|  |
| --- |
| **Lab 2:**  ***Task 1: To be familiar with linux basic command.***  ***Task 2 :To be familiar with vi editor***  ***Task 3 : To be familiar with execution of some program.*** |

***Introduction:***

***Linux basic command:***

1. ***ls - This command is used to list the files and directories within a directory.***

*$ ls*

*example\_directory file1.txt file2.txt*

1. ***pwd - This command is used to display the present working directory***

*$ pwd*

*/home/user/example\_directory*

1. *cd - This command is used to change the present working directory.*

*$ pwd*

*/home/user/example\_directory*

*$ cd /tmp*

*$ pwd*

*/tmp*

1. ***Date Command :***

*This command is used to display the current data and time.*

***Syntax*** *:*

*$date*

*$date +%ch*

***Options*** *: -*

*a = Abbrevated weekday.*

*A = Full weekday.*

*b = Abbrevated month.*

*B = Full month.*

*c = Current day and time.*

*C = Display the century as a decimal number.*

*d = Day of the month.*

*D = Day in „mm/dd/yy‟ format*

*h = Abbrevated month day.*

*H = Display the hour.*

*L = Day of the year.*

*m = Month of the year.*

*M = Minute.*

*P = Display AM or PM*

*S = Seconds*

*T = HH:MM:SS format*

*//u = Week of the year.*

*y = Display the year in 2 digit.*

*Y = Display the full year.*

*Z = Time zone .*

***To change the format :***

***Syntax*** *:*

*$date „+%H-%M-%S‟*

1. ***Calender Command :***

*This command is used to display the calendar of the year or the particular month of calendar year.*

***Syntax*** *:*

*a.$cal <year>*

*b.$cal <month> <year>*

*Here the first syntax gives the entire calendar for given year & the second Syntax gives the calendar of reserved month of that year.*

1. ***Echo Command :***

*This command is used to print the arguments on the screen .*

*Syntax : $echo <text>*

1. ***Multi line echo command :***

*To have the output in the same line , the following commands can be used.*

***Syntax*** *: $echo <text\>text*

*To have the output in different line, the following command can be used.*

***Syntax*** *: $echo “text*

*>line2*

*>line3”*

1. ***Banner Command :***

*It is used to display the arguments in „#‟ symbol .*

***Syntax*** *: $banner <arguments>*

***If not install then***

*Sudo apt-get install sysvbanner*

1. ***’who’ Command :***

*It is used to display who are the users connected to our computer currently.*

*Syntax : $who – option‟s*

***Options : -***

*H–Display the output with headers.*

*b–Display the last booting date or time or when the system was lastely rebooted.*

1. ***’who am i’ Command :***

*Display the details of the current working directory.*

*Syntax : $who am i*

1. ***’tty’ Command :***

*It will display the terminal name.*

***Syntax*** *: $tty*

1. ***’Binary’ Calculator Command :***

*It will change the „$‟ mode and in the new mode, arithmetic operations such as +,-,\*,/,%,n,sqrt(),length(),=, etc can be performed . This command is used to go to the binary calculus mode.*

***Syntax*** *:*

*$bc operations*

*^d*

*$*

*1 base –inputbase*

*0 base – outputbase are used for base conversions.*

*Base :*

*Decimal = 1 Binary = 2 Octal = 8 Hexa = 16*

1. ***’CLEAR’ Command :***

*It is used to clear the screen.*

***Syntax*** *: $clear*

1. ***’MAN’ Command :***

*It help us to know about the particular command and its options & working. It is like „help‟ command in windows .*

*Syntax : $man <command name>*

1. ***MANIPULATION Command :***

*It is used to manipulate the screen.*

***Syntax*** *: $tput <argument>*

*Arguments :*

*1.****Clear*** *– to clear the screen.*

*2.****Longname*** *– Display the complete name of the terminal.*

*3.****SMSO*** *– background become white and foreground become black color.*

*4.****rmso*** *– background become black and foreground becomes white color.*

*//5.****Cop R C*** *– Move to the cursor position to the specified location.*

*6.****Cols*** *– Display the number of columns in our terminals.*

1. ***LIST Command :***

*It is used to list all the contents in the current working directory.*

***Syntax*** *:*

***ls****$ ls – options <arguments>*

*If the command does not contain any argument means it is working in the Current directory.*

***Options*** *:*

*a– used to list all the files including the hidden files.*

*c– list all the files columnwise.*

*d- list all the directories.*

*m- list the files separated by commas.*

*p- list files include „/‟ to all the directories.*

*r- list the files in reverse alphabetical order.*

*f- list the files based on the list modification date.*

*x-list in column wise sorted order.*

***DIRECTORY RELATED COMMANDS :***

1. ***Present Working Directory Command :***

*To print the complete path of the current working directory.*

*Syntax : $pwd*

1. *MKDIR Command :*

*To create or make a new directory in a current directory .*

*Syntax : $mkdir <directory name>*

1. *CD Command :*

*To change or move the directory to the mentioned directory .*

*Syntax : $cd <directory name.*

1. *RMDIR Command :*

*To remove a directory in the current directory & not the current directory itself.*

*Syntax : $rmdir <directory name>*

|  |  |
| --- | --- |
| cd /home | enter to directory '/ home'   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=cd)] |
| # cd .. | go back one level   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=cd)] |
| # cd ../.. | go back two levels   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=cd)] |
| # cd | go to home directory   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=cd)] |
| # cd ~user1 | go to home directory   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=cd)] |
| # cd - | go to previous directory   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=cd)] |
| # cp file1 file2 | copying a file   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=cp)] |
| # cp dir/\* . | copy all files of a directory within the current work directory   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=cp)] |
| # cp -a /tmp/dir1 . | copy a directory within the current work directory   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=cp)] |
| # cp -a dir1 dir2 | copy a directory   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=cp)] |
| # cp file file1 | outputs the mime type of the file as text   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=file)] |
| # iconv –l | lists known encodings   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=iconv)] |
| # iconv -f fromEncoding -t toEncoding inputFile > outputFile | converting the coding of characters from one format to another   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=iconv)] |
| # find . -maxdepth 1 -name \*.jpg -print -exec convert | batch resize files in the current directory and send them to a thumbnails directory (requires convert from Imagemagick)   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=iconv)] |
| # ln -s file1 lnk1 | create a symbolic link to file or directory   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=ln)] |
| # ln file1 lnk1 | create a physical link to file or directory   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=ln)] |
| # ls | view files of directory   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=ls)] |
| # ls –F | view files of directory   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=ls)] |
| # ls –l | show details of files and directory   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=ls)] |
| # ls –a | show hidden files   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=ls)] |
| # ls \*[0-9]\* | show files and directory containing numbers   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=ls)] |
| # lstree | show files and directories in a tree starting from root(2)   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=lstree)] |
| # mkdir dir1 | create a directory called 'dir1'   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=mkdir)] |
| # mkdir dir1 dir2 | create two directories simultaneously   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=mkdir)] |
| # mkdir -p /tmp/dir1/dir2 | create a directory tree   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=mkdir)] |
| # mv dir1 new\_dir | rename / move a file or directory   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=mv)] |
| # pwd | show the path of work directory   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=pwd)] |
| # rm -f file1 | delete file called 'file1'   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=rm)] |
| # rm -rf dir1 | remove a directory called 'dir1' and contents recursively   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=rm)] |
| # rm -rf dir1 dir2 | remove two directories and their contents recursively   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=rm)] |
| # rmdir dir1 | delete directory called 'dir1'   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=rmdir)] |
| # touch -t 0712250000 file1 | modify timestamp of a file or directory - (YYMMDDhhmm)   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=touch)] |
| # tree | show files and directories in a tree starting from root(1)   [[man](http://www.linuxguide.it/command_line/linux-manpage/do.php?file=tree)] |

***FILE RELATED COMMANDS :***

1. ***CREATE A FILE :***

*To create a new file in the current directory we use CAT command.*

*This command is used to concatenate and display the contents of files.*

***Syntax :***

*$cat > filename.*

*The > symbol is redirectory we use cat command.*

*To come outside ctrl+d*

1. ***touch*** *- This command is used to create a new file or update the modification time of an existing file.*

*$ touch new\_file.txt*

*$ ls*

*example\_directory file1.txt file2.txt new\_file.txt*

1. ***DISPLAY A FILE :***

*To display the content of file mentioned we use CAT command without „>‟ operator.*

***Syntax*** *:*

*$ cat file1.txt*

*This is the contents of file1.txt*

*$ cat file2.txt*

*This is the contents of file2.txt*

*$cat <filename.*

*Options –s = to neglect the warning /error message.*

1. *cp - This command is used to copy files or directories.*

*$ ls*

*file1.txt file2.txt*

*$ cp file1.txt file1\_copy.txt*

*$ ls*

*file1.txt file1\_copy.txt file2.txt*

1. ***COPYING CONTENTS :***

*To copy the content of one file with another. If file doesnot exist, a new file is created and if the file exists with some data then it is overwritten.*

***Syntax :***

*$ cat <filename source> >> <destination filename>*

*$ cat <source filename> >> <destination filename> it is avoid overwriting.*

***Options*** *: -*

*-n content of file with numbers included with blank lines.*

*Syntax :*

*$cat –n <filename>*

1. ***SORTING A FILE :***

*To sort the contents in alphabetical order in reverse order.*

*Syntax :*

*$sort <filename >*

*Option : $ sort –r <filename>*

1. ***COPYING CONTENTS FROM ONE FILE TO ANOTHER :***

*To copy the contents from source to destination file . so that both contents are same.*

***Syntax*** *:*

*$cp <source filename> <destination filename>*

*$cp <source filename path > <destination filename path>*

1. ***MOVE Command :***

*To completely move the contents from source file to destination file and to remove the source file.*

***Syntax*** *:*

*$ mv <source filename> <destination filename>*

1. ***REMOVE Command :***

*To permanently remove the file we use this command .*

***Syntax*** *:*

*$rm <filename>*

1. ***WORD Command :***

*To list the content count of no of lines , words, characters .*

***Syntax*** *:*

*$wc<filename>*

***Options*** *:*

*-c – to display no of characters.*

*-l – to display only the lines.*

*-w – to display the no of words.*

1. ***LINE PRINTER :***

*To print the line through the printer, we use lp command.*

*Syntax :*

*$lp <filename>*

1. ***PAGE Command :***

*This command is used to display the contents of the file page wise & next page can be viewed by pressing the enter key.*

*Syntax :*

*$pg <filename>*

1. ***FILTERS AND PIPES***

***HEAD*** *: It is used to display the top ten lines of file.*

*Syntax: $head<filename>*

***TAIL*** *: This command is used to display the last ten lines of file.*

*Syntax: $tail<filename>*

***PAGE*** *: This command shows the page by page a screenfull of information is displayed after which the page command displays a prompt and passes for the user to strike the enter key to continue scrolling.*

*Syntax: $ls –a\p*

***MORE*** *: It also displays the file page by page .To continue scrolling with more command , press the space bar key.*

*Syntax: $more<filename>*

***GREP*** *:This command is used to search and print the specified patterns from the file. Syntax: $grep [option] pattern <filename>*

***SORT*** *: This command is used to sort the datas in some order.*

*Syntax: $sort<filename>*

***PIPE*** *: It is a mechanism by which the output of one command can be channeled into the input of another command.*

*Syntax: $who | wc-l*

*TR :The tr filter is used to translate one set of characters from the standard inputs to another.*

*Syntax: $tr “[a-z]” “[A-Z]”*

***COMMUNICATION THROUGH UNIX COMMANDS***

1. *MESG*

*Description: The message command is used to give permission to other users to send message to your terminal.*

*Syntax: $mesg y*

1. *Command: WRITE*

*Description: This command is used to communicate with other users, who are logged in at the same time.*

*Syntax: $write <user name>*

1. *Command: WALL*

*Description: This command sends message to all users those who are logged in using the unix server.*

*Syntax: $wall <message>*

1. *Command: MAIL*

*Description: It refers to textual information, that can be transferred from one user to another*

*Syntax: $mail <user name>*

1. *Command: REPLY*

*Description: It is used to send reply to specified user.*

*Syntax: $reply<user name>*

**Aim: To be familiar with VI-editor**

1. *Using the Visual editor:*

*An editor is the program that is used to edit source code.There are many text editors available for linux ,but the two most widely used are visual editor improved(VIM) and Emacs.Here we discuss VIM editor.*

1. *Creating and opening file:*

* *$vi <filename> it opens file it it is already exist,otherwise creates new file.*
* *$vi it opens VIM editors ,without file name.*

1. *VIM modes*

*Vim has many modes,but the two most important are the command mode(normal mode), and the insert mode.In command mode VIM interprets everything typed as a command to VIM. To insert the text the VIM editor must be in insert mode.To handle the file such as saving,quiting from file,VIM must be in command mode.*

*Initially VIM will be in command mode.*

* *Adding text to the file*

*There are two basic commands to inter to the insert mode.*

*a opens insert mode and enable to insert text after the cursor.*

*i opens insert mode and enables to inter text before the cursor.*

*- to return to the normal mode press Esc.*

* *Insert command*

*a -Append text after the cursor and enter insert mode*

*A -append text at the end of line and enter insert mode*

*i -Insert text before the curser and enter insert mode*

*I -Insert text at the beginning of the line and enter insert mode.*

*o -Open a line below the curser and enter the insert mode.*

*O -Open a line above the cursor and enter insert mode.*

* *Moving around in a file(cursor movement commands)*

*h -Move left*

*l -Move right*

*k -move up*

*j -move down*

*7h -Move 7 character to the left*

*17j -Move 17 lines down*

*4k -moves 4 lines up.*

1. *-go to the first character in the line*

*$ -Go to the last character in the line*

*G -Go to the last line in the file.*

*nG -GO to the line n in the file.*

*w -Move to the beginning of the nest word*

*5e -Move to the end of 5th word from the current position*

*Ctrl+F -scroll forward(down).*

*Ctrl+B -Scroll backward (up).*

* *Dleting text*

*For this Editor must be in Normal mode to delete text using commands.*

*x -Delete one character*

*dw -deletes word by word*

*d$ -deletes from cursor to the end of the line*

*dd -deletes the entire line*

*nx -delete n character ,beginning with the character at the cursor.*

*n+delete key -same as nx*

*ndw -delete n number of words*

*ndd -delete n number of lines*

* *Undo/redo*

*u -undo the last edit.*

*U -undoes the al edits in the current line*

*Ctrl+r -redo*

* *Replace the text mode*

*New text is inserted into the old text into the cursor*

*r -replace the character under the cursor*

*nr -replace the n number of characters*

*cw -replace the rest of the word*

*c$ -replace the rest of the line*

*R -enter Replace mode to make unlimited overwrites(Press Esc to go back to the normal mode)*

* *Change delete (cut)and copy(yank)*

*Change delete copy*

*Line cc dd yy*

*Character ch dl yl*

*Word cw dw yw*

*Paragraph above c{ d{ y{*

*Paragraph below c{ d{ y{*

*Sentence back c( d( y(*

*Sentence forward c( d( y(*

* *Paste*

*p -paste after the cursor*

*P -paste before the cursor*

* *Finding the text*

*Normal mode operation:*

*/searchtext -search the string from the cursor to the end of the file*

*?searchtext -search the string from the cursor to the beginning of the file*

*n -continue search in the same direction*

*N -continue search in opposite direction*

* *Save changes and Exit the visual editor*

*Normal mode operation:*

*:w -write (save ) the file*

*:q -quit if saved*

*:wq -save and quit or :x or ZZ*

*:wq filename -save with filename.*

*:q! -quit without saveing any changes*

*:e! Abandon the changes and reload the last saved file*

* *Getting help*

*:help -opens the help window*

*:help<command> -open help window for the command*

*:ctrl+j -jump to the help topic*

*:q -quit from help window.*

* *Compiling with GCC*

*A compiler turns human readable code into machine readable object code that can actually run.The compilers of choice on linux system are all part of the GNU compiler collection.Usually knoen as gcc. GCC supports the ANSI standard syntax for C.*

1. *Compile ,link and run the program*

*$gcc –o<exefile><sourcefile>*

*$./<exefile>*

1. *Example:*

*$gcc –o hello hello.c*

*$./hello*

1. *Note:*

*If you compile without <exefile>gcc saves the executable file as “a.out” file*

*$./a.out -to execute.*

1. *Compiling a single source file*

*$ gcc –c<sourcefile>*

*Example:*

*$gcc –c hello.c The resulting file is named hello.o*

*Overall syntax*

*$vi*

*#include<stdio.h>*

*………..*

*………*

*}*

*:wq name*

*$gcc –o name.c name*

*$./name.o*

*Do the following task*

1. *Practice each and every command mentioned above*
2. *Write c source file for execute “hello world “ and save it as file xyz.c in your directory*
3. *Write the c source file for calculating interest and save it as file name calc.c in your directory*

/\* 1. Program to print the process id \*/

#include<stdio.h>

main()

{i

nt pid,ppid;

pid=getpid();

ppid=getppid();

printf("\n Process Id is %d\n",pid);

printf("\n Parent Process Id is %d\n",ppid);

}

OUTPUT

process ID is 5198

parent process ID is 5129

|  |
| --- |
| **Questions related to exam:**   1. **List out any ten command used in linux and write their output.** 2. **What is VI editor? How you can write a program in VI editor and execute them?** |