

OPERATING SYSTEMS PROJECT REPORT

File Management System



Jaypee Institute of Information Technology

Submitted by

Ananya Kapoor	20103104	B4
Dhairya Sachdeva	20103098	B4

Submitted To

Dr. Ashish Mishra

DECLARATION

We hereby declare that the project titled “*File Management System*” submitted for the degree of Bachelors in Technology is a bonafide record submitted to Jaypee Institute of Information Technology, under the guidance of Dr. Ashish Mishra, has been carried out by our own efforts and is a record of our original work.

BACKGROUND:

File management is one of the basic and important features of operating system. Operating system is used to manage files of computer system. All the files with different extensions are managed by operating system.

A file is collection of specific information stored in the memory of computer system. File management is defined as the process of manipulating files in computer system, its management includes the process of creating, modifying and deleting the files.

Our files have several common characteristics built in. Each file is made up of data, but also, metadata is embedded into the file to help the operating system (OS) manage how the file works and how it is stored. Metadata records file information such as the author, file creation date, modified date, and file size.

File management is a process of maintaining any kind of records in a proper manner like your work document or your money records this is the process to divide things in different stages and in writing from so that in future when needed it will be easy to get that particular record.

INTRODUCTION:

Computer users spend time every day interacting with digital files and folders, including creating, downloading, naming, moving, saving, copying, reviewing, navigating, searching for, sharing, and deleting them. This activity, called file management (FM).

File management is an art of storing, naming, sorting and handling documents files in a systematic manner. So that in future it will easy to retrieve data.

A file management system is a type of software that manages data files in a computer system. It has limited capabilities and is designed to manage individual or group files, such as special office documents and records.

The following are some of the tasks performed by file management of operating system of any computer system:

1. It helps to create new files in computer system and placing them at the specific locations.
2. It helps in easily and quickly locating these files in computer system.

3. It helps to stores the files in separate folders known as directories. These directories help users to search file quickly or to manage the files according to their types or uses.

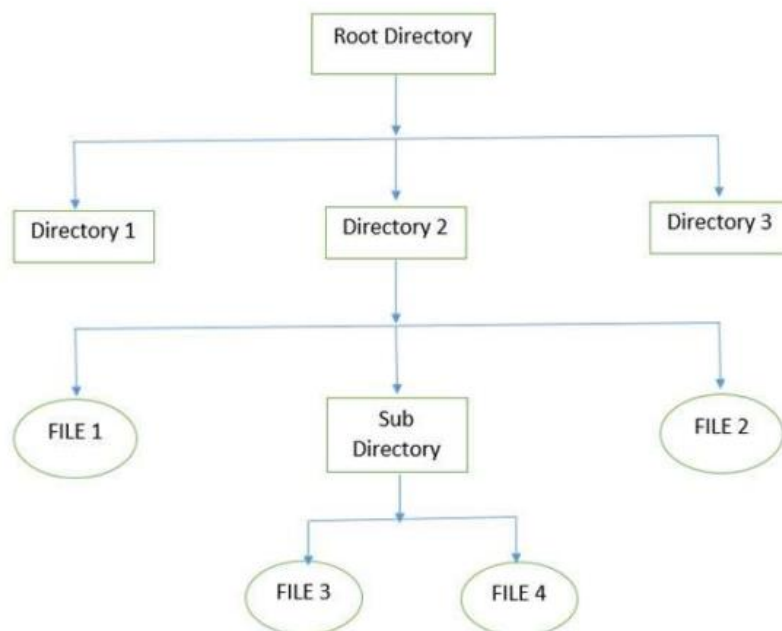
4. It helps the user to modify the data of files or to modify the name of the file in the directories etc.

File management helps users to organize their valuable documents in a systematic manner for better and efficient use of it.

PROJECT PARADIGM

The file is actually the collection of associated information. This file-system prearranged into directory for efficient usage. Every directory has a number of files and other directories. The directory is defined as a bit which distinguish the entries that explained file and subdirectories in the recent directory. By theoretically we may change the file into a directory by changing its bit.

A file system is considered as an element of an operating system that manage the storage space and operation of files on media like disks.



The above figure shows the general hierarchy of the storage in an operating system. In this figure the root directory is present at the highest level in the hierarchical structure. It includes all the subdirectories in which the files are

stored. Subdirectory is a directory present inside another directory in the file storage system. The directory base storage system ensures better organization of files in the memory of the computer system.

MECHANISM AND WORKING:

Menu Code:

```
#include <stdio.h>

int main(void) {
printf("=====\n");
printf("-----File Management Project-----\n");
printf("=====\n");
printf("Welcome, The Main Menu is given below:\n");
printf("1- List all Files and Directories\n");
printf("2- Create New Files\n");
printf("3- Delete Existing Files\n");
printf("4- Rename Files\n");
printf("5- Edit File Content\n");
printf("6- Search Files\n");
printf("7- Details of Particular File\n");
printf("8- View Content of File\n");
printf("9- Sort File Content\n");
printf("10- List only Directories(Folders)\n");
printf("11- List Files of Particular Extension\n");
printf("12- Count Number of Directories\n");
printf("13- Count Number of Files\n");
printf("14- Sort Files in a Directory\n");
printf("0- Exit\n");
printf("\nWhat action you want to Perform?\nEnter 1-14\n");
return 0;
}
```

Main Code:

```
#!/bin/bash

i="0"

while [ $i -lt 100 ]

do

gcc project.c -o proj

./proj

read opt1

if [ $opt1 == 1 ]

then

echo "List all files and Directories here.."

echo "Showing all files and directories...."

sleep 3

echo "Loading.."

sleep 3

echo "-----OutPut-----"

ls

echo " "

elif [ $opt1 == 2 ]

then

echo "Create New Files here.."

echo "Which type of file you want to create !"

echo "1- .c"

echo "2- .sh"

echo "3- .txt"

echo "Enter your choice from 1-3"

read filechoice

if [ $filechoice == 1 ]

then

echo "Enter File Name without .c Extension"

read filename
```

```

touch $filename.c
echo "-----OutPut-----"
echo "File Created Successfully"
echo " "
elif [ $filechoice == 2 ]
then
echo "Enter File Name without .sh Extension"
read filename2
touch $filename2.sh
echo "-----OutPut-----"
echo "File Created Successfully"
echo " "
elif [ $filechoice == 3 ]
then
echo "Enter File Name without .txt Extension"
read filename3
touch $filename3.txt
echo "-----OutPut-----"
echo "File Created Successfully"
echo " "
else
echo "Inavlid Input..Try Again."
echo " "
fi
elif [ $opt1 == 3 ]
then
echo "Delete existing files here.. "
echo "Enter name of File you want to Delete!"
echo "Note: Please Enter full Name with Extension."
read delfile
echo "-----OutPut-----"

```

```

if [ -f "$delfile" ];
then
if [ -f "$delfile" ];
then
rm $delfile
echo "Successfully Deleted."
echo " "
else
echo "File Does not Exist..Try again"
echo " "
fi
elif [ $opt1 == 4 ]
then
echo "-----OutPut-----"
echo "Rename files here.."
echo "Enter Old Name of File with Extension.."
read old
echo "Checking for file..."
sleep 3
if [ -f "$old" ];
then
echo "Ok File Exist."
echo "Now Enter New Name for file with Extension"
read new
mv $old $new
echo "Successfully Rename."
echo "Now Your File Exist with $new Name"
else
echo "$old does not exist..Try again with correct filename."
fi
echo " "

```



```
elif [ $opt1 == 5 ]
then
echo "Edit file content here.."
echo "Enter File Name with Extension : "
read edit
echo "-----OutPut-----"
echo "Checking for file.."
sleep 3
if [ -f "$edit" ];
then
echo "Opening file.."
sleep 3
nano $edit
echo " "
else
echo "$edit File does not exist..Try again."
fi

elif [ $opt1 == 6 ]
then
echo "Search files here.."
echo "Enter File Name with Extension to search"
read f
echo "-----OutPut-----"
if [ -f "$f" ];
then
echo "Searching for $f File"
echo "File Found."
find /home -name $f
echo " "
else
```

```
echo "File Does not Exist..Try again."
echo " "
fi
elif [ $opt1 == 7 ]
then
then
echo "Detail of file here.."
echo "Enter File Name with Extension to see Detail : "
read detail
echo "-----OutPut-----"
echo "Checking for file.."
sleep 4
if [ -f "$detail" ];
then
echo "Loading Properties.."
stat $detail
else
echo "$detail File does not exist..Try again"
fi
echo " "
elif [ $opt1 == 8 ]
then
echo "View content of file here.."
echo "Enter File Name : "
read readfile
echo "-----OutPut-----"
if [ -f "$readfile" ];
then
echo "Showing file content.."
sleep 3
cat $readfile
```

```

else
    echo "$readfile does not exist"
fi
echo " "
elif [ $opt1 == 9 ]
then
    echo "Sort files content here.."
    echo "Enter File Name with Extension to sort :"
    read sortfile
    echo "-----OutPut-----"
    if [ -f "$sortfile" ];
    then
        echo "Sorting File Content.."
        sleep 3
        sort $sortfile
    else
        echo "$sortfile File does not exist..Try again."
    fi
    echo " "
elif [ $opt1 == 10 ]
then
    echo "-----OutPut-----"
    echo "List of all Directories here.."
    echo "showing all Directories..."
    sleep 3
    ls -d */
    echo " "
elif [ $opt1 == 11 ]
then
    echo "List of Files with Particular extensions here.."
    echo "Which type of file list you want to see?"

```

```
echo "1- .c"
echo "2- .sh"
echo "3- .txt"
echo "Enter your choice from 1-3"
read extopt
echo "-----OutPut-----"
if [ $extopt == 1 ]
then
echo "List of .c Files shown below."
sleep 3
ls *.c
elif [ $extopt == 2 ]
then
echo "List of .sh Files shown below."
sleep 3
ls *.sh
elif [ $extopt == 3 ]
then
echo "List of .txt Files shown below."
sleep 3
ls *.txt
else
echo "Invalid Input..Try again.."
fi
echo " "
elif [ $opt1 == 12 ]
then
echo "-----OutPut-----"
echo "Total number of Directories here.."
sleep 3
echo "Counting.."
```

```
sleep 3
echo "Number of Directories are : "
echo */ | wc -w
echo " "
elif [ $opt1 == 13 ]
then
echo "-----OutPut-----"
echo "Total Numbers of Files in Current Directory here.."
echo "Loading all files.."
sleep 3
echo "Number of Files are : "
ls -l | grep -v 'total' | grep -v '^d' | wc -l
echo " "
elif [ $opt1 == 14 ]
then
echo "-----OutPut-----"
echo "Sort Files here.."
echo "Sorting.."
sleep 3
ls | sort
echo " "
elif [ $opt1 == 0 ]
then
echo "Good Bye.."
echo "Successfully Exit"
break
else
echo "Invalid Input..Try again...."
fi
i=$((i+1))
done
```

OUTPUT SCREENSHOTS:

Main Menu:

Main menu of Project that display all the available option to the users. The users need to choose one out of 14 and the particular command will be executed according to the user input.

```
-----File Management Project-----
Welcome, The Main Menu is given below:
1- List all Files and Directories
2- Create New Files
3- Delete Existing Files
4- Rename Files
5- Edit File Content
6- Search Files
7- Details of Particular File
8- View Content of File
9- Sort File Content
10- List only Directories(Folders)
11- List Files of Particular Extension
12- Count Number of Directories
13- Count Number of Files
14- Sort Files in a Directory
0- Exit

What action you want to Perform?
Enter 1-14
```

Choice 01 Output:

If user enter 1 then the List of all Files and Directories will be displayed.

```
What action you want to Perform?
Enter 1-14
1
List all files and Directories here..
Showing all files and directories....
Loading..
-----OutPut-----
F1.txt  another.txt  copypaste.txt.save  hello.c  lab471.sh  lab6task2.c  project.c  test1
F2.txt  assign1.c    copypaste.txt.save.1  ila.sh   lab473.sh  lab6task3.c  project.sh  test2
F3.txt  assign1.txt  empty.txt           lab1.sh  lab474.c   lab8task3.c  record.txt  test3
File.txt assign1T2.c  fact.c              lab2.c   lab5t0.sh  lab9Task02.c  rubab.txt   text.txt
IoT     combine.sh   fibo.c              lab3.sh  lab5t001.sh  main.rs     sample.c    tracefile.txt
OEL_1   copyPaste.txt  fiboNew.c           lab3.txt  lab5t002.sh  mylog        shell.sh     tracing.sh
OEL_2   copypaste.txt  grep.txt            lab471   lab5task2.sh  proj         test.sh      z.txt.txt
```

Choice 02 Output:

If user wants to create new file, then he needs to enter 2.

```

What action you want to Perform?
Enter 1-14
2
Create New Files here..
Which type of file you want to create !
1- .c
2- .sh
3- .txt
Enter your choice from 1-3
1
Enter File Name without .c Extension
z
-----OutPut-----
File Created Successfully
=====

```

Choice 03 Output:

If user wants to delete existing file, then he needs to enter 3.

```

What action you want to Perform?
Enter 1-14
3
Delete existing files here..
Enter name of File you want to Delete!
Note: Please Enter full Name with Extension.
z.c
-----OutPut-----
Successfully Deleted.
=====

```

Choice 04 Output:

If user wants to rename an existing file, then he needs to enter 4.

```

What action you want to Perform?
Enter 1-14
4
-----OutPut-----
Rename files here..
Enter Old Name of File with Extension..
ila.sh
Checking for file...
Ok File Exist.
Now Enter New Name for file with Extension
ali.sh
Successfully Rename.
Now Your File Exist with ali.sh Name
=====

```

Choice 05 Output:

If user wants to edit file content, then he needs to enter 5.

```

What action you want to Perform?
Enter 1-14
5
Edit file content here..
Enter File Name with Extension :
F1.txt
-----OutPut-----
Checking for file..
Opening file..
=====

```

Choice 06 Output:

If user wants to search for a file, then he needs to enter 6.

```
What action you want to Perform?
Enter 1-14
6
Search files here..
Enter File Name with Extension to search
ali.sh
-----OutPut-----
Searching for ali.sh File
File Found.
/home/aqib/ali.sh
=====
```

Choice 07 Output:

If user wants to see the details of file, then he needs to enter 7.

```
What action you want to Perform?
Enter 1-14
7
Detail of file here..
Enter File Name with Extension to see Detail :
F2.txt
-----OutPut-----
Checking for file..
Loading Properties..
File: F2.txt
Size: 13          Blocks: 0          IO Block: 512    regular file
Device: 2h/2d Inode: 1970324836990827 Links: 1
Access: (0666/-rw-rw-rw-) Uid: ( 1000/   aqib)  Gid: ( 1000/   aqib)
Access: 2020-02-29 14:08:21.277237900 +0500
Modify: 2020-06-07 00:41:53.547858800 +0500
Change: 2020-06-07 00:41:53.547858800 +0500
Birth: -
=====
```

Choice 08 Output:

If user wants to view content of file, then he needs to enter 8.

```
What action you want to Perform?
Enter 1-14
8
View content of file here..
Enter File Name :
F3.txt
-----OutPut-----
Showing file content..
B
=====
```

Choice 09 Output:

If user wants to sort the file content, then he needs to enter 9.


```

Enter File Name with Extension to sort :
project.c
-----OutPut-----
Sorting File Content..

printf("0- Exit\n");
printf("1- List all Files and Directories\n");
printf("10- List only Directories(Folders)\n");
printf("11- List Files of Particular Extension\n");
printf("12- Count Number of Directories\n");
printf("13- Count Number of Files\n");
printf("14- Sort Files in a Directory\n");
printf("2- Create New Files\n");
printf("3- Delete Existing Files\n");
printf("4- Rename Files\n");
printf("5- Edit File Content\n");
printf("6- Search Files\n");
printf("7- Details of Particular File\n");
printf("8- View Content of File\n");
printf("9- Sort File Content\n");
printf("\nWhat action you want to Perform?\nEnter 1-14\n");
printf("-----File Management Project-----\n");
printf("-----\n");
printf("-----\n");
printf("Welcome, The Main Menu is given below:\n");
#include <stdio.h>
int main(void) {
return 0;
}
=====

```

Choice 10 Output:

If user wants to list all directories, then he needs to enter 10.

```

What action you want to Perform?
Enter 1-14
10
-----OutPut-----
List of all Directories here..
showing all Directories...
Loading..
IoT/ OEL_1/ OEL_2/ mylog/ test1/ test2/ test3/
=====

```

Choice 11 Output:

If user wants to list all files with the same extension, then he needs to enter 11.

```

What action you want to Perform?
Enter 1-14
11
List of Files with Particular extensions here..
Which type of file list you want to see?
1- .c
2- .sh
3- .txt
Enter your choice from 1-3
2
-----OutPut-----
List of .sh Files shown below.
Loading..
ali.sh      lab1.sh  lab4T1.sh lab5t0.sh  lab5t002.sh  project.sh  test.sh
combine.sh  lab3.sh  lab4T3.sh lab5t001.sh lab5task2.sh  shell.sh    tracing.s
=====

```

Choice 12 Output:

If user wants to number of directories, then he needs to enter 12.

```

What action you want to Perform?
Enter 1-14
12
-----OutPut-----
Total number of Directories here..
Loading all directories..
Counting..
Number of Directories are :
7
=====

```

Choice 13 Output:

If user wants to count number of files, then he needs to enter 13.

```

What action you want to Perform?
Enter 1-14
13
-----OutPut-----
Total Numbers of Files in Current Directory here..
Loading all files..
Counting..
Number of Files are :
49

```

Choice 14 Output:

If user wants to sort all files in a directory, then he needs to enter 14.

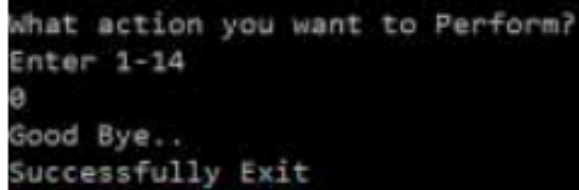
```

lab471
lab471.sh
lab473.sh
lab474.c
lab5t0.sh
lab5t001.sh
lab5t002.sh
lab5task2.sh
lab6task2.c
lab6task3.c
lab8task3.c
lab9Task02.c
main.rs
mylog
proj
project.c
project.sh
record.txt
rubab.txt
sample.c
shell.sh
test.sh
test1
test2
test3
text.txt
tracefile.txt
tracing.sh
z.txt.txt

```

Exit option:

If user wants to exit from Management system, then he needs to enter 0.



```
What action you want to Perform?  
Enter 1-14  
0  
Good Bye..  
Successfully Exit
```

FUNCTIONALITIES:

The following are some of the functionalities or tasks performed by file management

system:

1. List all Files and Directories.
2. Create New Files.
3. Delete Existing Files.
4. Rename an Existing Files.
5. Edit Files Content.
6. Search for Files.
7. Details of Particular File.
8. View Content of File.
9. Sort Files Content.
10. List only Directories.
11. List Files of particular Extension.
12. Count Number of Directories.
13. Sort all Files in a Directories.

The details of all above functionalities are already explained under MECHANISM AND WORKING heading in the form of code of each function.

REMAINING CODE MODULES, API'S AND PLATFORMS:

No other remaining side work apart from the displayed work above is used in this project.

All the functionalities and code of each function is explained above. In this project we use Ubuntu subsystem terminal with C language and bash scripting. So, no others platform, API or plug ins used in this project.

FUTURE WORK:

This is the most basic version of file management system. So, in future we can improve the current version's functionalities and can add more new functionalities to the system. In the current version of files management system there are 13 different options for a user to manage files and directories. In future we can add more choices for users by understanding the advanced concept about file management in Linux operating system. So, this will definitely help users to manage files in a more easy and comfortable manners.

CONCLUSION:

The project contains some basic functionalities regarding file management like creating new files, delete existing files, rename files, edit files, read or write files and so on. All the functionalities are working on the basis of user's input from keyboard. There are different basic functions that users can perform on files. These functions are written in C language and bash scripting. All these functionalities are discussed above in the form of code as well as in simple natural language. So, everyone having the basic knowledge of computer can use this file management system to perform different functions on files.

REFERENCES:

- Main Idea from includehelp.com <https://www.includehelp.com/operating-systems/file-management-in-operatingsystem.aspx>
- https://linuxhint.com/linux_stat_command/
- How to rename a file by Mazhar MIK on askubuntu.com <https://askubuntu.com/questions/280768/how-to-rename-a-file-interminal#:~:text=A%20simple%20way%20to%20rename,from%20one%20name%20to%20another.&text=where%20%E2%80%9Cfile1.>>