Experiment No 3
Class: SE Comp
Year: 2020-21
Performed by: Danyl Fernandes, 72

Aim: Implement any one of the basic commands of linux like ls, cp, mv, and other kernel APIs

Theory:

- The Linux API is the kernel–user space API, which allows programs in user space to access system resources and services of the Linux kernel.
- It is composed of the System Call Interface of the Linux kernel and the subroutines in the GNU C Library (glibc).
- The focus of the development of the Linux API has been to provide the usable features of the specifications defined in POSIX in a way which is reasonably compatible, robust and performant, and to provide additional useful features not defined in POSIX, just as the kernel– user space APIs of other systems implementing the POSIX API also provide additional features not defined in POSIX.

System Call Interface of the Linux kernel:

- System Call Interface is the denomination for the entirety of all implemented and available system calls in a kernel.
- A C standard library is a wrapper around the system calls of the Linux kernel; the combination of the Linux kernel System Call Interface and a C standard library is what builds the Linux API.
    - GNU C Library (glibc)
    - Embedded GLIBC
    - uClibc
    - klibc

Commands:

- **ls:** is a linux shell command that lists directory contents and file structures
- **cp:** cp stands for copy. This command is used to copy files or groups of files or directories. It creates an exact image of a file. This commands requires at least two filenames in its arguments
- **mv:** mv stands for move. It is used to move one or more files or directories from one location to another in a file system like UNIX

Conclusion:

- We were successfully able to implement the commands using Kernel API