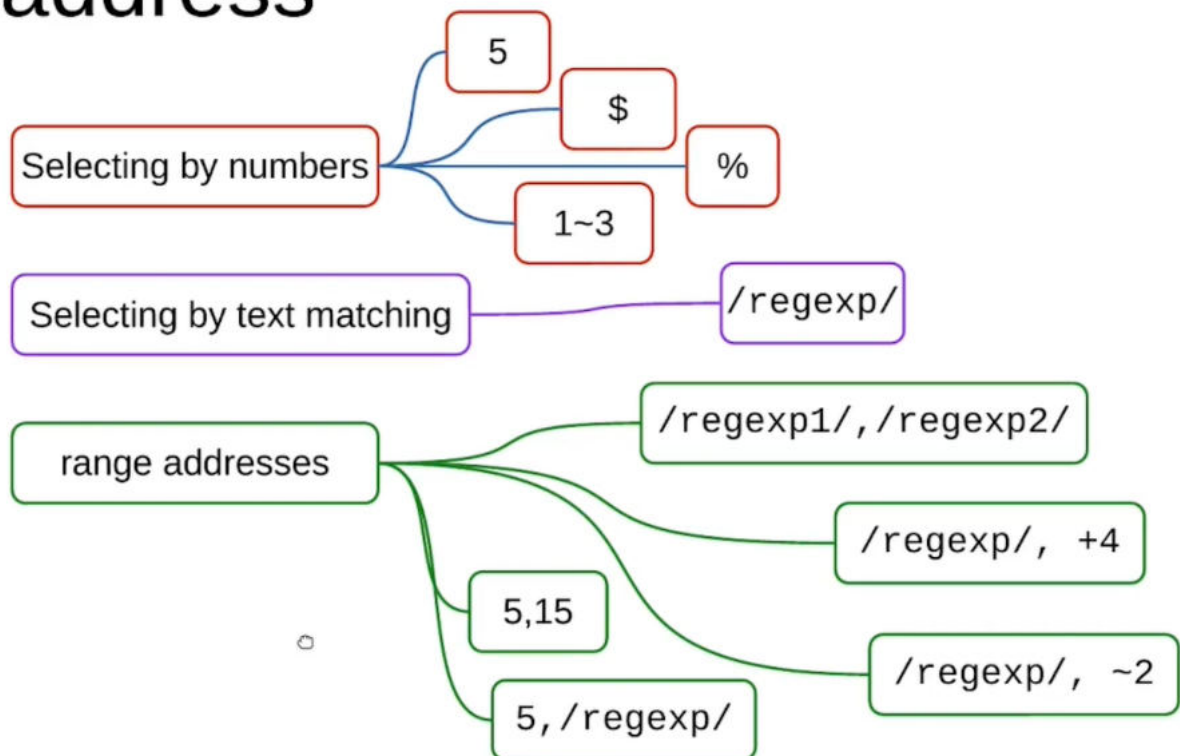
`sed` statements

Grouping commands

`{ cmd; cmd; }`

address

- `5` → select the 5th line

- `$` → select the last line

- `%` → select all the lines

- `1~3` → starting from the 1st line, select every 3rd line

- `/regex/` → select the line by matching the regular expression

- `/regex1/, /regex2/` → from the line of the 1st occurrence of `regex1` to the line of the 1st occurrence of `regex2`

- `/regex/, +4` → action has to be taken on the next 4 lines, starting from the first match of `regex`

- `/regex/, ~2` → the line where the 1st regular expression is matched, the next line which is a multiple of 2 will be matched

- `5, 15` → execute from the 5th line to 15th line

- `5, /regex/` → from the 5th line till the first occurrence of `regex`

actions

p	Print the pattern space
d	Delete the pattern space
s	Substitute using regex match s/pattern/replacement/g
=	Print current input line number, \n
#	comment
i	Insert above current line
a	Append below current line
c	Change current line

programming

b *label*	Branch unconditionally to *label*
:label	Specify location of *label* for branch command
N	Add a new line to the pattern space and append next line of input into it.
q	Exit sed without processing any more commands or input lines
t *label*	Branch to *label* only if there was a successful substitution was made
T *label*	Branch to *label* only if there was **no** successful substitution was made
w *filename*	Write pattern space to *filename*
x	Exchange the contents of hold and pattern spaces

bash **+** **sed**

- Including sed inside shell script

- heredoc feature

- Use with other shell scripts on the command line using pipe

Examples

`sample.txt`

```
L1 Linux is a free, opensource, secure and reliable os
L2 text editing: vim or emacs in place of notepad
L3 text processing: latex, libreoffice write in place of microsoft word
L4 spreadsheet: libreoffice calc in place of microsoft excel
L5 slides: libreoffice impress in place of microsoft powerpoint
L6 image processing: gimp, inkscape, imagej in place of adobe photoshop
L7 browsing: mozilla firefox in place of microsoft edge
L8 email: mozilla thunderbird in place of microsoft outlook
L9 symbolic math: sympy, sagemath in place of maple, mathematica
L10 array math: octave, numpy scilab in place of matlab
L11 literature: jabref, calibre in place of mendeley
L12 video editing: kdenlive in place of camtasia
L13 screen recording: obs in place of camtasia
L14 fetch files: wget, curl
L15 security: ufw, netstat, nmap
```

Default action is to print

`sed -e "" sample.txt`

The above command takes an optional `-n` flag to negate whatever command we give to `sed`

Suppress automatic printing of pattern space

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -n -e "" sample.txt
```

Print every line with it's line number

```
sed -e "=" sample.txt
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -e "=" sample.txt
1
L1 Linux is a free, opensource, secure and reliable os
2
L2 text editing: vim or emacs in place of notepad
3
L3 text processing: latex, libreoffice write in place of microsoft word
4
L4 spreadsheet: libreoffice calc in place of microsoft excel
5
L5 slides: libreoffice impress in place of microsoft powerpoint
6
L6 image processing: gimp, inkscape, imagej in place of adobe photoshop
7
L7 browsing: mozilla firefox in place of microsoft edge
8
L8 email: mozilla thunderbird in place of microsoft outlook
9
L9 symbolic math: sympy, sagemath in place of maple, mathematica
10
L10 array math: octave, numpy scilab in place of matlab
11
L11 literature: jabref, calibre in place of mendeley
12
L12 video editing: kdenlive in place of camtasia
13
L13 screen recording: obs in place of camtasia
14
L14 fetch files: wget, curl
15
L15 security: ufw, netstat, nmap
```

Print the 5th line, and don't print anything else by default

```
sed -n -e "5p" sample.txt
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -n -e "5p" sample.txt
L5 slides: libreoffice impress in place of microsoft powerpoint
```

If we use `sed -e "5p" sample.txt` , line #5 will be printed twice

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -e "5p" sample.txt
L1 Linux is a free, opensource, secure and reliable os
L2 text editing: vim or emacs in place of notepad
L3 text processing: latex, libreoffice write in place of microsoft word
L4 spreadsheet: libreoffice calc in place of microsoft excel
L5 slides: libreoffice impress in place of microsoft powerpoint
L5 slides: libreoffice impress in place of microsoft powerpoint
L6 image processing: gimp, inkscape, imagej in place of adobe photoshop
L7 browsing: mozilla firefox in place of microsoft edge
L8 email: mozilla thunderbird in place of microsoft outlook
L9 symbolic math: sympy, sagemath in place of maple, mathematica
L10 array math: octave, numpy scilab in place of matlab
L11 literature: jabref, calibre in place of mendeley
L12 video editing: kdenlive in place of camtasia
L13 screen recording: obs in place of camtasia
L14 fetch files: wget, curl
L15 security: ufw, netstat, nmap
```

We can also replace the double quotes with single quotes

We also use single quotes when we don't want the shell to interpret certain special characters

Print every line except line #5

`sed -n -e '5!p' sample.txt`

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -n -e '5!p' sample.txt
L1 Linux is a free, opensource, secure and reliable os
L2 text editing: vim or emacs in place of notepad
L3 text processing: latex, libreoffice write in place of microsoft word
L4 spreadsheet: libreoffice calc in place of microsoft excel
L6 image processing: gimp, inkscape, imagej in place of adobe photoshop
L7 browsing: mozilla firefox in place of microsoft edge
L8 email: mozilla thunderbird in place of microsoft outlook
L9 symbolic math: sympy, sagemath in place of maple, mathematica
L10 array math: octave, numpy scilab in place of matlab
L11 literature: jabref, calibre in place of mendeley
L12 video editing: kdenlive in place of camtasia
L13 screen recording: obs in place of camtasia
L14 fetch files: wget, curl
L15 security: ufw, netstat, nmap
```

To print the last line

`sed -n -e '$p' sample.txt`

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -n -e '$p' sample.txt
L15 security: ufw, netstat, nmap
```

To print all lines except the last line

`sed -n -e '$!p' sample.txt`

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -n -e '$!p' sample.txt
L1 Linux is a free, opensource, secure and reliable os
L2 text editing: vim or emacs in place of notepad
L3 text processing: latex, libreoffice write in place of microsoft word
L4 spreadsheet: libreoffice calc in place of microsoft excel
L5 slides: libreoffice impress in place of microsoft powerpoint
L6 image processing: gimp, inkscape, imagej in place of adobe photoshop
L7 browsing: mozilla firefox in place of microsoft edge
L8 email: mozilla thunderbird in place of microsoft outlook
L9 symbolic math: sympy, sagemath in place of maple, mathematica
L10 array math: octave, numpy scilab in place of matlab
L11 literature: jabref, calibre in place of mendeley
L12 video editing: kdenlive in place of camtasia
L13 screen recording: obs in place of camtasia
L14 fetch files: wget, curl
```

Print line #5 to line #8

`sed -n -e '5,8p' sample.txt`

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -n -e '5,8p' sample.txt
L5 slides: libreoffice impress in place of microsoft powerpoint
L6 image processing: gimp, inkscape, imagej in place of adobe photoshop
L7 browsing: mozilla firefox in place of microsoft edge
L8 email: mozilla thunderbird in place of microsoft outlook
```

Now, combine two commands

`sed -n -e '=; 5,8p' sample.txt`

Print the line numbers, and print the lines from 5 to 8

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -n -e '=; 5,8p' sample.txt
1
2
3
4
5
L5 slides: libreoffice impress in place of microsoft powerpoint
6
L6 image processing: gimp, inkscape, imagej in place of adobe photoshop
7
L7 browsing: mozilla firefox in place of microsoft edge
8
L8 email: mozilla thunderbird in place of microsoft outlook
9
10
11
12
13
14
15
```

If we don't want those blank line numbers to show up, we write the following command

```
sed -n -e '5,8{=;p}' sample.txt
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -n -e '5,8{=;p}' sample.txt
5
L5 slides: libreoffice impress in place of microsoft powerpoint
6
L6 image processing: gimp, inkscape, imagej in place of adobe photoshop
7
L7 browsing: mozilla firefox in place of microsoft edge
8
L8 email: mozilla thunderbird in place of microsoft outlook
```

Print every odd line (or starting from line #1, print every 2nd line)

```
sed -n -e '1~2p' sample.txt
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -n -e '1~2p' sample.txt
L1 Linux is a free, opensource, secure and reliable os
L3 text processing: latex, libreoffice write in place of microsoft word
L5 slides: libreoffice impress in place of microsoft powerpoint
L7 browsing: mozilla firefox in place of microsoft edge
L9 symbolic math: sympy, sagemath in place of maple, mathematica
L11 literature: jabref, calibre in place of mendeley
L13 screen recording: obs in place of camtasia
L15 security: ufw, netstat, nmap
```

Similarly, print every 3rd line starting from line #1

```
sed -n -e '1~3p' sample.txt
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -n -e '1~3p' sample.txt
L1 Linux is a free, opensource, secure and reliable os
L4 spreadsheet: libreoffice calc in place of microsoft excel
L7 browsing: mozilla firefox in place of microsoft edge
L10 array math: octave, numpy scilab in place of matlab
L13 screen recording: obs in place of camtasia
```

Negate the above command

```
sed -n -e '1~3!p' sample.txt
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -n -e '1~3!p' sample.txt
L2 text editing: vim or emacs in place of notepad
L3 text processing: latex, libreoffice write in place of microsoft word
L5 slides: libreoffice impress in place of microsoft powerpoint
L6 image processing: gimp, inkscape, imagej in place of adobe photoshop
L8 email: mozilla thunderbird in place of microsoft outlook
L9 symbolic math: sympy, sagemath in place of maple, mathematica
L11 literature: jabref, calibre in place of mendeley
L12 video editing: kdenlive in place of camtasia
L14 fetch files: wget, curl
L15 security: ufw, netstat, nmap
```

Print the lines that contain the following regex pattern (here, in this case, it is a phrase)

```
sed -n -e '/microsoft/p' sample.txt
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -n -e '/microsoft/p' sample.txt
L3 text processing: latex, libreoffice write in place of microsoft word
L4 spreadsheet: libreoffice calc in place of microsoft excel
L5 slides: libreoffice impress in place of microsoft powerpoint
L7 browsing: mozilla firefox in place of microsoft edge
L8 email: mozilla thunderbird in place of microsoft outlook
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -n -e '/in place of/p' sample.txt
L2 text editing: vim or emacs in place of notepad
L3 text processing: latex, libreoffice write in place of microsoft word
L4 spreadsheet: libreoffice calc in place of microsoft excel
L5 slides: libreoffice impress in place of microsoft powerpoint
L6 image processing: gimp, inkscape, imagej in place of adobe photoshop
L7 browsing: mozilla firefox in place of microsoft edge
L8 email: mozilla thunderbird in place of microsoft outlook
L9 symbolic math: sympy, sagemath in place of maple, mathematica
L10 array math: octave, numpy scilab in place of matlab
L11 literature: jabref, calibre in place of mendeley
L12 video editing: kdenlive in place of camtasia
L13 screen recording: obs in place of camtasia
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -n -e '/in place of/!p' sample.txt
L1 Linux is a free, opensource, secure and reliable os
L14 fetch files: wget, curl
L15 security: ufw, netstat, nmap
```

Print n number of lines after the matching regex pattern

```
sed -n -e '/adobe/,+2p' sample.txt
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -n -e '/adobe/,+2p' sample.txt
L6 image processing: gimp, inkscape, imagej in place of adobe photoshop
L7 browsing: mozilla firefox in place of microsoft edge
L8 email: mozilla thunderbird in place of microsoft outlook
```

Delete a line (line #5 in this case) and print the remaining lines

```
sed -e '5d' sample.txt
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -e '5d' sample.txt
L1 Linux is a free, opensource, secure and reliable os
L2 text editing: vim or emacs in place of notepad
L3 text processing: latex, libreoffice write in place of microsoft word
L4 spreadsheet: libreoffice calc in place of microsoft excel
L6 image processing: gimp, inkscape, imagej in place of adobe photoshop
L7 browsing: mozilla firefox in place of microsoft edge
L8 email: mozilla thunderbird in place of microsoft outlook
L9 symbolic math: sympy, sagemath in place of maple, mathematica
L10 array math: octave, numpy scilab in place of matlab
L11 literature: jabref, calibre in place of mendeley
L12 video editing: kdenlive in place of camtasia
L13 screen recording: obs in place of camtasia
L14 fetch files: wget, curl
L15 security: ufw, netstat, nmap
```

Delete a range of lines (and print the remaining lines, *duh*)

```
sed -e '5,8d' sample.txt
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -e '5,8d' sample.txt
L1 Linux is a free, opensource, secure and reliable os
L2 text editing: vim or emacs in place of notepad
L3 text processing: latex, libreoffice write in place of microsoft word
L4 spreadsheet: libreoffice calc in place of microsoft excel
L9 symbolic math: sympy, sagemath in place of maple, mathematica
L10 array math: octave, numpy scilab in place of matlab
L11 literature: jabref, calibre in place of mendeley
L12 video editing: kdenlive in place of camtasia
L13 screen recording: obs in place of camtasia
L14 fetch files: wget, curl
L15 security: ufw, netstat, nmap
```

Delete the first line

```
sed -e '1d' sample.txt
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -e '1d' sample.txt
L2 text editing: vim or emacs in place of notepad
L3 text processing: latex, libreoffice write in place of microsoft word
L4 spreadsheet: libreoffice calc in place of microsoft excel
L5 slides: libreoffice impress in place of microsoft powerpoint
L6 image processing: gimp, inkscape, imagej in place of adobe photoshop
L7 browsing: mozilla firefox in place of microsoft edge
L8 email: mozilla thunderbird in place of microsoft outlook
L9 symbolic math: sympy, sagemath in place of maple, mathematica
L10 array math: octave, numpy scilab in place of matlab
L11 literature: jabref, calibre in place of mendeley
L12 video editing: kdenlive in place of camtasia
L13 screen recording: obs in place of camtasia
L14 fetch files: wget, curl
L15 security: ufw, netstat, nmap
```

Delete the last line

```
sed -e '$d' sample.txt
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -e '$d' sample.txt
L1 Linux is a free, opensource, secure and reliable os
L2 text editing: vim or emacs in place of notepad
L3 text processing: latex, libreoffice write in place of microsoft word
L4 spreadsheet: libreoffice calc in place of microsoft excel
L5 slides: libreoffice impress in place of microsoft powerpoint
L6 image processing: gimp, inkscape, imagej in place of adobe photoshop
L7 browsing: mozilla firefox in place of microsoft edge
L8 email: mozilla thunderbird in place of microsoft outlook
L9 symbolic math: sympy, sagemath in place of maple, mathematica
L10 array math: octave, numpy scilab in place of matlab
L11 literature: jabref, calibre in place of mendeley
L12 video editing: kdenlive in place of camtasia
L13 screen recording: obs in place of camtasia
L14 fetch files: wget, curl
```

Delete the lines with the following regex pattern

```
sed -e '/microsoft/d' sample.txt
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -e '/microsoft/d' sample.txt
L1 Linux is a free, opensource, secure and reliable os
L2 text editing: vim or emacs in place of notepad
L6 image processing: gimp, inkscape, imagej in place of adobe photoshop
L9 symbolic math: sympy, sagemath in place of maple, mathematica
L10 array math: octave, numpy scilab in place of matlab
L11 literature: jabref, calibre in place of mendeley
L12 video editing: kdenlive in place of camtasia
L13 screen recording: obs in place of camtasia
L14 fetch files: wget, curl
L15 security: ufw, netstat, nmap
```

Search and replace the first regex with the second regex pattern on all lines

```
sed -e 's/microsoft/MICROSOFT/g' sample.txt
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -e 's/microsoft/MICROSOFT/g' sample.txt
L1 Linux is a free, opensource, secure and reliable os
L2 text editing: vim or emacs in place of notepad
L3 text processing: latex, libreoffice write in place of MICROSOFT word
L4 spreadsheet: libreoffice calc in place of MICROSOFT excel
L5 slides: libreoffice impress in place of MICROSOFT powerpoint
L6 image processing: gimp, inkscape, imagej in place of adobe photoshop
L7 browsing: mozilla firefox in place of MICROSOFT edge
L8 email: mozilla thunderbird in place of MICROSOFT outlook
L9 symbolic math: sympy, sagemath in place of maple, mathematica
L10 array math: octave, numpy scilab in place of matlab
L11 literature: jabref, calibre in place of mendeley
L12 video editing: kdenlive in place of camtasia
L13 screen recording: obs in place of camtasia
L14 fetch files: wget, curl
L15 security: ufw, netstat, nmap
```

Search and replace the first regex with the second one on line #1

```
sed -e '1s/linux/LINUX/g' sample.txt
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -e '1s/linux/LINUX/g' sample.txt
L1 LINUX is a free, opensource, secure and reliable os
L2 text editing: vim or emacs in place of notepad
L3 text processing: latex, libreoffice write in place of microsoft word
L4 spreadsheet: libreoffice calc in place of microsoft excel
L5 slides: libreoffice impress in place of microsoft powerpoint
L6 image processing: gimp, inkscape, imagej in place of adobe photoshop
L7 browsing: mozilla firefox in place of microsoft edge
L8 email: mozilla thunderbird in place of microsoft outlook
L9 symbolic math: sympy, sagemath in place of maple, mathematica
L10 array math: octave, numpy scilab in place of matlab
L11 literature: jabref, calibre in place of mendeley
L12 video editing: kdenlive in place of camtasia
L13 screen recording: obs in place of camtasia
L14 fetch files: wget, curl
L15 security: ufw, netstat, nmap
```

`sed -e '1,$s/in place of/in lieu of/g' sample.txt`

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -e '1,$s/in place of/in lieu of/g' sample.txt
L1 linux is a free, opensource, secure and reliable os
L2 text editing: vim or emacs in lieu of notepad
L3 text processing: latex, libreoffice write in lieu of microsoft word
L4 spreadsheet: libreoffice calc in lieu of microsoft excel
L5 slides: libreoffice impress in lieu of microsoft powerpoint
L6 image processing: gimp, inkscape, imagej in lieu of adobe photoshop
L7 browsing: mozilla firefox in lieu of microsoft edge
L8 email: mozilla thunderbird in lieu of microsoft outlook
L9 symbolic math: sympy, sagemath in lieu of maple, mathematica
L10 array math: octave, numpy scilab in lieu of matlab
L11 literature: jabref, calibre in lieu of mendeley
L12 video editing: kdenlive in lieu of camtasia
L13 screen recording: obs in lieu of camtasia
L14 fetch files: wget, curl
L15 security: ufw, netstat, nmap
```

Print the lines without the first few characters at the beginning of each line

(Here, we are removing it from line #3 to line #6)

`sed -E -e '3,6s/^L[[:digit:]]+ //g' sample.txt`

Takes in an optional flag `-E` to let `sed` know that we're using the extended regex engine

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -E -e '3,6s/^L[[:digit:]]+ //g' sample.txt
L1 linux is a free, opensource, secure and reliable os
L2 text editing: vim or emacs in place of notepad
text processing: latex, libreoffice write in place of microsoft word
spreadsheet: libreoffice calc in place of microsoft excel
slides: libreoffice impress in place of microsoft powerpoint
image processing: gimp, inkscape, imagej in place of adobe photoshop
L7 browsing: mozilla firefox in place of microsoft edge
L8 email: mozilla thunderbird in place of microsoft outlook
L9 symbolic math: sympy, sagemath in place of maple, mathematica
L10 array math: octave, numpy scilab in place of matlab
L11 literature: jabref, calibre in place of mendeley
L12 video editing: kdenlive in place of camtasia
L13 screen recording: obs in place of camtasia
L14 fetch files: wget, curl
L15 security: ufw, netstat, nmap
```

To apply the above command to all the lines

```
sed -E -e 's/^L[[:digit:]]+ //g' sample.txt
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -E -e 's/^L[[:digit:]]+ //g' sample.txt
linux is a free, opensource, secure and reliable os
text editing: vim or emacs in place of notepad
text processing: latex, libreoffice write in place of microsoft word
spreadsheet: libreoffice calc in place of microsoft excel
slides: libreoffice impress in place of microsoft powerpoint
image processing: gimp, inkscape, imagej in place of adobe photoshop
browsing: mozilla firefox in place of microsoft edge
email: mozilla thunderbird in place of microsoft outlook
symbolic math: sympy, sagemath in place of maple, mathematica
array math: octave, numpy scilab in place of matlab
literature: jabref, calibre in place of mendeley
video editing: kdenlive in place of camtasia
screen recording: obs in place of camtasia
fetch files: wget, curl
security: ufw, netstat, nmap
```

Remove line numbers from line #3 to the line till it encounters a regex pattern

```
sed -E -e '3,/symbolic/s/^L[[:digit:]]+ //g' sample.txt
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -E -e '3,/symbolic/s/^L[[:digit:]]+ //g' sample.txt
L1 linux is a free, opensource, secure and reliable os
L2 text editing: vim or emacs in place of notepad
text processing: latex, libreoffice write in place of microsoft word
spreadsheet: libreoffice calc in place of microsoft excel
slides: libreoffice impress in place of microsoft powerpoint
image processing: gimp, inkscape, imagej in place of adobe photoshop
browsing: mozilla firefox in place of microsoft edge
email: mozilla thunderbird in place of microsoft outlook
symbolic math: sympy, sagemath in place of maple, mathematica
L10 array math: octave, numpy scilab in place of matlab
L11 literature: jabref, calibre in place of mendeley
L12 video editing: kdenlive in place of camtasia
L13 screen recording: obs in place of camtasia
L14 fetch files: wget, curl
L15 security: ufw, netstat, nmap
```

Remove the line numbers starting from line #1 and do it every 3rd line

`sed -E -e '1~3s/^L[[:digit:]]+ //g' sample.txt`

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -E -e '1~3s/^L[[:digit:]]+ //g' sample.txt
linux is a free, opensource, secure and reliable os
L2 text editing: vim or emacs in place of notepad
L3 text processing: latex, libreoffice write in place of microsoft word
spreadsheet: libreoffice calc in place of microsoft excel
L5 slides: libreoffice impress in place of microsoft powerpoint
L6 image processing: gimp, inkscape, imagej in place of adobe photoshop
browsing: mozilla firefox in place of microsoft edge
L8 email: mozilla thunderbird in place of microsoft outlook
L9 symbolic math: sympy, sagemath in place of maple, mathematica
array math: octave, numpy scilab in place of matlab
L11 literature: jabref, calibre in place of mendeley
L12 video editing: kdenlive in place of camtasia
screen recording: obs in place of camtasia
L14 fetch files: wget, curl
L15 security: ufw, netstat, nmap
```

Similarly, do the opposite of remove line numbers starting from line #1 and doing it every 2nd line

`sed -E -e '1~2!s/^L[[:digit:]]+ //g' sample.txt`

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -E -e '1~2!s/^L[[:digit:]]+ //g' sample.txt
L1 linux is a free, opensource, secure and reliable os
text editing: vim or emacs in place of notepad
L3 text processing: latex, libreoffice write in place of microsoft word
spreadsheet: libreoffice calc in place of microsoft excel
L5 slides: libreoffice impress in place of microsoft powerpoint
image processing: gimp, inkscape, imagej in place of adobe photoshop
L7 browsing: mozilla firefox in place of microsoft edge
email: mozilla thunderbird in place of microsoft outlook
L9 symbolic math: sympy, sagemath in place of maple, mathematica
array math: octave, numpy scilab in place of matlab
L11 literature: jabref, calibre in place of mendeley
video editing: kdenlive in place of camtasia
L13 screen recording: obs in place of camtasia
fetch files: wget, curl
L15 security: ufw, netstat, nmap
```

Remove the line numbers starting from a line having a regex pattern to a line having another regex pattern

```
sed -E -e '/text/,/video/s/^L[[:digit:]]+ //g' sample.txt
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -E -e '/text/,/video/s/^L[[:digit:]]+ //g' sample.txt
L1 linux is a free, opensource, secure and reliable os
text editing: vim or emacs in place of notepad
text processing: latex, libreoffice write in place of microsoft word
spreadsheet: libreoffice calc in place of microsoft excel
slides: libreoffice impress in place of microsoft powerpoint
image processing: gimp, inkscape, imagej in place of adobe photoshop
browsing: mozilla firefox in place of microsoft edge
email: mozilla thunderbird in place of microsoft outlook
symbolic math: sympy, sagemath in place of maple, mathematica
array math: octave, numpy scilab in place of matlab
literature: jabref, calibre in place of mendeley
video editing: kdenlive in place of camtasia
L13 screen recording: obs in place of camtasia
L14 fetch files: wget, curl
L15 security: ufw, netstat, nmap
```

Insert a line at line #1

Syntax within single quotes: `<position>i <text-to-insert>`

```
sed -e '1i --------------- header ----------------' sample.txt
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -e '1i ---------------- header ----------------' sample.txt
---------------- header ----------------
L1 linux is a free, opensource, secure and reliable os
L2 text editing: vim or emacs in place of notepad
L3 text processing: latex, libreoffice write in place of microsoft word
L4 spreadsheet: libreoffice calc in place of microsoft excel
L5 slides: libreoffice impress in place of microsoft powerpoint
L6 image processing: gimp, inkscape, imagej in place of adobe photoshop
L7 browsing: mozilla firefox in place of microsoft edge
L8 email: mozilla thunderbird in place of microsoft outlook
L9 symbolic math: sympy, sagemath in place of maple, mathematica
L10 array math: octave, numpy scilab in place of matlab
L11 literature: jabref, calibre in place of mendeley
L12 video editing: kdenlive in place of camtasia
L13 screen recording: obs in place of camtasia
L14 fetch files: wget, curl
L15 security: ufw, netstat, nmap
```

Similarly, to insert header and append footer in a single line ...

Syntax to append a footer: `$a <text-to-append>`

```
sed -e '1i --------------- header ----------------' -e '$a ---------------- footer --------------' sample.txt
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -e '1i ---------------- header ----------------' -e '$a ---------------- footer ----------------' sample.txt
---------------- header ----------------
L1 linux is a free, opensource, secure and reliable os
L2 text editing: vim or emacs in place of notepad
L3 text processing: latex, libreoffice write in place of microsoft word
L4 spreadsheet: libreoffice calc in place of microsoft excel
L5 slides: libreoffice impress in place of microsoft powerpoint
L6 image processing: gimp, inkscape, imagej in place of adobe photoshop
L7 browsing: mozilla firefox in place of microsoft edge
L8 email: mozilla thunderbird in place of microsoft outlook
L9 symbolic math: sympy, sagemath in place of maple, mathematica
L10 array math: octave, numpy scilab in place of matlab
L11 literature: jabref, calibre in place of mendeley
L12 video editing: kdenlive in place of camtasia
L13 screen recording: obs in place of camtasia
L14 fetch files: wget, curl
L15 security: ufw, netstat, nmap
---------------- footer ----------------
```

So, it seems like that `i` inserts before the mentioned line and `a` appends after the line

Insert a line before line #4 and append a line after line #8 in the same syntax

```
sed -e '4i -------- header --------' -e '8a -------- footer --------' sample.txt
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -e '4i --------- header --------' -e '8a --------- footer --------' sample.txt
L1 linux is a free, opensource, secure and reliable os
L2 text editing: vim or emacs in place of notepad
L3 text processing: latex, libreoffice write in place of microsoft word
--------- header ---------
L4 spreadsheet: libreoffice calc in place of microsoft excel
L5 slides: libreoffice impress in place of microsoft powerpoint
L6 image processing: gimp, inkscape, imagej in place of adobe photoshop
L7 browsing: mozilla firefox in place of microsoft edge
L8 email: mozilla thunderbird in place of microsoft outlook
--------- footer ---------
L9 symbolic math: sympy, sagemath in place of maple, mathematica
L10 array math: octave, numpy scilab in place of matlab
L11 literature: jabref, calibre in place of mendeley
L12 video editing: kdenlive in place of camtasia
L13 screen recording: obs in place of camtasia
L14 fetch files: wget, curl
L15 security: ufw, netstat, nmap
```

Similarly, we can do this insert and append using regex patterns instead of line numbers

```
sed -e '/microsoft/i ---- WATCHOUT ----' -e '/in place of/a ---- ALTERNATIVE ----'
sample.txt
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -e '/microsoft/i ---- WATCHOUT ----' -e '/in place of/a ---- ALTERNATIVE ----' sample.txt
L1 linux is a free, opensource, secure and reliable os
L2 text editing: vim or emacs in place of notepad
---- ALTERNATIVE ----
---- WATCHOUT ----
L3 text processing: latex, libreoffice write in place of microsoft word
---- ALTERNATIVE ----
---- WATCHOUT ----
L4 spreadsheet: libreoffice calc in place of microsoft excel
---- ALTERNATIVE ----
---- WATCHOUT ----
L5 slides: libreoffice impress in place of microsoft powerpoint
---- ALTERNATIVE ----
L6 image processing: gimp, inkscape, imagej in place of adobe photoshop
---- ALTERNATIVE ----
---- WATCHOUT ----
L7 browsing: mozilla firefox in place of microsoft edge
---- ALTERNATIVE ----
---- WATCHOUT ----
L8 email: mozilla thunderbird in place of microsoft outlook
---- ALTERNATIVE ----
L9 symbolic math: sympy, sagemath in place of maple, mathematica
---- ALTERNATIVE ----
L10 array math: octave, numpy scilab in place of matlab
---- ALTERNATIVE ----
L11 literature: jabref, calibre in place of mendeley
---- ALTERNATIVE ----
L12 video editing: kdenlive in place of camtasia
---- ALTERNATIVE ----
L13 screen recording: obs in place of camtasia
---- ALTERNATIVE ----
L14 fetch files: wget, curl
L15 security: ufw, netstat, nmap
```

Insert a line of text starting at line #1 and every 5 lines

```
sed -e '1~5i ------ BREAK ------' sample.txt
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -e '1~5i ------ BREAK ------' sample.txt
------ BREAK ------
L1 linux is a free, opensource, secure and reliable os
L2 text editing: vim or emacs in place of notepad
L3 text processing: latex, libreoffice write in place of microsoft word
L4 spreadsheet: libreoffice calc in place of microsoft excel
L5 slides: libreoffice impress in place of microsoft powerpoint
------ BREAK ------
L6 image processing: gimp, inkscape, imagej in place of adobe photoshop
L7 browsing: mozilla firefox in place of microsoft edge
L8 email: mozilla thunderbird in place of microsoft outlook
L9 symbolic math: sympy, sagemath in place of maple, mathematica
L10 array math: octave, numpy scilab in place of matlab
------ BREAK ------
L11 literature: jabref, calibre in place of mendeley
L12 video editing: kdenlive in place of camtasia
L13 screen recording: obs in place of camtasia
L14 fetch files: wget, curl
L15 security: ufw, netstat, nmap
```

Replace any line with a matching regex pattern with another line

```
sed -e '/microsoft/c ------ CENSORED ------' sample.txt
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -e '/microsoft/c ------ CENSORED ------' sample.txt
L1 linux is a free, opensource, secure and reliable os
L2 text editing: vim or emacs in place of notepad
------ CENSORED ------
------ CENSORED ------
------ CENSORED ------
L6 image processing: gimp, inkscape, imagej in place of adobe photoshop
------ CENSORED ------
------ CENSORED ------
L9 symbolic math: sympy, sagemath in place of maple, mathematica
L10 array math: octave, numpy scilab in place of matlab
L11 literature: jabref, calibre in place of mendeley
L12 video editing: kdenlive in place of camtasia
L13 screen recording: obs in place of camtasia
L14 fetch files: wget, curl
L15 security: ufw, netstat, nmap
```

Replace line #1 and every 3rd line after that with another line

```
sed -e '1~3c ------ CENSORED ------' sample.txt
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -e '1~3c ------ CENSORED ------' sample.txt
------ CENSORED ------
L2 text editing: vim or emacs in place of notepad
L3 text processing: latex, libreoffice write in place of microsoft word
------ CENSORED ------
L5 slides: libreoffice impress in place of microsoft powerpoint
L6 image processing: gimp, inkscape, imagej in place of adobe photoshop
------ CENSORED ------
L8 email: mozilla thunderbird in place of microsoft outlook
L9 symbolic math: sympy, sagemath in place of maple, mathematica
------ CENSORED ------
L11 literature: jabref, calibre in place of mendeley
L12 video editing: kdenlive in place of camtasia
------ CENSORED ------
L14 fetch files: wget, curl
L15 security: ufw, netstat, nmap
```

`sed` commands from a file

`hf.sed`

```
#!/usr/bin/sed -f
1i ------ HEADER ------
$a ------ FOOTER ------
1,5s/in place of/in lieu of/g
6i --- SIMPLER STUFF FROM HERE ONWARDS ---
6,$s/in place of.*//g
```

To run the above sed file

```
sed -f hf.sed sample.txt
```

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -f hf.sed sample.txt
------ HEADER ------
L1 linux is a free, opensource, secure and reliable os
L2 text editing: vim or emacs in lieu of notepad
L3 text processing: latex, libreoffice write in lieu of microsoft word
L4 spreadsheet: libreoffice calc in lieu of microsoft excel
L5 slides: libreoffice impress in lieu of microsoft powerpoint
--- SIMPLER STUFF FROM HERE ONWARDS ---
L6 image processing: gimp, inkscape, imagej
L7 browsing: mozilla firefox
L8 email: mozilla thunderbird
L9 symbolic math: sympy, sagemath
L10 array math: octave, numpy scilab
L11 literature: jabref, calibre
L12 video editing: kdenlive
L13 screen recording: obs
L14 fetch files: wget, curl
L15 security: ufw, netstat, nmap
------ FOOTER ------
```

To clean up the input file we used previously in `awk`

`block-ex6.input`

```
This file contains comments and numbers
Lines containing roll number and the fee paid need to be picked up
mm22b901  21000
Sum of the total fee paid needs to be printed at the end
Also a list of students, who paid how much
mm22b902 21000
Input text may contain some commends in between
mm22b906      21800
mm22b903 21500 paid through DD
mm22b905   22000
updated as on Jan 26, 2022 by Phani
```

`sed` script

`clean.sed`

```
/[[:alpha:]]{2}[[:digit:]]{2}[[:alpha:]][[:digit:]]+/!d
s/[ ]+/ /g
s/ ([[:digit:]]+).*/ \1/g
```

After we run the `sed` script on `block-ex6.input`

`sed -E -f clean.sed block-ex6.input`

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -E -f clean.sed block-ex6.input
mm22b901 21000
mm22b902 21000
mm22b906 21800
mm22b903 21500
mm22b905 22000
```

To join lines using `sed`

Joining lines requires `sed` to read one more line into the buffer

And this is achieved in a limited number of commands

`sed` commands can be constructed as loops and can perform complicated actions

Input file: `sample-split.txt`

```
L1 linux is a \ free, opensource, secure and reliable os
L2 text editing: vim or emacs in place of notepad
L3 text processing: latex, libreoffice writer \
in place of microsoft word
L4 spreadsheet: libreoffice calc in place of microsoft excel
L5 slides: libreoffice impress in place\
of microsoft \
powerpoint
L6 image processing: gimp, inkscape, imagej in place of\
adobe photoshop
L7 browsing: mozilla firefox in place of microsoft edge
L8 email: mozilla thunderbird in place of microsoft outlook
L9 symbolic math: sympy, sagemath in place of maple,\
mathematica
L10 array math: octave, numpy scilab in place of matlab
L11 literature: jabref, calibre in place of mendeley
L12 video editing: kdenlive in place of camtasia
L13 screen recording: obs in place of camtasia
L14 fetch files: wget, curl
L15 security: ufw, netstat, nmap
```

Join the lines that are ending in a backslash

`join.sed`

```
#!/usr/bin/sed -f
:x /\\$/N
/\\/s/\\\n//g
/\\$/bx
```

This is a loop (*ayo wut*)

Every line of the above `sed` file will be executed for every single that is read into the buffer from the input file

colon `:` indicates a label, and it is ignored while execution, it is used for branching

First line → Whenever there is a `\` followed by the end of the line, it will read the next line in the buffer, it's basically like joining the line following the one whic is being processed

Second line → On the lines which have `\` if there's a `\n` character after the `\` it will be replaced with null, basically to remove the `\` character

Third line → Those lines which contain `\` at the end of the line, we branch to the first line and repeat the steps again
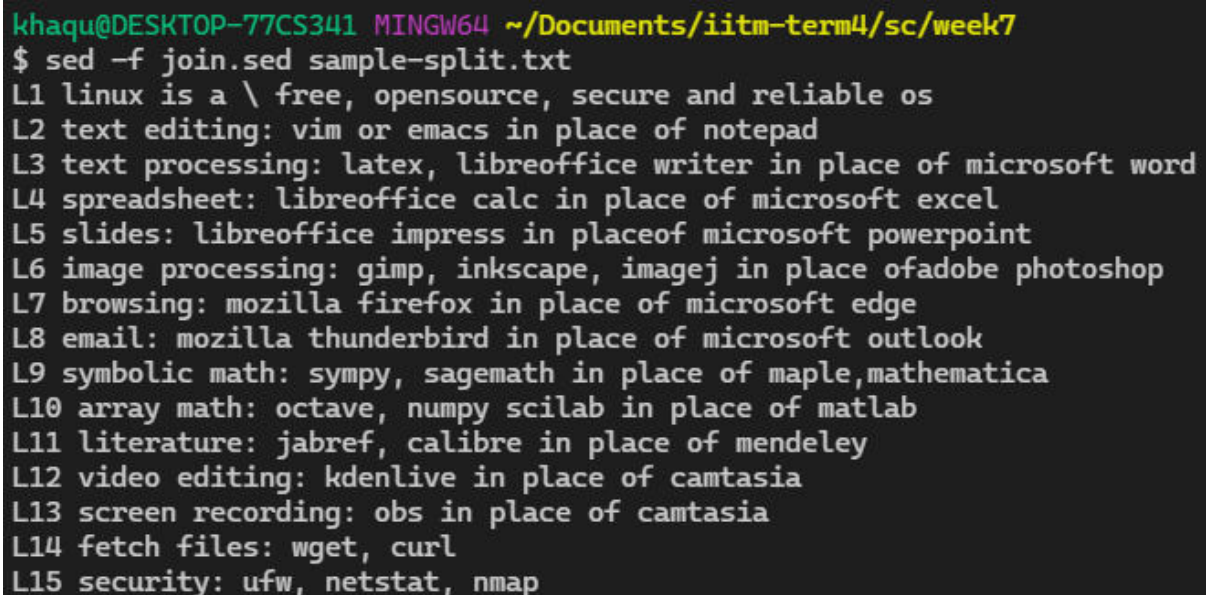
When do we not branch and go on to the next cycle? It is when the line doesn't contain `\` at the end of the line, which means that a recursive action is performed

This recursive loop can go infinite if not designed properly

To run the above `sed` file on `sample-split.txt`

`sed -f join.sed sample-split.txt`

```
khaqu@DESKTOP-77CS341 MINGW64 ~/Documents/iitm-term4/sc/week7
$ sed -f join.sed sample-split.txt
L1 linux is a \ free, opensource, secure and reliable os
L2 text editing: vim or emacs in place of notepad
L3 text processing: latex, libreoffice writer in place of microsoft word
L4 spreadsheet: libreoffice calc in place of microsoft excel
L5 slides: libreoffice impress in placeof microsoft powerpoint
L6 image processing: gimp, inkscape, imagej in place ofadobe photoshop
L7 browsing: mozilla firefox in place of microsoft edge
L8 email: mozilla thunderbird in place of microsoft outlook
L9 symbolic math: sympy, sagemath in place of maple,mathematica
L10 array math: octave, numpy scilab in place of matlab
L11 literature: jabref, calibre in place of mendeley
L12 video editing: kdenlive in place of camtasia
L13 screen recording: obs in place of camtasia
L14 fetch files: wget, curl
L15 security: ufw, netstat, nmap
```

There is an optional `--debug` flag that can be passed with the `sed` command and it will display a plethora of information

Command: `sed --debug -f join.sed sample-split.txt`

Output:

```
SED PROGRAM:
  :x
  /\\\\$/ N
  /\\\\/ s/\\\\\n//g
  /\\\\$/ b x
INPUT:   'sample-split.txt' line 1
PATTERN: L1 linux is a \\ free, opensource, secure and reliable os
COMMAND: :x
COMMAND: /\\\\$/ N
COMMAND: /\\\\/ s/\\\\\n//g
PATTERN: L1 linux is a \\ free, opensource, secure and reliable os
COMMAND: /\\\\$/ b x
END-OF-CYCLE:
L1 linux is a \ free, opensource, secure and reliable os
INPUT:   'sample-split.txt' line 2
PATTERN: L2 text editing: vim or emacs in place of notepad
COMMAND: :x
COMMAND: /\\\\$/ N
COMMAND: /\\\\/ s/\\\\\n//g
COMMAND: /\\\\$/ b x
END-OF-CYCLE:
L2 text editing: vim or emacs in place of notepad
INPUT:   'sample-split.txt' line 3
PATTERN: L3 text processing: latex, libreoffice writer \\
COMMAND: :x
COMMAND: /\\\\$/ N
PATTERN: L3 text processing: latex, libreoffice writer \\\nin place of microsoft word
COMMAND: /\\\\/ s/\\\\\n//g
MATCHED REGEX REGISTERS
  regex[0] = 46-48 '\
'
PATTERN: L3 text processing: latex, libreoffice writer in place of microsoft word
COMMAND: /\\\\$/ b x
END-OF-CYCLE:
L3 text processing: latex, libreoffice writer in place of microsoft word
INPUT:   'sample-split.txt' line 5
PATTERN: L4 spreadsheet: libreoffice calc in place of microsoft excel
COMMAND: :x
COMMAND: /\\\\$/ N
COMMAND: /\\\\/ s/\\\\\n//g
COMMAND: /\\\\$/ b x
END-OF-CYCLE:
L4 spreadsheet: libreoffice calc in place of microsoft excel
INPUT:   'sample-split.txt' line 6
PATTERN: L5 slides: libreoffice impress in place\\
COMMAND: :x
COMMAND: /\\\\$/ N
PATTERN: L5 slides: libreoffice impress in place\\\nof microsoft \\
COMMAND: /\\\\/ s/\\\\\n//g
MATCHED REGEX REGISTERS
  regex[0] = 39-41 '\
'
PATTERN: L5 slides: libreoffice impress in placeof microsoft \\
COMMAND: /\\\\$/ b x
COMMAND: :x
COMMAND: /\\\\$/ N
```

```
PATTERN: L5 slides: libreoffice impress in placeof microsoft \\\npowerpoint
COMMAND: /\\\\/ s/\\\\\n//g
MATCHED REGEX REGISTERS
  regex[0] = 52-54 '\
'
PATTERN: L5 slides: libreoffice impress in placeof microsoft powerpoint
COMMAND: /\\\\$/ b x
END-OF-CYCLE:
L5 slides: libreoffice impress in placeof microsoft powerpoint
INPUT:   'sample-split.txt' line 9
PATTERN: L6 image processing: gimp, inkscape, imagej in place of\\
COMMAND: :x
COMMAND: /\\\\$/ N
PATTERN: L6 image processing: gimp, inkscape, imagej in place of\\\nadobe photoshop
COMMAND: /\\\\/ s/\\\\\n//g
MATCHED REGEX REGISTERS
  regex[0] = 55-57 '\
'
PATTERN: L6 image processing: gimp, inkscape, imagej in place ofadobe photoshop
COMMAND: /\\\\$/ b x
END-OF-CYCLE:
L6 image processing: gimp, inkscape, imagej in place ofadobe photoshop
INPUT:   'sample-split.txt' line 11
PATTERN: L7 browsing: mozilla firefox in place of microsoft edge
COMMAND: :x
COMMAND: /\\\\$/ N
COMMAND: /\\\\/ s/\\\\\n//g
COMMAND: /\\\\$/ b x
END-OF-CYCLE:
L7 browsing: mozilla firefox in place of microsoft edge
INPUT:   'sample-split.txt' line 12
PATTERN: L8 email: mozilla thunderbird in place of microsoft outlook
COMMAND: :x
COMMAND: /\\\\$/ N
COMMAND: /\\\\/ s/\\\\\n//g
COMMAND: /\\\\$/ b x
END-OF-CYCLE:
L8 email: mozilla thunderbird in place of microsoft outlook
INPUT:   'sample-split.txt' line 13
PATTERN: L9 symbolic math: sympy, sagemath in place of maple,\\
COMMAND: :x
COMMAND: /\\\\$/ N
PATTERN: L9 symbolic math: sympy, sagemath in place of maple,\\\nmathematica
COMMAND: /\\\\/ s/\\\\\n//g
MATCHED REGEX REGISTERS
  regex[0] = 52-54 '\
'
PATTERN: L9 symbolic math: sympy, sagemath in place of maple,mathematica
COMMAND: /\\\\$/ b x
END-OF-CYCLE:
L9 symbolic math: sympy, sagemath in place of maple,mathematica
INPUT:   'sample-split.txt' line 15
PATTERN: L10 array math: octave, numpy scilab in place of matlab
COMMAND: :x
COMMAND: /\\\\$/ N
COMMAND: /\\\\/ s/\\\\\n//g
COMMAND: /\\\\$/ b x
END-OF-CYCLE:
```

```
L10 array math: octave, numpy scilab in place of matlab
INPUT:   'sample-split.txt' line 16
PATTERN: L11 literature: jabref, calibre in place of mendeley
COMMAND: :x
COMMAND: /\\\\$/ N
COMMAND: /\\\\/ s/\\\\\n//g
COMMAND: /\\\\$/ b x
END-OF-CYCLE:
L11 literature: jabref, calibre in place of mendeley
INPUT:   'sample-split.txt' line 17
PATTERN: L12 video editing: kdenlive in place of camtasia
COMMAND: :x
COMMAND: /\\\\$/ N
COMMAND: /\\\\/ s/\\\\\n//g
COMMAND: /\\\\$/ b x
END-OF-CYCLE:
L12 video editing: kdenlive in place of camtasia
INPUT:   'sample-split.txt' line 18
PATTERN: L13 screen recording: obs in place of camtasia
COMMAND: :x
COMMAND: /\\\\$/ N
COMMAND: /\\\\/ s/\\\\\n//g
COMMAND: /\\\\$/ b x
END-OF-CYCLE:
L13 screen recording: obs in place of camtasia
INPUT:   'sample-split.txt' line 19
PATTERN: L14 fetch files: wget, curl
COMMAND: :x
COMMAND: /\\\\$/ N
COMMAND: /\\\\/ s/\\\\\n//g
COMMAND: /\\\\$/ b x
END-OF-CYCLE:
L14 fetch files: wget, curl
INPUT:   'sample-split.txt' line 20
PATTERN: L15 security: ufw, netstat, nmap
COMMAND: :x
COMMAND: /\\\\$/ N
COMMAND: /\\\\/ s/\\\\\n//g
COMMAND: /\\\\$/ b x
END-OF-CYCLE:
L15 security: ufw, netstat, nmap
```

Version Control

⊙ Type	📙 Lecture
🗓 Date	@February 21, 2022
☰ Lecture #	2
⟳ Lecture URL	https://youtu.be/tzYx6DKtvGw
⟳ Notion URL	https://21f1003586.notion.site/Version-Control-bff372d37e544b94a72544dafd13d3c7
# Week #	7

Version Control

A group of programmers are working on a major project → tons of code in various files (modular approach)

Each programmer has multiple versions of each file they worked on

So, the basic idea of version control → trace back to a working version of code

Some version control systems are ...

SVN → Centrally hosted & managed version system

It has one master that keeps track of what version of the code that we are officially supporting

Then the master is going to ensure that the same version is replicated by others

Git → Distributed version control system

RAID

Redundant Array of Inexpensive/Independent Disks

It is a data storaga virtualization technology that combines multiple physical disk drive components into one or more logical units for the purposes of data redundancy, performance improvement or both

Since it is an array, it not only provides redundacy, but also speeds up the data access speed

As the speed limit of SATA interface is 6Gb/s, and we have 3 disks in an RAID array

When we access a file that is stored in 3 parts in those 3 disks, we can effectively utilize upto 18Gb/s of bandwidth

Workstations usually come with a RAID controller (it is a PCI card) and it determines how fast we can access data and the capacity of disks too

Hot Spare → It is a disk that we keep plugged in but it is not in use, but if any of the disks were to fail, the hot spare is automatically put to use

In a RAID config, **usable space < actual space**

Suppose, we have 4 x 8TB disks, so the actual space if 32TB, but the usable space might be 26TB

Git

- Distributed
- If the "master" server is hacked
  - Not much of an issue as every user (collaborator) has a copy of everything
- It doesn't require a server, but we can have a server
- **remote** → It is a server with which we sync our code, usually considered as the master version
- **Protocol used** → `git`
- Options to run git →

- locally run git server

- campus git server

- gitlab

- github

Git Cheat Sheet: https://education.github.com/git-cheat-sheet-education.pdf