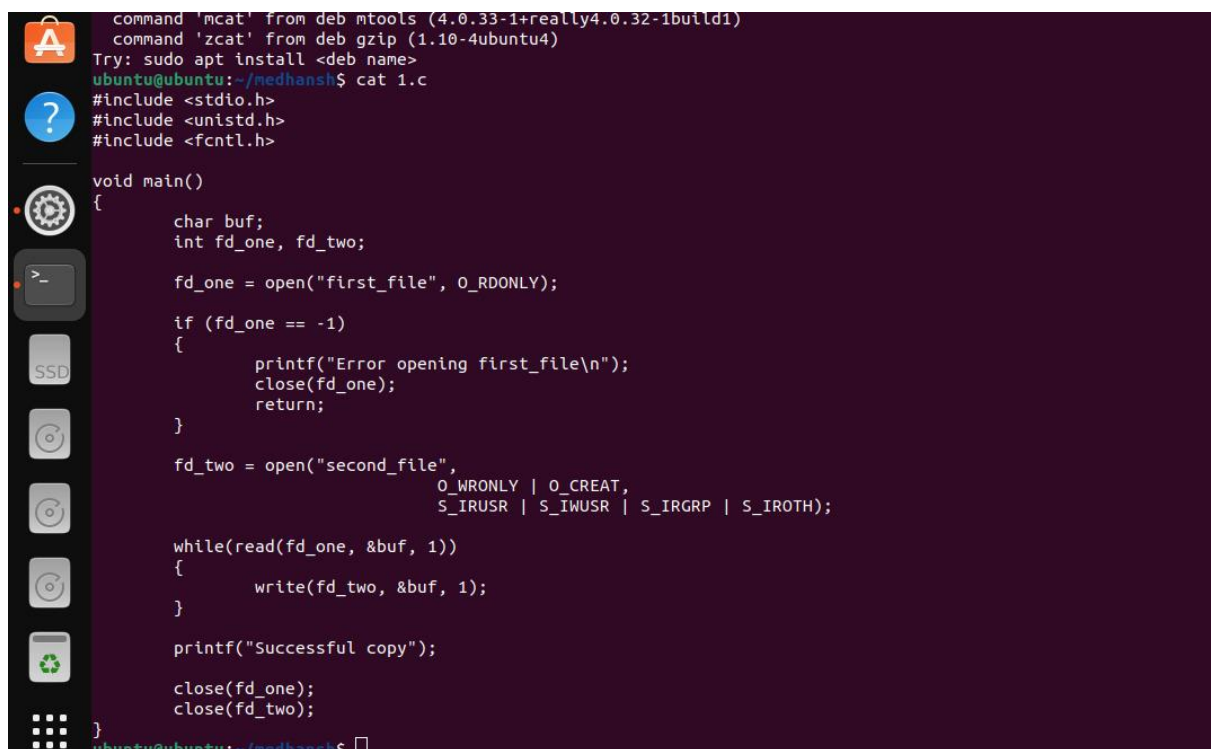


EXPERIMENT-3

AIM-The objective of this laboratory is to introduce the working behind the copy(cp) command. Copy(cp) command uses system calls like open, read, write and lseek to copy the contents of one file to another and read what users enters and write the same in the file.

1. Create a program using system calls to copy either the first half or the second half of a file into a new file.

A terminal window with a dark purple background and a sidebar on the left containing icons for applications, help, settings, terminal, SSD, and trash. The terminal text shows the installation of 'mcat' and 'zcat' from Ubuntu repositories, followed by the execution of 'cat 1.c' which displays the source code of a C program. The program opens 'first_file' in read-only mode and 'second_file' in write-only mode with O_CREAT. It then reads from 'first_file' in a loop and writes to 'second_file' until the end of the file is reached. Finally, it prints 'Successful copy' and closes both files.

```
command 'mcat' from deb mtools (4.0.33-1+really4.0.32-1build1)
command 'zcat' from deb gzip (1.10-4ubuntu4)
Try: sudo apt install <deb name>
ubuntu@ubuntu:~/nedhansh$ cat 1.c
#include <stdio.h>
#include <unistd.h>
#include <fcntl.h>

void main()
{
    char buf;
    int fd_one, fd_two;

    fd_one = open("first_file", O_RDONLY);

    if (fd_one == -1)
    {
        printf("Error opening first_file\n");
        close(fd_one);
        return;
    }

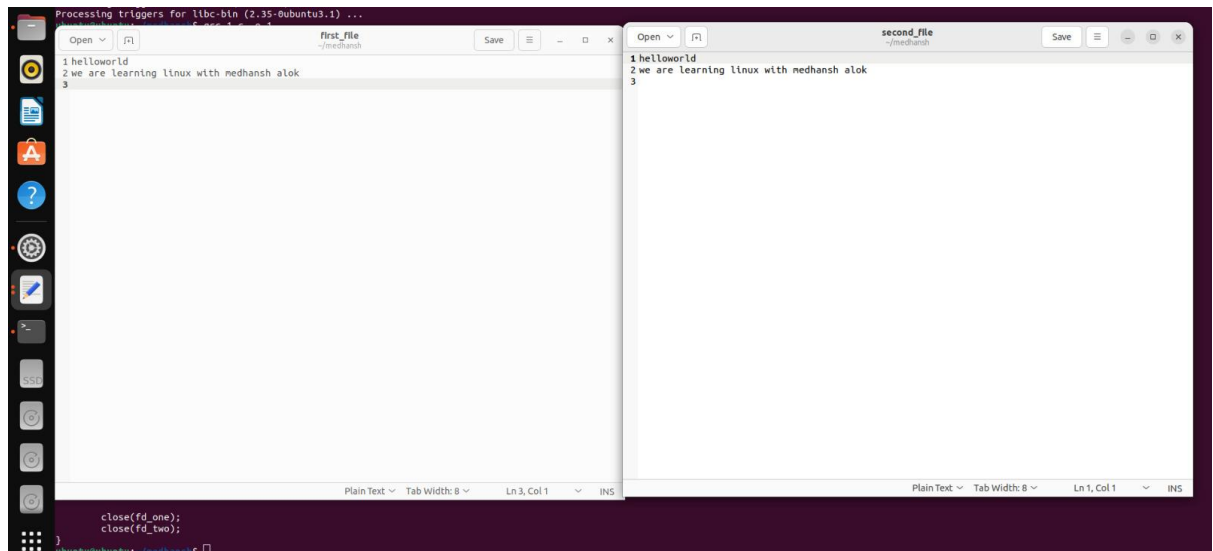
    fd_two = open("second_file",
                  O_WRONLY | O_CREAT,
                  S_IRUSR | S_IWUSR | S_IRGRP | S_IROTH);

    while(read(fd_one, &buf, 1))
    {
        write(fd_two, &buf, 1);
    }

    printf("Successful copy");

    close(fd_one);
    close(fd_two);
}
```

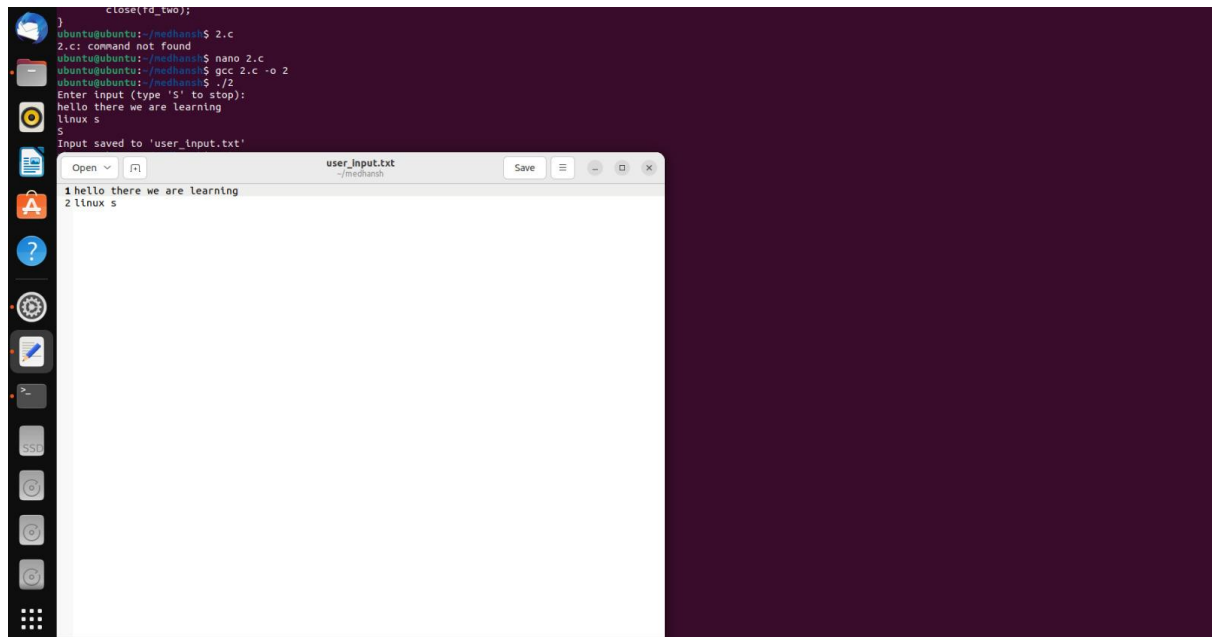
ubuntu@ubuntu:~/nedhansh\$



2. Develop a program using system calls to read input from the console until the user inputs 'S', and then save the input into a file.



```
close(fd_two);
}
ubuntu@ubuntu:~/medhansh$ 2.c
2.c: command not found
ubuntu@ubuntu:~/medhansh$ nano 2.c
ubuntu@ubuntu:~/medhansh$ gcc 2.c -o 2
ubuntu@ubuntu:~/medhansh$ ./2
Enter input (type 'S' to stop):
hello there we are learning
linux s
S
Input saved to 'user_input.txt'
```



```
user_input.txt
~/medhansh
Save
1 hello there we are learning
2 linux s
```

3. Design a program using system calls to read the contents of a file without using a char array and display the contents directly on the console.

```
bytes_written = write(fd, &input, sizeof(input));
if (bytes_written == -1) {
    perror("Error writing to file");
    close(fd);
    return 1;
}
}

close(fd);
printf("Input saved to 'user_input.txt'\n");

return 0;
}

ubuntu@ubuntu:~/medhansh$ nano 3.c
ubuntu@ubuntu:~/medhansh$ gcc 3.c -o 3
ubuntu@ubuntu:~/medhansh$ ./3
Linux is fun and loving!!!
ubuntu@ubuntu:~/medhansh$ cat 3.c
#include <stdio.h>
#include <fcntl.h>
#include <unistd.h>
#include <sys/stat.h>
#include <sys/types.h>

int main() {
    int fd = open("your_file.txt", O_RDONLY);

    if (fd == -1) {
        perror("Error opening file");
        return 1;
    }

    struct stat file_stats;
    if (fstat(fd, &file_stats) == -1) {
        perror("Error getting file size");
        close(fd);
        return 1;
    }

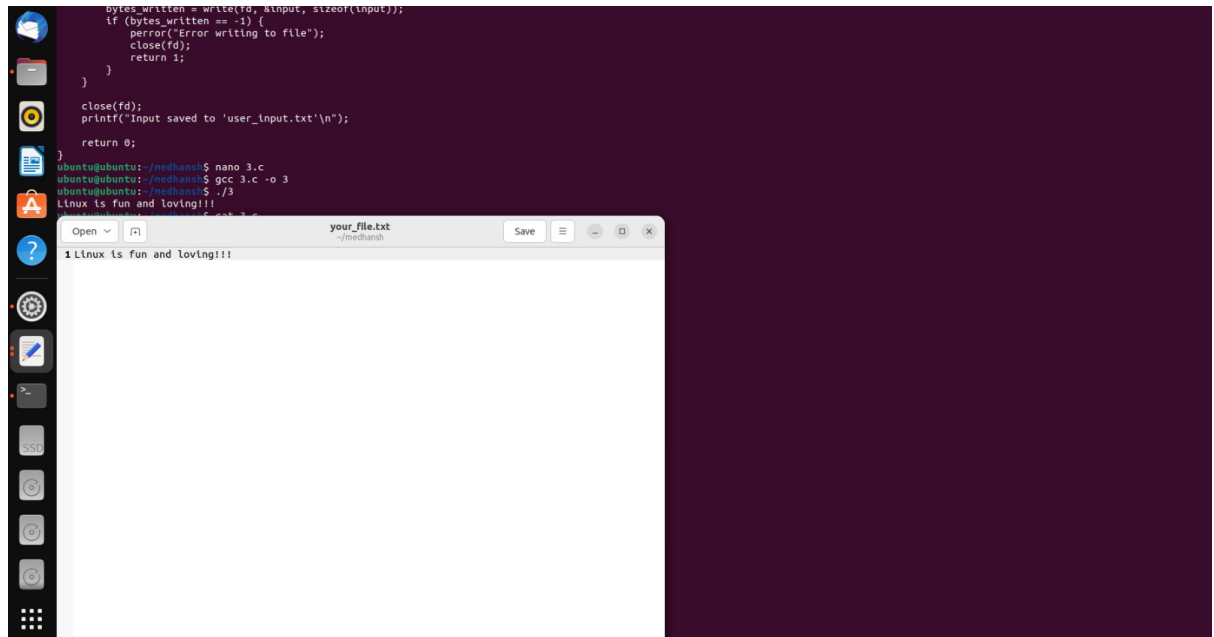
    off_t file_size = file_stats.st_size;
    char placeholder;

    for (off_t i = 0; i < file_size; i++) {
        read(fd, &placeholder, sizeof(char));
        write(STDOUT_FILENO, &placeholder, sizeof(char));
    }

    close(fd);

    return 0;
}

ubuntu@ubuntu:~/medhansh$
```



Submitted to
Ashish Kumar Singh

Submitted by
Medhansh Alok
12219310