

Aim : Study of Basic Linux Commands : ls, echo, read, cat, touch, test, loops, arithmetic comparison, conditional loops, grep, sed etc.

Theory :

1. echo :- echo command in linux is used to display line of text/string that are passed as an argument. This is a built in command that is mostly used in shell scripts and batch files to output status text to the screen or a file.

Syntax : \$ echo [option] [string] or  
echo [string]

Example : \$ echo " Operating System "

2. ls :- The ls command is one of the most commonly used commands in daily linux/unix operations. That lists directory contents of files and directories.

Example : \$ ls / applications will display all the files and folders stored in the applications folder.

3. read :- read command in linux system is used to read from a file descriptor. This command read up the total number of bytes from the specified file descriptor into the buffer.

Zeal Education Institutes



Syntax : `$ read`.

Example : `read` command without any option : The `read` command without any asks for the user's input and exit once the user provides some input.

4. `cat` :- `cat` (concatenate) command is very frequently used in Linux.

It reads data from the file and gives their content as output.

~~It can be used to cut parts of a line by byte position, character and field.~~

It helps us to create, view, concatenate files.

Syntax :- `$ cat filename`.

It will show content of given filename.

5. `touch` :- It is used to create the file with content.

Syntax :- `$ touch file_name`.

Example :- `touch file1`.

6. `test` :- A `test` command is used to test the validity of command.

It checks whether the command/expression is true or false.

Syntax :- `$ test [expression]`

Example :- `test "variable1" operator "variable2"`.

7. `loops` :- The `for` loop using in and list of values. This `for` loop contains a number of variables in the list and will execute for each item in the list.





**syntax :-** For varname in list  
do  
    echo "statement"  
done.

8. **Pwd :-** This command displays the current working directory of the terminal.

**syntax :-** \$ pwd

9. **cd :-** change to a directory.  
This command is used to change the current working directory of the user.

**Syntax :-** \$ cd / pathname/

10. **mkdir :-** This command is used to create a new directory.

**syntax :-** \$ mkdir directory-name.

11. **sort :-** This command sort the results of a search either alphabetically or numeric files, file contents and directories can be sorted using this command.

12. **cp :-** This commands copies files and directories.

A copy of the files / directory copies remains in the working directory.

**Syntax :-** \$ cp { filename }

Zeal Education Institutes



13. `mv :-` This command moves files and directory from one directory to another. The directory once moved is deleted from the working directory.

syntax :- `$ mv {Filename} | pathname`.

14. `grep :-` This command is used to search a particular string | word in a text file. This is similar to "`ctrl + F`", executed via a CLI.

syntax :- `$ grep <flag or element> _to_search> {Filename}`.

15. `sed :-` sed command in linux/unix stands for stream editor and it can perform lots of functions on file like searching, find and replace, insertion or deletion.

syntax :- `$ sed options ... [script] [INPUTFILE...]`

Example :- Consider the below text file as an input.  
`$ cat > geekfile.txt`.

- Replacing or substituting string :- sed command is mostly used to replace the text in a file.
- The below simple sed command replaces the word "unix" with "linux" in the file.

ex. :- `$ sed 's/unix/linux/' geekfile.txt`.



Aim : Write a program to implement an address book with options given below :-

- a) Create address book
- b) View address book
- c) Insert a record
- d) Delete a record
- e) Modify a record
- f) Exit

Theory :

Data : data is collection of facts, such as numbers, words, measurements, observations, or just descriptions of things.

Database : A large amount of data that is stored in a computer and can easily be used, added, stored and accessed electronically.

Commands -

1. CREATE Command :- This command is used to create database.

Syntax :- CREATE DATABASE dbname ;

2. SHOW :- This command is used to show all available databases.

Syntax :- SHOW DATABASES ;

Zeal Education Institutes

Zeal Education Institutes



3. **CREATE TABLE :-** This statement is used to create a new table in database.

Syntax :-  

```
CREATE TABLE table_name(  
    column1 datatype,  
    column2 datatype,  
    column3 datatype,  
    ....  
);
```

4. **Insert INTO :-** This statement is used to insert new records in a table

Syntax :-  

```
INSERT INTO table_name (column1,  
    column2, column3, ...)  
VALUES (value1, value2, value3, ...);
```

5. **DELETE :-** This statement is used to delete existing records in table.

Syntax :-  

```
DELETE FROM table_name WHERE  
condition ;
```

6. **UPDATE :-** The update statement is used to modify the existing records in a table.

Syntax :-  

```
UPDATE table_name SET column1 = value1, ...  
WHERE condition ;
```





Aim : Process control system calls : The demonstration of FORK, EXECVE and WAIT system calls along with zombie and orphan states.

Implement the C program in which main program accepts the integers to be sorted. Main program uses the FORK system call to create a new process called a child process. Parent process sorts the integers using sorting algorithm and wait for child process using WAIT system call to sort the integers using any sorting algorithm. Also demonstrate zombie and orphan states.

Theory :

Fork () System Call —

1. Fork() system call creates the processes. It return process id without taking any arguments. Fork() system call creates the new process.
2. This newly created process then is the child process of the caller.
3. Both parent and child executes the next instruction following Fork system call.
4. The return of negative value by Fork() indicates creation of child is unsuccessful.



5. Fork() returns 0 for child process which is newly created. It then returns process id of child to its parent which is an integer.

6. The getpid() function is used to retrieve process id of the process.

wait() System Call —

1. The wait() system call suspend execution of the current process until one of its children terminates.

2. The waitpid() system call suspends execution of the current process until a child specified by pid argument has changed state.

execve() System Call —

1. execve() executes the program pointed by filename. filename must be either a binary executable, or a script starting with a line of the form "#!  
interpreter [arg]".

2. execve() does not return on success, and the text, data, bss, and stack of the calling process are overwritten by that of the program loaded.



### Zombie Process :-

1. zombie process or defunct process is a process that has completely execution but still has an entry in the process table.
2. It is a process in the "terminated state".
3. This process occurs for child processes, where the entry is still needed to allow the parent process to read its child's exist status.
4. Once the exit status is read via the wait system call, the zombie's entry is removed from the process table and it is said to be "reaped".
5. A child process always first becomes a zombie before being removed from the resource table.

### Orphan Process :-

1. An orphan process is a computer process whose parent process has finished or terminated, through it remains running itself.
2. Orphan process is the process for which parent is dead. Suppose a program forks a new child process and while



the child is performing some work, Its parent dies. This child process then becomes orphan.

3. Orphan process doesn't remain orphan for a long time because after its parent death, the child process is adopted by the init process which we already know is the parent of all the processes.
4. When the child process is adopted by the init process then its ppid value becomes 1, which is the pid for init process.

Conclusion :-

Hence, we studied to implement C program using `fork()` system call and demonstrate zombie and orphan state.