Linux Project
Basic Command and Directory Hierarchy
1. Content

- Directory Hierarchy
- Basic Commands
- Intermediate Commands
- Shell Basic Operators
- Special Characters
- Shell Script
- Environmental & Shell Variables

2. Directory Hierarchy

- Directory structure
- root directory /
- /bin
  Directory that stores essential commands
  e.g. cat, mv, rmdir
- home directory /home
  Saved user setting, saved files etc.
- Pathname: /home/“user name”
- Why directory is important?
  Directory is like a folder, things are stored in different directories

3. Basic Commands – Directory Operation

- How to know which directory you are in?
  pwd - path name, print working directory

- What are stored in the directory?
  ls - Listing files and directories, listing everything in the current directory

- How do I change my directory?
  cd - change directory e.g. cd desktop

- What if I want to make a new directory?
  mkdir - make directory

- What if I want to remove a directory?
  rmdir - remove directory
  Be aware that Linux will only allow you to delete a directory when it is empty!!

4. Basic Commands – File Operation

- How to move a file from one directory to another one?
  mv - move
  E.g. mv filename directory_name = move the file to the desired directory

- How do I copy a file?
  cp - copy
  cp file1 file2 = make a copy of file1 as file2 (REMARK: input the pathname if you are not in the same directory as the file)

- How do I remove a file?
  rm - remove
  rm filename = remove the file (REMARK: a prompt will appear to confirm the command, you need to type y for confirmation)

5. Basic Commands – Reading File

- There are several ways to read the content of a file

  cat - concatenate

  more - to read a long file that doesn't fit in one single window view
  type q to quit more
  use / to search word e.g. /science to find the word "science"

  head – default to print the first 10 lines, can change by options
  E.g. head -5 file = print the first 5 lines

  tail – default to print the last 10 lines , can change by options
  E.g. tail -5 file = print the last 5 lines

- But when to use which commends?
  Scenario 1: To read multiple files → cat (cat file1 file2)
  Scenario 2: To read a very long passage more
  Scenario 3: To read the first few lines to confirm the file content head
  Scenario 4: To read the last few lines in the file tail

- As different commends provide different output, it depends on what you need.
- What if the commend window become too messy?
  clear - clear screen, it will clear the previous lines

- Can I read the content more specifically? YES

  grep - print the lines that contain the specific word in specific file

E.g. grep science Science.txt

Options:

-i neglect upper or lower case

-v display line that doesn't match

-n precede each matching line with the line number

-c print only the total count of matched lines

- Sometimes, the number of lines is important.
- For the file on follows, the number of lines indicates the number of earthquakes.

wc - word count

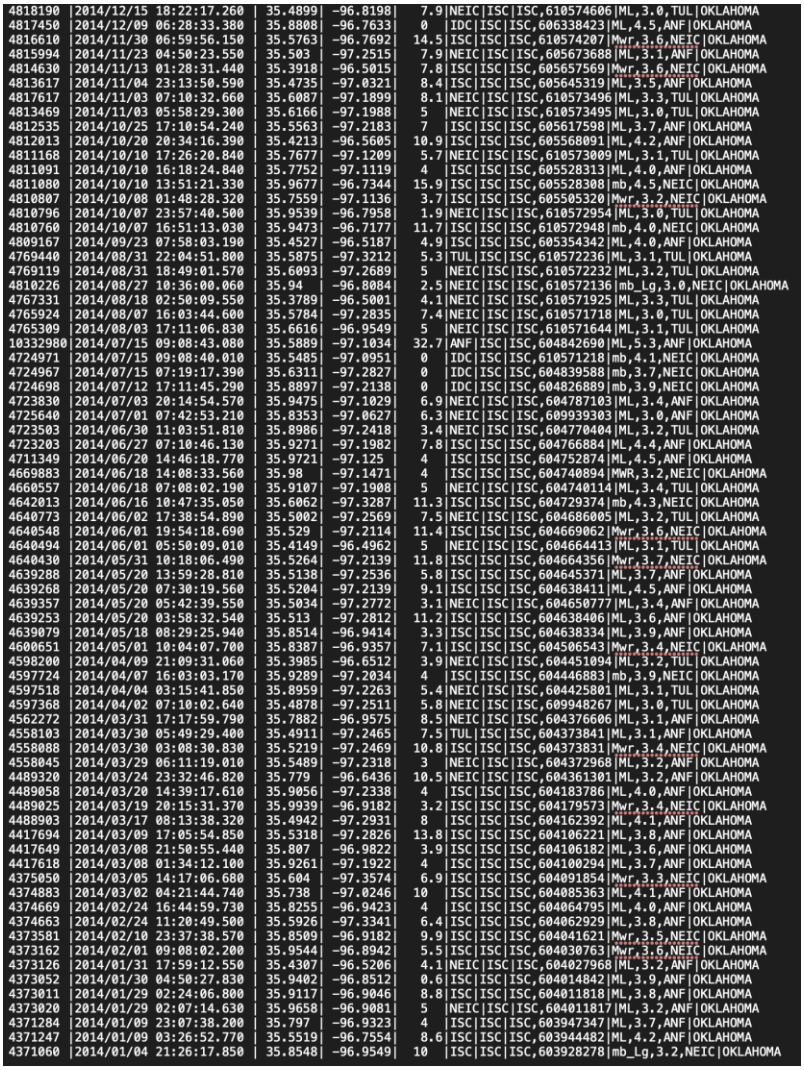
E.g. wc -w science.txt = no. of words in the file “science.txt”

Options:

-w word count

-l line count

-m character count



```
4818190 |2014/12/15 18:22:17.260 | 35.4899| -96.8198|   7.9|NEIC|ISC|ISC,610574606|ML,3.0,TUL|OKLAHOMA
4817450 |2014/12/09 06:28:33.380 | 35.8808| -96.7633|   0  |IDC|ISC|ISC,606338423|ML,4.5,ANF|OKLAHOMA
4816610 |2014/11/30 06:59:56.150 | 35.5763| -96.7692|  14.5|ISC|ISC|ISC,610574207|Mwr,3.6,NEIC|OKLAHOMA
4815994 |2014/11/23 04:50:23.550 | 35.503 | -97.2515|   7.9|NEIC|ISC|ISC,605673688|ML,3.1,ANF|OKLAHOMA
4814630 |2014/11/13 01:28:31.440 | 35.3918| -96.5015|   7.8|ISC|ISC|ISC,605657569|Mwr,3.6,NEIC|OKLAHOMA
4813617 |2014/11/04 23:13:50.590 | 35.4735| -97.0321|   8.4|ISC|ISC|ISC,605645319|ML,3.5,ANF|OKLAHOMA
4817617 |2014/11/03 07:10:32.660 | 35.6087| -97.1899|   8.1|NEIC|ISC|ISC,610573496|ML,3.3,TUL|OKLAHOMA
4813469 |2014/11/03 05:58:29.300 | 35.6166| -97.1988|   5  |NEIC|ISC|ISC,610573495|ML,3.0,TUL|OKLAHOMA
4812535 |2014/10/25 17:10:54.240 | 35.5563| -97.2183|   7  |ISC|ISC|ISC,605617598|ML,3.7,ANF|OKLAHOMA
4812013 |2014/10/20 20:34:16.390 | 35.4213| -96.5605|  10.9|ISC|ISC|ISC,605568091|ML,4.2,ANF|OKLAHOMA
4811168 |2014/10/10 17:26:20.840 | 35.7677| -97.1209|   5.7|NEIC|ISC|ISC,610573009|ML,3.1,TUL|OKLAHOMA
4811091 |2014/10/10 16:18:24.840 | 35.7752| -97.1119|   4  |ISC|ISC|ISC,605528313|ML,4.0,ANF|OKLAHOMA
4811080 |2014/10/10 13:51:21.330 | 35.9677| -96.7344|  15.9|ISC|ISC|ISC,605528308|mb,4.5,NEIC|OKLAHOMA
4810807 |2014/10/08 01:48:28.320 | 35.7559| -97.1136|   3.7|ISC|ISC|ISC,605505320|Mwr,3.2,NEIC|OKLAHOMA
4810796 |2014/10/07 23:57:40.500 | 35.9539| -96.7958|   1.9|NEIC|ISC|ISC,610572954|ML,3.0,TUL|OKLAHOMA
4810760 |2014/10/07 16:51:13.030 | 35.9473| -96.7177|  11.7|ISC|ISC|ISC,610572948|mb,4.0,NEIC|OKLAHOMA
4809167 |2014/09/23 07:58:03.190 | 35.4527| -96.5187|   4.9|ISC|ISC|ISC,605354342|ML,4.0,ANF|OKLAHOMA
4769440 |2014/08/31 22:04:51.800 | 35.5875| -97.3212|   5.3|TUL|ISC|ISC,610572236|ML,3.1,TUL|OKLAHOMA
4769119 |2014/08/31 18:49:01.570 | 35.6093| -97.2689|   5  |NEIC|ISC|ISC,610572232|ML,3.2,TUL|OKLAHOMA
4810226 |2014/08/27 10:36:00.060 | 35.94  | -96.8084|   2.5|NEIC|ISC|ISC,610572136|mb_Lg,3.0,NEIC|OKLAHOMA
4767331 |2014/08/18 02:50:09.550 | 35.3789| -96.5001|   4.1|NEIC|ISC|ISC,610571925|ML,3.3,TUL|OKLAHOMA
4765924 |2014/08/07 16:03:44.600 | 35.5784| -97.2835|   7.4|NEIC|ISC|ISC,610571718|ML,3.0,TUL|OKLAHOMA
4765309 |2014/08/03 17:11:06.830 | 35.6616| -96.9549|   5  |NEIC|ISC|ISC,610571644|ML,3.1,TUL|OKLAHOMA
10332980|2014/07/15 09:08:43.080 | 35.5889| -97.1034|  32.7|ANF|ISC|ISC,604842690|ML,5.3,ANF|OKLAHOMA
4724971 |2014/07/15 09:08:40.010 | 35.5485| -97.0951|   0  |IDC|ISC|ISC,610571218|mb,4.1,NEIC|OKLAHOMA
4724967 |2014/07/15 07:19:17.390 | 35.6311| -97.2827|   0  |IDC|ISC|ISC,604839588|mb,3.7,NEIC|OKLAHOMA
4724698 |2014/07/12 17:11:45.290 | 35.8897| -97.2138|   0  |IDC|ISC|ISC,604826889|mb,3.9,NEIC|OKLAHOMA
4723830 |2014/07/03 20:14:54.570 | 35.9475| -97.1029|   6.9|NEIC|ISC|ISC,604787103|ML,3.4,ANF|OKLAHOMA
4725640 |2014/07/01 07:42:53.210 | 35.8353| -97.0627|   6.3|NEIC|ISC|ISC,609939303|ML,3.0,ANF|OKLAHOMA
4723503 |2014/06/30 11:03:51.810 | 35.8986| -97.2418|   3.4|NEIC|ISC|ISC,604770404|ML,3.2,TUL|OKLAHOMA
4723203 |2014/06/27 07:10:46.130 | 35.9271| -97.1982|   7.8|ISC|ISC|ISC,604766884|ML,4.4,ANF|OKLAHOMA
4711349 |2014/06/20 14:46:18.770 | 35.9721| -97.125 |   4  |ISC|ISC|ISC,604752874|ML,4.5,ANF|OKLAHOMA
4669883 |2014/06/18 14:08:33.560 | 35.98  | -97.1471|   4  |ISC|ISC|ISC,604740894|MWR,3.2,NEIC|OKLAHOMA
4660557 |2014/06/18 07:08:02.190 | 35.9107| -97.1908|   5  |NEIC|ISC|ISC,604740114|ML,3.4,TUL|OKLAHOMA
4642013 |2014/06/16 10:47:35.050 | 35.6062| -97.3287|  11.3|ISC|ISC|ISC,604729374|mb,4.3,NEIC|OKLAHOMA
4640773 |2014/06/02 17:38:54.890 | 35.5002| -97.2569|   7.5|NEIC|ISC|ISC,604686005|ML,3.2,TUL|OKLAHOMA
4640548 |2014/06/01 19:54:18.690 | 35.529 | -97.2114|  11.4|ISC|ISC|ISC,604669062|Mwr,3.6,NEIC|OKLAHOMA
4640494 |2014/06/01 05:50:09.010 | 35.4149| -96.4962|   5  |NEIC|ISC|ISC,604664413|ML,3.1,TUL|OKLAHOMA
4640430 |2014/05/31 10:18:06.490 | 35.5264| -97.2139|  11.8|ISC|ISC|ISC,604664356|Mwr,3.7,NEIC|OKLAHOMA
4639288 |2014/05/20 13:59:28.810 | 35.5138| -97.2536|   5.8|ISC|ISC|ISC,604645371|ML,3.7,ANF|OKLAHOMA
4639268 |2014/05/20 07:30:19.560 | 35.5204| -97.2139|   9.1|ISC|ISC|ISC,604638411|ML,4.5,ANF|OKLAHOMA
4639357 |2014/05/20 05:42:39.550 | 35.5034| -97.2772|   3.1|NEIC|ISC|ISC,604650777|ML,3.4,ANF|OKLAHOMA
4639253 |2014/05/20 03:58:32.540 | 35.513 | -97.2812|  11.2|ISC|ISC|ISC,604638406|ML,3.6,ANF|OKLAHOMA
4639079 |2014/05/18 08:29:25.940 | 35.8514| -96.9414|   3.3|ISC|ISC|ISC,604638334|ML,3.9,ANF|OKLAHOMA
4600651 |2014/05/01 10:04:07.700 | 35.8387| -96.9357|   7.1|ISC|ISC|ISC,604506543|Mwr,3.4,NEIC|OKLAHOMA
4598200 |2014/04/09 21:09:31.060 | 35.3985| -96.6512|   3.9|NEIC|ISC|ISC,604451094|ML,3.2,TUL|OKLAHOMA
4597724 |2014/04/07 16:03:03.170 | 35.9289| -97.2034|   4  |ISC|ISC|ISC,604446883|mb,3.9,NEIC|OKLAHOMA
4597518 |2014/04/04 03:15:41.850 | 35.8959| -97.2263|   5.4|NEIC|ISC|ISC,604425801|ML,3.1,TUL|OKLAHOMA
4597368 |2014/04/02 07:10:02.640 | 35.4878| -97.2511|   5.8|NEIC|ISC|ISC,609948267|ML,3.0,TUL|OKLAHOMA
4562272 |2014/03/31 17:17:59.790 | 35.7882| -96.9575|   8.5|NEIC|ISC|ISC,604376606|ML,3.1,ANF|OKLAHOMA
4558103 |2014/03/30 05:49:29.400 | 35.4911| -97.2465|   7.5|TUL|ISC|ISC,604373841|ML,3.1,ANF|OKLAHOMA
4558088 |2014/03/30 03:08:30.830 | 35.5219| -97.2469|  10.8|ISC|ISC|ISC,604373831|Mwr,3.4,NEIC|OKLAHOMA
4558045 |2014/03/29 06:11:19.010 | 35.5489| -97.2318|   3  |NEIC|ISC|ISC,604372968|ML,3.2,ANF|OKLAHOMA
4489320 |2014/03/24 23:32:46.820 | 35.779 | -96.6436|  10.5|NEIC|ISC|ISC,604361301|ML,3.2,ANF|OKLAHOMA
4489058 |2014/03/20 14:39:17.610 | 35.9056| -97.2338|   4  |ISC|ISC|ISC,604183786|ML,4.0,ANF|OKLAHOMA
4489025 |2014/03/19 20:15:31.370 | 35.9939| -96.9182|   3.2|ISC|ISC|ISC,604179573|Mwr,3.4,NEIC|OKLAHOMA
4488903 |2014/03/17 08:13:38.320 | 35.4942| -97.2931|   8  |ISC|ISC|ISC,604162392|ML,4.1,ANF|OKLAHOMA
4417694 |2014/03/09 17:05:54.850 | 35.5318| -97.2826|  13.8|ISC|ISC|ISC,604106221|ML,3.8,ANF|OKLAHOMA
4417649 |2014/03/08 21:50:55.440 | 35.807 | -96.9822|   3.9|ISC|ISC|ISC,604106182|ML,3.6,ANF|OKLAHOMA
4417618 |2014/03/08 01:34:12.100 | 35.9261| -97.1922|   4  |ISC|ISC|ISC,604100294|ML,3.7,ANF|OKLAHOMA
4375050 |2014/03/05 14:17:06.680 | 35.604 | -97.3574|   6.9|ISC|ISC|ISC,604091854|Mwr,3.3,NEIC|OKLAHOMA
4374883 |2014/03/02 04:21:44.740 | 35.738 | -97.0246|  10  |ISC|ISC|ISC,604085363|ML,4.1,ANF|OKLAHOMA
4374669 |2014/02/24 16:44:59.730 | 35.8255| -96.9423|   4  |ISC|ISC|ISC,604064795|ML,4.0,ANF|OKLAHOMA
4374663 |2014/02/24 11:20:49.500 | 35.5926| -97.3341|   6.4|ISC|ISC|ISC,604062929|ML,3.8,ANF|OKLAHOMA
4373581 |2014/02/10 23:37:38.570 | 35.8509| -96.9182|   9.9|ISC|ISC|ISC,604041621|Mwr,3.5,NEIC|OKLAHOMA
4373162 |2014/02/01 09:08:02.200 | 35.9544| -96.8942|   5.5|ISC|ISC|ISC,604030763|Mwr,3.6,NEIC|OKLAHOMA
4373126 |2014/01/31 17:59:12.550 | 35.4307| -96.5206|   4.1|NEIC|ISC|ISC,604027968|ML,3.2,ANF|OKLAHOMA
4373052 |2014/01/30 04:50:27.830 | 35.9402| -96.8512|   0.6|ISC|ISC|ISC,604014842|ML,3.9,ANF|OKLAHOMA
4373011 |2014/01/29 02:24:06.800 | 35.9117| -96.9046|   8.8|ISC|ISC|ISC,604011818|ML,3.8,ANF|OKLAHOMA
4373020 |2014/01/29 02:07:14.630 | 35.9658| -96.9081|   5  |NEIC|ISC|ISC,604011817|ML,3.2,ANF|OKLAHOMA
4371284 |2014/01/09 23:07:38.200 | 35.797 | -96.9323|   4  |ISC|ISC|ISC,603947347|ML,3.7,ANF|OKLAHOMA
4371247 |2014/01/09 03:26:52.770 | 35.5519| -96.7554|   8.6|ISC|ISC|ISC,603944482|ML,4.2,ANF|OKLAHOMA
4371060 |2014/01/04 21:26:17.850 | 35.8548| -96.9549|  10  |ISC|ISC|ISC,603928278|mb_Lg,3.2,NEIC|OKLAHOMA
```

6. Basic Commands – Assists

- Sometimes you may forgot the use or options of a commend. For this, you may try these commends.

  man - manuals, give information of command
  e.g. man wc

  whatis - give short description of a command
  E.g. whatis wc

  apropos - give command with the desired word in their manual
  e.g. apropos keyword give a list of commands with the word "keyword" in their manual

7. Intermediate Commands

- Except the basic operations, the shell environment can do things more complicated than that. For example, more specific action on files?

  awk
  awk options pattern {action} variables filename
  without pattern: every line
  without action: print every line the match the pattern

8. Intermediate Commands – awk

- awk is a very convenient tool for data processing, it can manipulate the file based on certain pattern to do certain action.

- Generalised code: awk options 'pattern {action}' variables filename
  Pattern: where to do the action e.g. NR>1: line number > 1
  Without pattern: the action will be acted on every line
  Action: what to do to the data e.g. print $4: print column 4
  Without action: print every line that match the pattern

- awk example
  awk '/00/' file1 - print every line with 00 in the text within file1

  awk '$4==00' file1 - print every line with 00 in the fourth column. $ indicate the column, == means equal to.

  awk '{print $11}' file1 - print column 11 in file1

  awk '{print $8, $7}' file1 - print column 8 and 7 with a space to separate in file 1. " , " means space

  awk '{print $8,"E",$7,"N"}' file1 - words in "" will be print as letter.

  awk 'NR>1' file1 - print everything below first row in file 1.

awk 'NR>1{angle=$7+45; print angle}' file1 – try to interpret this

9. Shell Basic Operators

- These operators not only useful in awk, but also in loops
  - < less than
  - <= less than or equal to
  - == equal to
  - != not equal to
  - >= greater than or equal to
  - > greater than

10. Special Characters

- Other than the operators, there are characters carrying special meaning in shell

  > - Redirect output,
  cat > list1 : a single cat command will read input from user and > will direct all input to list1
  >> - append
  cat >> list1 : cat will again read the input and add to list1 without replacing the initial content
  sort – sorting
  sort list1 : make list1 in alphabetical order (REMARK: only sort according to the first word)
  | - pipes, pipes will take the result of first commands to run the second command
  e.g. cat file1 | wc –l
  cat file1 will read file1, then pipe the result to run wc –l to count the no. of lines

- Sometimes, we need to operate with different files with similar name and need to organise them. Wildcards in this case will be very helpful.

  Wildcards:
  * - matches one or more occurrences of any character, including no character.
  e.g. ls list* : it will show everything start with list
  e.g. ls *list will show everything end with list

  ? - represents or matches a single occurrence of any character
  e.g. ls list? will show everything with one letter after list
  e.g. ls list??? will show everything with three letters after list
  e.g. ls ???list show file end with "list" with three letters before it

11. Shell Script

- In shell environment, the every command is run right after our input.Although “;” can be used to continuously input different commends, it increase the possibility of error and the difficulty of improvement.Therefore, a script is used.

- A script can type a series of commands for the terminal to run
  - gedit - an text editor to write a script
    - gedit file1 & : open a “file1” in gedit, & let command run in background
    - Without &: cannot run additional command until exit gedit
    - #: to add comments to the code

12. How to run a script

- At the beginning of the code, you need to define the file is a script in what shell environment (csh, tcsh, bash etc.)
  - To define, add “#!/bin/tcsh” in the first line

- After finished the script, save the script and run these code
  - chmod - make the file executable
    - Code: chmod +x filename
  - To execute, type the name of the modded file

13. Environment & Shell Variable

- Sometimes, the data is repeating throughout the whole script, or the data is continuously changing in a loop. To simplify the work, variables are used. There are two types of variable:
  - Environment variable
    - System-wide available
    - i.e. Once set in environment, can be used in all shell
  - Shell variable
    - Only available in the shell

14. Shell variable – Example

- How to set variable?
  - set - set variable
    - e.g. set directory=~/groupwork/act1
    - $directory can be then used to replace the whole path

- Example: Get the number of earthquakes from 1971 to 2021 around the centre (35.599, -96.752) with radius be 50km.

```
#!/bin/tcsh
set yr = 1971
while ( $yr <= 2021)
        FetchEvent -s $yr-01-01 -e $yr-12-31 --mag 3:10 --radius 35.599:-96.752:0.5 >!
        EQ_$yr.txt
        if ($yr == 1971) then
        wc -l EQ_$yr.txt >! EQ_count.txt
        else
        wc -l EQ_$yr.txt >> EQ_count.txt
        endif
        @ yr++
end
```