AC Patil
College *of* Engineering

Jawahar Education Society's
A. C. Patil College of Engineering, Kharghar
Navi Mumbai 410210

Student Name:

Course Name: C.S.E. (IoT CS BC)

Year: S.E.

Roll No.:

PRN No.: 2211110

Course code: CSL403

Semester: IV

Experiment Evaluation Sheet

Experiment No.: 2

Experiment Name:
Explore the usage of system calls for file, directory and process management.

Sr No.	**Evaluation Criteria**	**Marks (Out of 9)**	**Performance Date**	**Correction Date and Signature of Instructor**
1	**Experiment Performance**			
2	**Journal Performance**			
3	**Punctuality**			
Total				

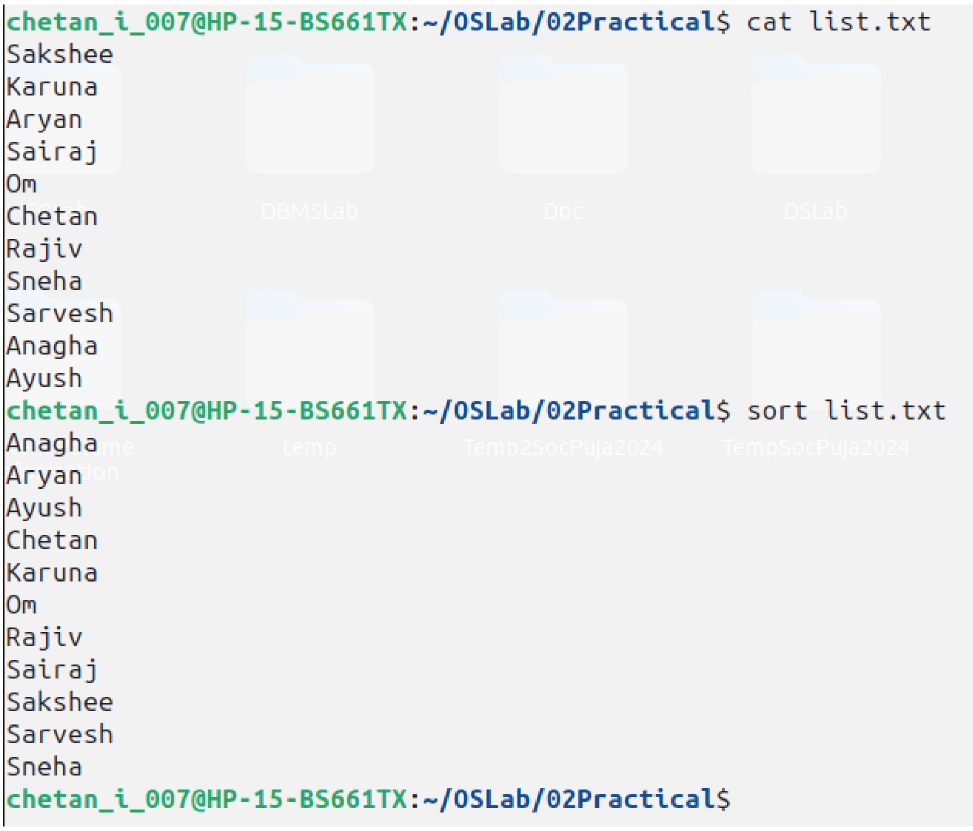
Aim : Explore the usage of system calls for file, directory and process management.
open, read, write, close, getpid, sort, grep, awk,

Software required : Terminal

Theory :

i) Program illustrating sort command :

The ‘sort’ command in Linux is a powerful utility designed to arrange the contents of a file or input stream in a specified order. Its primary purpose is to organize records alphabetically or numerically, making it a valuable tool for data manipulation and analysis.
Syntax :
Sort file_name.txt

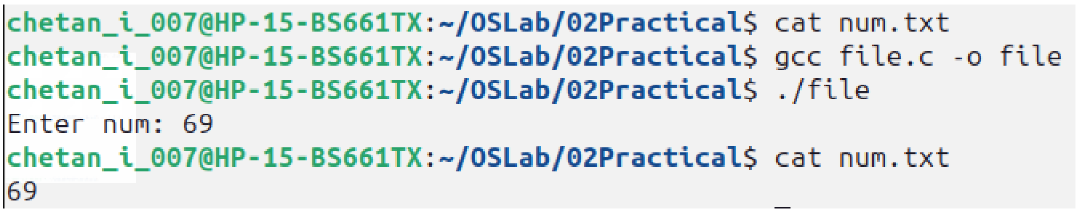


```
chetan_i_007@HP-15-BS661TX:~/OSLab/02Practical$ cat list.txt
Sakshee
Karuna
Aryan
Sairaj
Om
Chetan
Rajiv
Sneha
Sarvesh
Anagha
Ayush
chetan_i_007@HP-15-BS661TX:~/OSLab/02Practical$ sort list.txt
Anagha
Aryan
Ayush
Chetan
Karuna
Om
Rajiv
Sairaj
Sakshee
Sarvesh
Sneha
chetan_i_007@HP-15-BS661TX:~/OSLab/02Practical$
```

ii) Program to illustrate Open and Close system call:

The gcc command is the GNU Compiler Collection and is used to compile C programs. It compiles the C program in the file and produces an executable file using the -o option.
The ‘./’ specifies the current directory and executes the compiled program.
These commands are essential for compiling and executing C programs in a Linux environment

```
#include <stdio.h>
#include <stdlib.h>
int main() {
        int num;
        FILE *fptr;
        fptr = fopen("num.txt","w");
        if(fptr == NULL) {
                printf("Error!");
                exit(1);
        }
        printf("Enter num: ");
        scanf("%d",&num);
        fprintf(fptr,"%d\n",num);
        fclose(fptr);
        return 0;
}
```



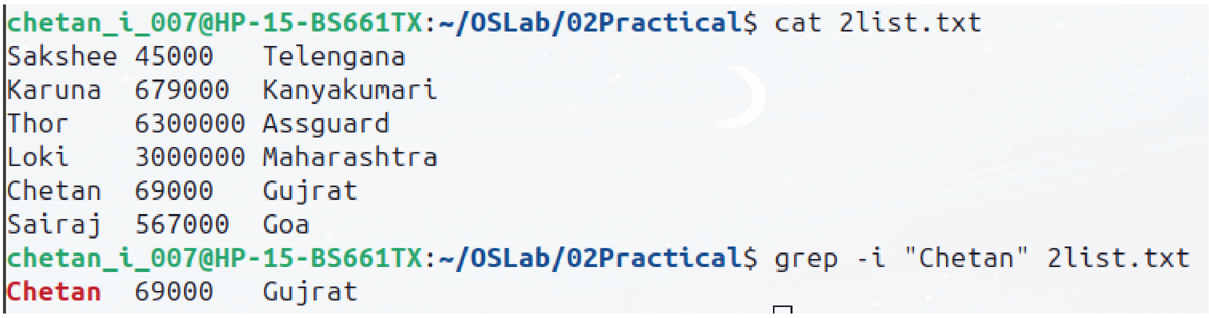
```
chetan_i_007@HP-15-BS661TX:~/OSLab/02Practical$ cat num.txt
chetan_i_007@HP-15-BS661TX:~/OSLab/02Practical$ gcc file.c -o file
chetan_i_007@HP-15-BS661TX:~/OSLab/02Practical$ ./file
Enter num: 69
chetan_i_007@HP-15-BS661TX:~/OSLab/02Practical$ cat num.txt
69
```

iii) Program illustrating grep command :

The grep command in Linux is a powerful and versatile utility used for searching and manipulating text within files. grep searches for the specified pattern within one or more files. If no files are provided, it reads from standard input. By default, grep is case-sensitive. The -i option enables caseinsensitive matching, broadening the scope of search results.

Syntax:

grep [options] pattern [files]



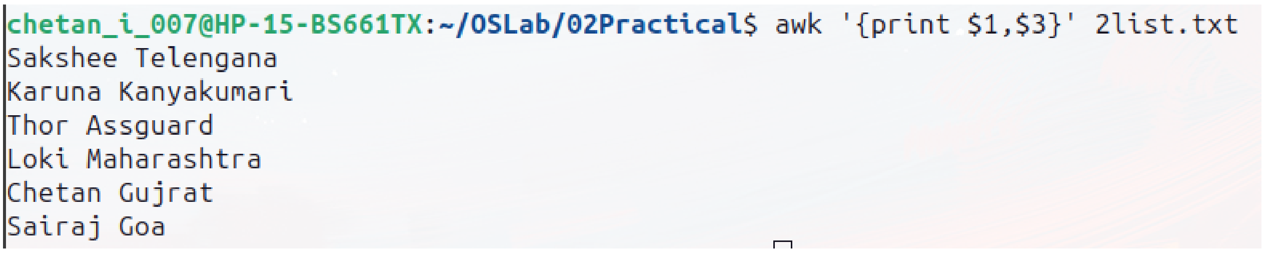
```
chetan_i_007@HP-15-BS661TX:~/OSLab/02Practical$ cat 2list.txt
Sakshee 45000   Telengana
Karuna  679000  Kanyakumari
Thor    6300000 Assguard
Loki    3000000 Maharashtra
Chetan  69000   Gujrat
Sairaj  567000  Goa
chetan_i_007@HP-15-BS661TX:~/OSLab/02Practical$ grep -i "Chetan" 2list.txt
Chetan  69000   Gujrat
```

iv) Program illustrating {awk} command:

The awk command in Linux is a powerful text processing tool that allows users to perform pattern scanning and text extraction. Named after its original developers—Alfred Aho, Peter Weinberger, and Brian Kernighan—awk excels at processing and analyzing structured textual data, making it a versatile choice for data manipulation.

Syntax :

$ awk 'pattern { action }' file(s)]



```
chetan_i_007@HP-15-BS661TX:~/OSLab/02Practical$ awk '{print $1,$3}' 2list.txt
Sakshee Telengana
Karuna Kanyakumari
Thor Assguard
Loki Maharashtra
Chetan Gujrat
Sairaj Goa
```

Conclusion :

Tools like sort for arranging data, grep for efficient text search, and awk for versatile structured data processing were explored