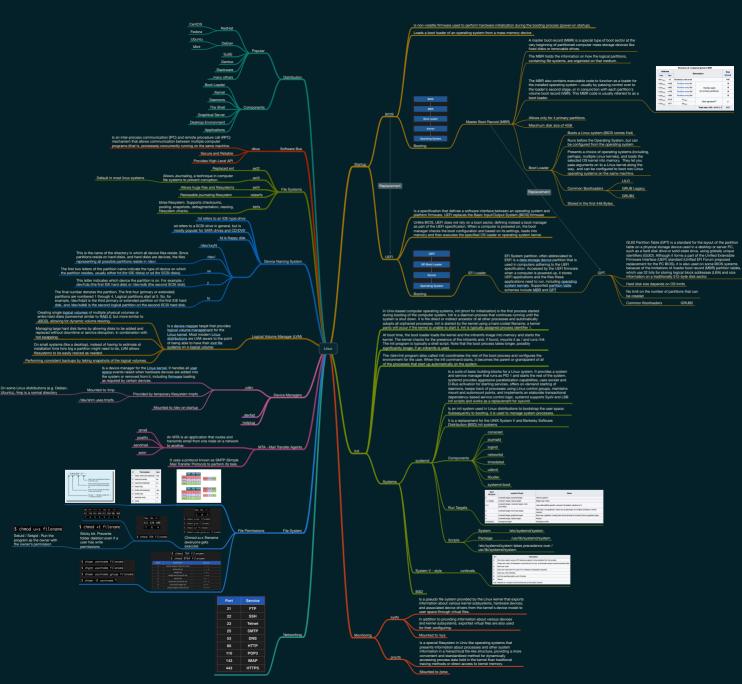
	root directory.
	Only root user has write privilege under this
Directory that contains temporary files created	alrectory.
by system and users. 8. /tmp – Temporary Files	Please note that /root is root user's home directory, which is not same as /.
Files under this directory are deleted when system is rebooted.	Contains binary executables.
Contains binaries, libraries, documentation, and source-code for second level programs.	Common linux commands you need to use in single-user modes are located under this
/usr/bin contains binary files for user programs. If you can't find a user binary under /bin, look under /usr/bin. For example: at, awk, cc, less, scp	2. /bin – User Binaries directory. Commands used by all the users of the system are located here.
/usr/sbin contains binary files for system	For example: ps, ls, ping, grep, cp.
administrators. If you can't find a system binary under /sbin, look under /usr/sbin. For example: atd, cron, sshd, useradd, userdel	Just like /bin, /sbin also contains binary executables.
/usr/lib contains libraries for /usr/bin and /usr/ sbin	But, the linux commands located under this directory are used typically by system administrator, for system maintenance
/usr/local contains users programs that you install from source. For example, when you install apache from source, it goes under /usr/ local/apache2	purpose. For example: iptables, reboot, fdisk, ifconfig, swapon
Home directories for all users to store their personal files.	Contains configuration files required by all programs.
For example: /home/john, /home/nikita	4. /etc – Configuration Files This also contains startup and shutdown shell scripts used to start/stop individual programs.
Contains boot loader related files.	For example: /etc/resolv.conf, /etc/
Kernel initrd, vmlinux, grub files are located under /boot – Boot Loader Files	logrotate.conf Contains device files.
For example: initrd.img-2.6.32-24-generic, vmlinuz-2.6.32-24-generic	These include terminal devices, usb, or any device attached to the system.
Contains library files that supports the binaries located under /bin and /sbin	Common Directories For example: /dev/tty1, /dev/usbmon0
Library filenames are either Id* or lib*.so.*	Contains information about system process.
For example: Id-2.11.1.so, libncurses.so.5.7 opt stands for optional. Contains add-on applications from individual	This is a pseudo filesystem contains information about running process. For example: /proc/[pid] directory contains information about the process with that particular pid.
vendors. 13. /opt – Optional add-on Applications add-on applications should be installed under either /opt/ or /opt/ sub-directory.	This is a virtual filesystem with text information about system resources. For example: /proc/uptime
Temporary mount directory where sysadmins 14. /mnt - Mount Directory can mount filesystems.	Is an interface to the kernel. Specifically, it provides a fliesystem-like view of information and configuration settings that the kernel
Temporary mount directory for removable devices.	provides, much like /proc.
For examples, /media/cdrom for CD-ROM; / media – Removable Media Devices media/floppy for floppy drives; /media/ cdrecorder for CD writer	Writing to these files may or may not write to the actual device, depending on the setting you're changing. It isn't only for managing devices, though that's a common use case.
srv stands for service.	var stands for variable files.
Contains server specific services related data. 16. /srv – Service Data	Content of the files that are expected to grow
For example, /srv/cvs contains CVS related	can be found under this directory. 7. /var – Variable Files This includes — system log files //var/log):
data.	7. /var – Variable Files This includes — system log files (/var/log); packages and database files (/var/lbq); packages and database files (/var/lbq); print queues (/var/spool); lock files (/var/mail); temp files needed across reboots (/var/tmp);
	everano),

Every single file and directory starts from the



		Р).
		Buffers The vi editor has a total of 27 buffers: 26 named buffers (a-z) and 1 unnamed buffer that is overwritten by each new operation.
Delete character to the right of cursor x		Named buffers
Delete character to the left of cursor X		The Unnamed buffer
Delete to the end of the line D	Deleting	
Delete current line dd		Just about any keystroke or action can be done X number of times by prefixing it with a
Delete current line :d		number. Force Multipliers
Yank the current line yy		For example, to move the cursor to line 5, you would press 5G. Moving 12 words to the right
Yank the current line :y	Yanking	is accomplished with 12W.
"/home/rbrunson/file1" 57L, 1756C 18,1 Top		vi filename
The numbers 18,1 on the right side of the message line are the current		Opening vi +/string filename open a file with a search string
line and column numbers, and the Top text is the current position of the cursor. This changes to be Bot if you entered the last half of the file. The other value possible is All, which simply means that all the contents of the file are currently on the screen.	Message line	vi +18 /etc/inittab edit a file and have the cursor jump to a particular line Vi has two modes insertion mode and command mode. The editor
Exit, saving changes :x	'	begins in command mode, where the cursor movement and text deletion and pasting occur. Insertion mode begins upon entering
Exit as long as there have been no changes :q		Modes an insertion or change command. [ESC] returns the editor to
Exit and save changes if any have been made ZZ	Quitting	command mode (where you can quit, for example by typing :q!). Most commands execute as soon as you type them except for
Exit and ignore any changes :q!		"colon" commands which execute when you press the ruturn key.
Move left h		You can only get to last line mode from command mode, and you get into last line
Move down j	$\overline{}$	mode by pressing the colon key
Move up k	<u> </u>	LastLine mode You can do simple things, like quitting your vi
		session You can also perform come amoring the course
		You can also perform some amazing vi search commands or vim search and replace
Move to next word w Move to next blank delimited word W		commands
		i Insert before cursor
Move to the beginning of the word b Move to the beginning of blank delimted word B		I Insert before line
		a Append after cursor
Move to the end of the word e		Inserting Text A Append after line
Move to the end of Blank delimited word E		o Open a new line after current line
Move a sentence back		O Open a new line before current line
Move a sentence forward		r Replace one character
Move a paragraph back {	Motion	R Replace many characters
Move a paragraph forward }		You press U in Command mode to undo a single operation or the latest in a series of
Move to the begining of the line 0		Lindo Operations changes. If you opened a file, made 30
Move to the end of the line \$		changes, and then pressed the U key 30 times, you'd end up with the exact same file
Move to the first line of the file 10		you had opened.
Move to the last line of the file G		/string Search forward for string
Move to nth line of the file no	G. C.	?string Search back for string
Move to nth line of the file :n		n Search for next instance of string
Move forward to c fc	—	N Search for previous instance of string
Move back to c Fo		:s/pattern/string/flags Replace pattern with string according to flag
Move to top of screen H		g Flag - Replace all occurences of pattern
Move to middle of screen M		c Flag - Confirm replaces.
Move to botton of screen L	_	& Repeat last :s command
Move to associated (), {}, [] %		:w file Write to file
		r file Read file in after line
		:n Go to next file
		:p Go to previos file
		:e file Edit file

You have seen that while you are editing, your last deletion (d or x) or yank (y) is saved in a buffer (a place in stored memory). You can access the contents of that buffer and put the saved text back in your file with the put command (p or

Replace line with output from program

!!program

					. (dot)	Any single character except newline
Matches if the line contains the value Hello	/Hello/					zero or more occurances of any character
Matches if the line contains TEST by itself	/^TEST\$/				[]	Any single character specified in the set
Matches if the line starts with any letter	/^[a-zA-Z]/				[^]	Any single character not specified in the set
Matches if the first character of the line is a-z and there is at least one more of any	/^[a-z].*/				^	Anchor - beginning of the line
character following it					\$	Anchor - end of line
Matches if line ends with 2134	/2134\$/	Examples	Regex(7)	Rules	\<	Anchor - beginning of word
Matches is the line contains 21 or 35					b	Anchor - end of word
Note the use of () with the pipe symbol to specify the 'or' condition	/\(21 35\)/				\(\)	Grouping - usually used to group conditions
Matches if there are zero or more numbers in	//0.01*/			<u> </u>	\n	Contents of nth grouping
the line	/[0-9]*/				+	One or more occurrences of the preceding element
Matches if the first character is not a # in the line	/^[^#]/					Can mean the start and end of a regular
						expression