OS - Practical - 1

Name: Rishabh Jain

Branch &amp; Sem: CSE - IV

Sec: A

Roll no.:54

Aim: Write a C program to implement I/O System Calls of Linux.

a) Create a file

b) Read contents of a file

c) Write to a file

d) Read contents of a file in a reverse order

e) Search the file to find the given pattern

f) Delete a file

g) To print file status using stat

h)To print file status using fstat

CODE:

*#include* <stdio.h>

*#include* <fcntl.h>

*#include* <unistd.h>

*#include* <sys/types.h>

*#include* <string.h>

*#include* <stdlib.h>

void main()

{

    char data;

    char buf[100], pat[10], temp[1024];

    int ch1;

    int ch = 0;

    int fd, i, n, m;

    int charcount = 0;

    int ret;

    char fname[10];

    char \*newline;

    FILE \*fp;

    struct stat stats;

*while* (1)

    {

        printf("\n\nPerform different Operations on files\n ");

        printf("1.) Create File \n 2.) Read File\n 3.) Write on File\n 4.) Read the content of the file in reverse order\n 5.) Search the Entered String in the File\n 6.) Delete the File\n 7.) File status using stat\n 8.) File status using fstat\n 9.) Exit\n");

        printf("Enter your choice from above Options :\t ");

        scanf("%d", &ch1);

*switch* (ch1)

        {

*case* 1:

            printf("Enter file name which you have to create.\n");

            scanf("%s", fname);

            fd = creat(fname, 0777);

*if* (fd == -1)

            {

                printf("Error\n");

            }

*else*

            {

                printf("File created\n");

                printf("Enter the contents to write in File: \n");

                scanf("%s", buf);

                charcount = strlen(buf);

                write(fd, buf, charcount);

            }

            close(fd);

*break*;

*case* 2:

            printf("Enter file name which you have to open: \n");

            scanf("%s", fname);

            fd = open(fname, O\_RDONLY, 0777);

*if* (fd == -1)

            {

                printf("Error\n");

            }

*else*

            {

                printf("File Opened\n");

                n = lseek(fd, 0, 2);

                printf("The size of the file is: %d\n", n);

                printf("The contents are as follows: \n ");

                lseek(fd, 0, 0);

*for* (int i = 1; i <= n; i++)

                {

                    read(fd, &data, 1);

                    printf("%c", data);

                }

            }

            close(fd);

*break*;

*case* 3:

            printf("Enter file name which you have to open:\n");

            scanf("%s", fname);

            fd = open(fname, O\_WRONLY | O\_APPEND, 0777);

*if* (fd == -1)

            {

                printf("Error\n");

            }

*else*

            {

                printf("File Opened\n");

                printf("The contents to be written are : \n");

                scanf("%s", buf);

                charcount = strlen(buf);

                write(fd, buf, charcount);

            }

            close(fd);

*break*;

*case* 4:

            printf("Enter file name which you have to open:\n");

            scanf("%s", fname);

            fd = open(fname, O\_RDONLY, 0777);

*if* (fd == 1)

            {

                printf("Error\n");

            }

*else*

            {

                printf("File Opened\n");

                printf("The contents in reverse order are : \n");

                n = lseek(fd, 0, 2);

                lseek(fd, -1, 2);

*while* (n-- > 0)

                {

                    read(fd, &data, 1);

                    printf("%c", data);

                    lseek(fd, -2, 1);

                }

                close(fd);

*break*;

            case 5:

*/\*printf("Enter file name: ");*

*scanf("%s",fname);*

*fd = open(fname, O\_RDWR | O\_CREAT, 0777);*

*if(fd == -1)*

*printf("FILE NOT FOUND/ CREATED\n");*

*else{*

*printf("Enter a String to compare : ");*

*scanf("%s",str);*

*n = lseek(fd, 0, 2);*

*int len = strlen(str);*

*lseek(fd, 0, 0);*

*int ptr = 1;*

*while (n - len > 0){*

*char temp[len + 1];*

*read(fd, temp, sizeof(temp));*

*str[len] = temp[len];*

*if (strstr(temp, str) != 0){*

*printf("Match Found\n");*

*}*

*lseek(fd, ptr, 0);*

*ptr++;*

*n--;*

*}*

*}\*/*

                printf("\nEnter file name: ");

                scanf("%s", fname);

                printf("Enter the pattern to be searched (If the pattern is present ,all contents would be displayed): ");

                scanf("%s", pat);

                fp = fopen(fname, "r");

                printf("File Opened\n");

*while* (fgets(temp, 1000, fp) != NULL)

                {

*if* (newline = strstr(temp, "\n"))

                    {

                        \*newline = '\0';

                    }

*if* (strstr(temp, pat) != 0)

                    {

                        printf("%s\n", temp);

                    }

                }

                close(fd);

*break*;

            case 6:

                printf("ENTER FILE NAME TO BE DELETED:\n");

                scanf("%s", fname);

                unlink(fname);

*break*;

            case 7:

                printf("\nEnter file name: ");

                scanf("%s", fname);

*if* (stat(fname, &stats) == 0)

                {

                    printf("\nFile access: ");

*if* (stats.st\_mode & R\_OK)

                        printf("read ");

*if* (stats.st\_mode & W\_OK)

                        printf("write ");

*if* (stats.st\_mode & X\_OK)

                        printf("execute");

*// File size*

                    printf("\nFile size: %d", stats.st\_size);

                }

*break*;

            case 8:

                printf("\nEnter file name: ");

                scanf("%s", fname);

                fd = open(fname, O\_RDWR | O\_CREAT, 0777);

*if* (fstat(fd, &stats) == 0)

                {

                    printf("\nFile access: ");

*if* (stats.st\_mode & R\_OK)

                        printf("read ");

*if* (stats.st\_mode & W\_OK)

                        printf("write ");

*if* (stats.st\_mode & X\_OK)

                        printf("execute");

*// File size*

                    printf("\nFile size: %d", stats.st\_size);

                }

*break*;

            case 9:

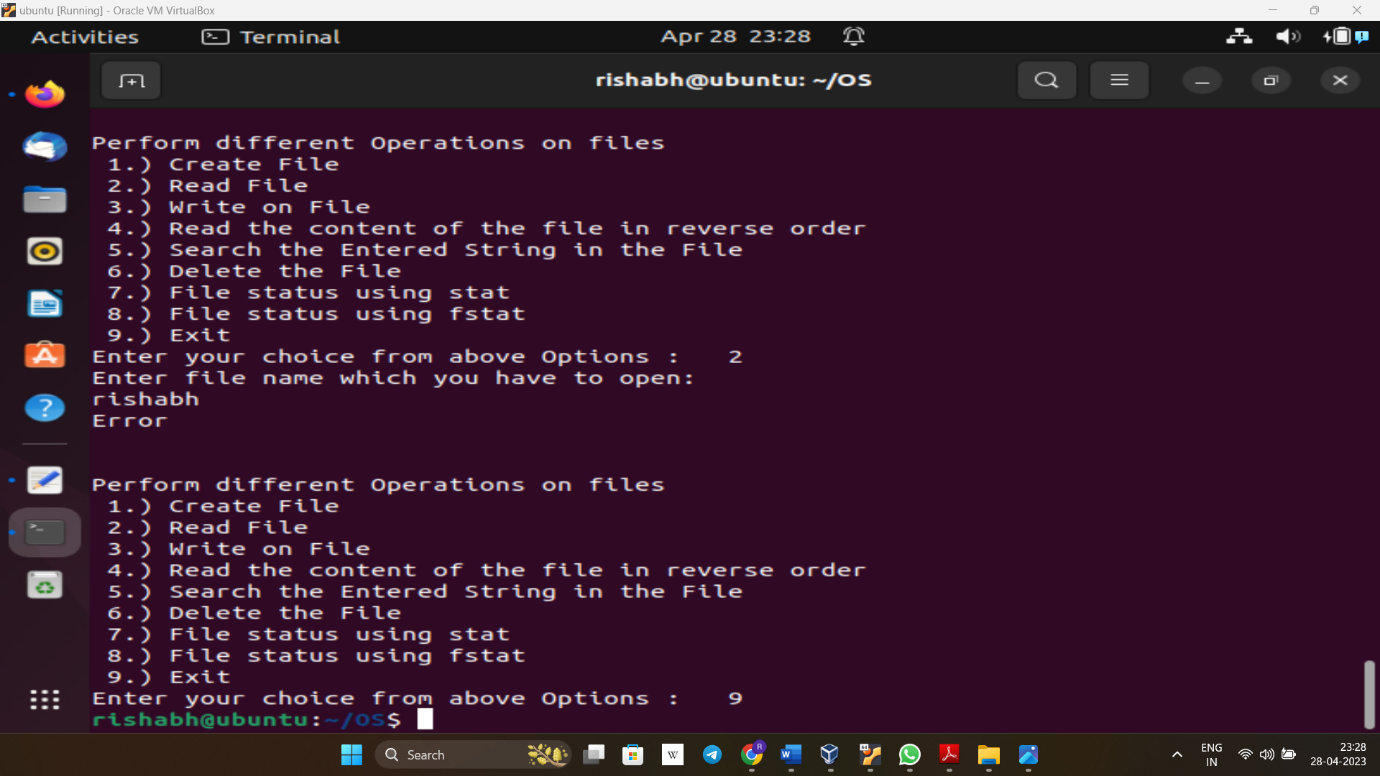
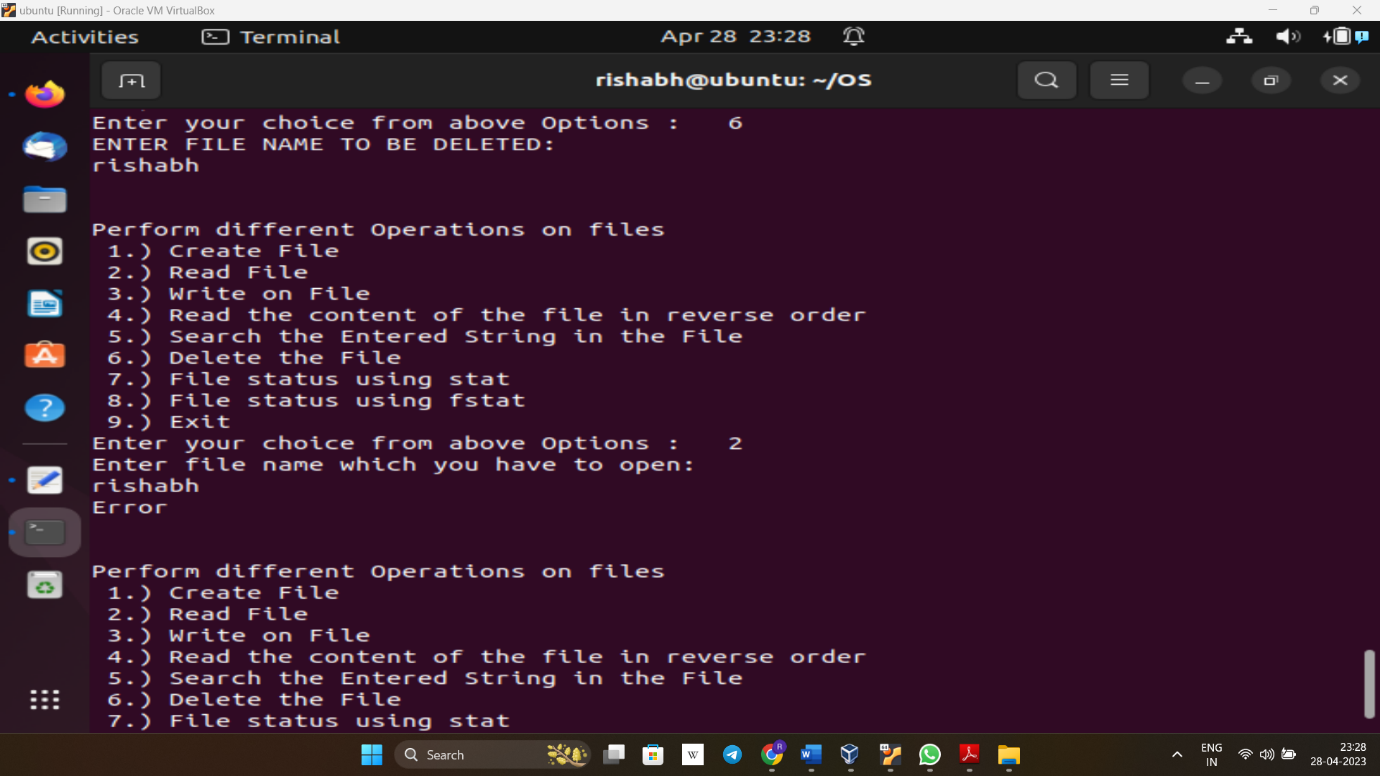
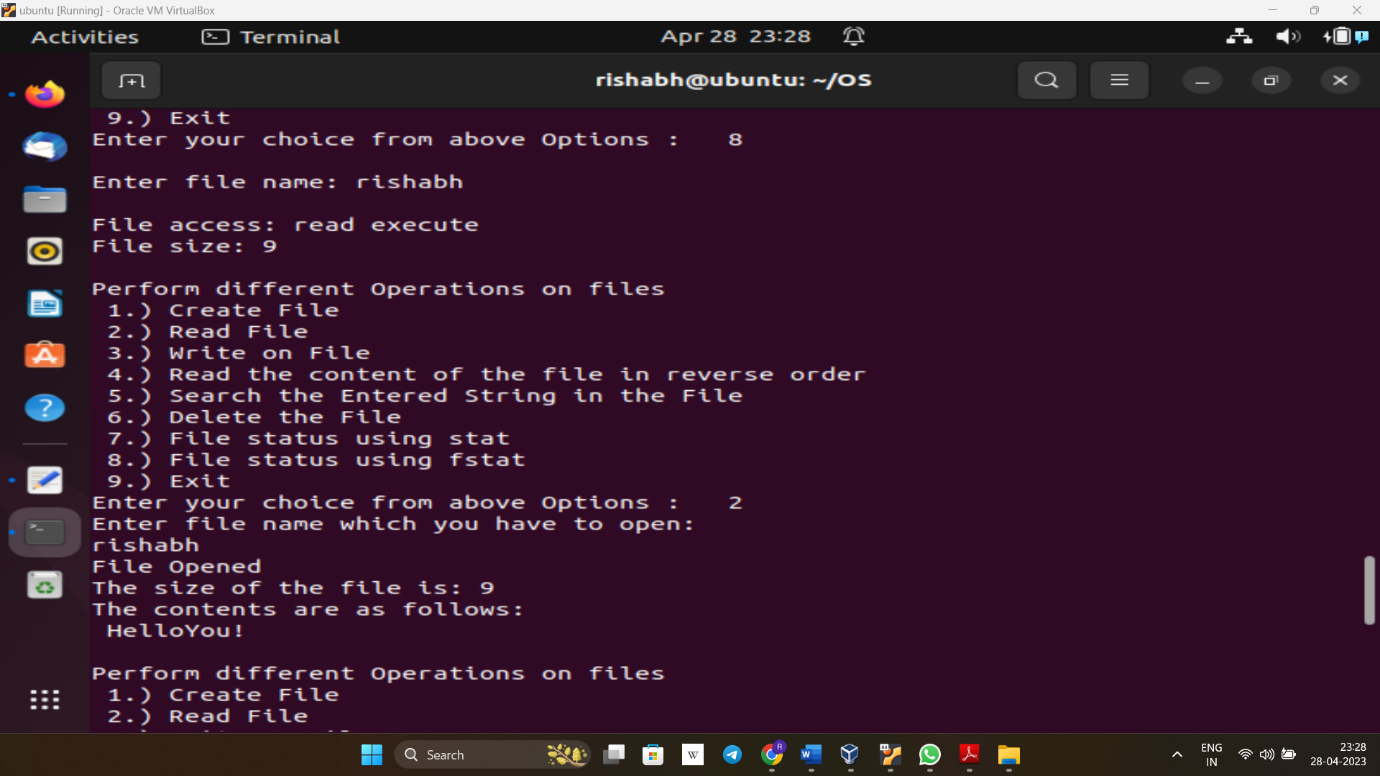
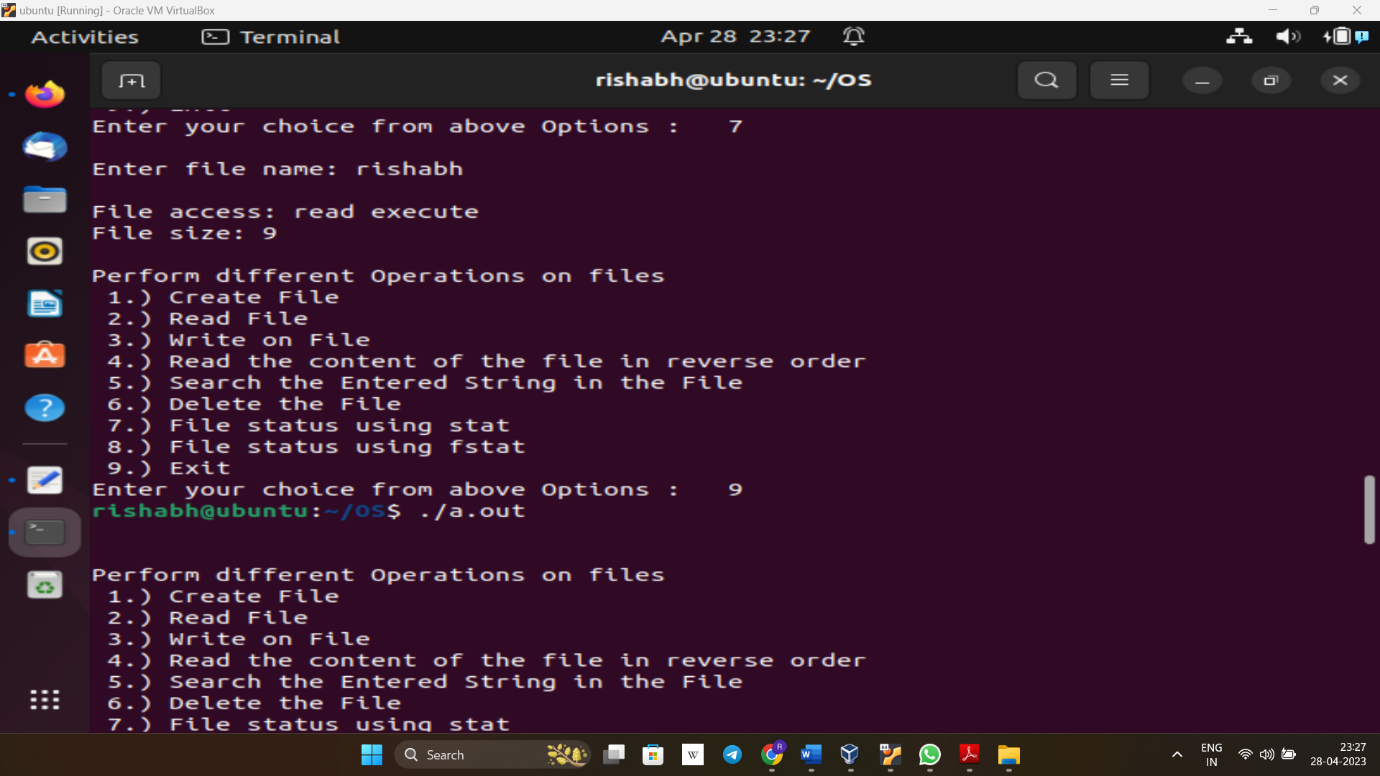
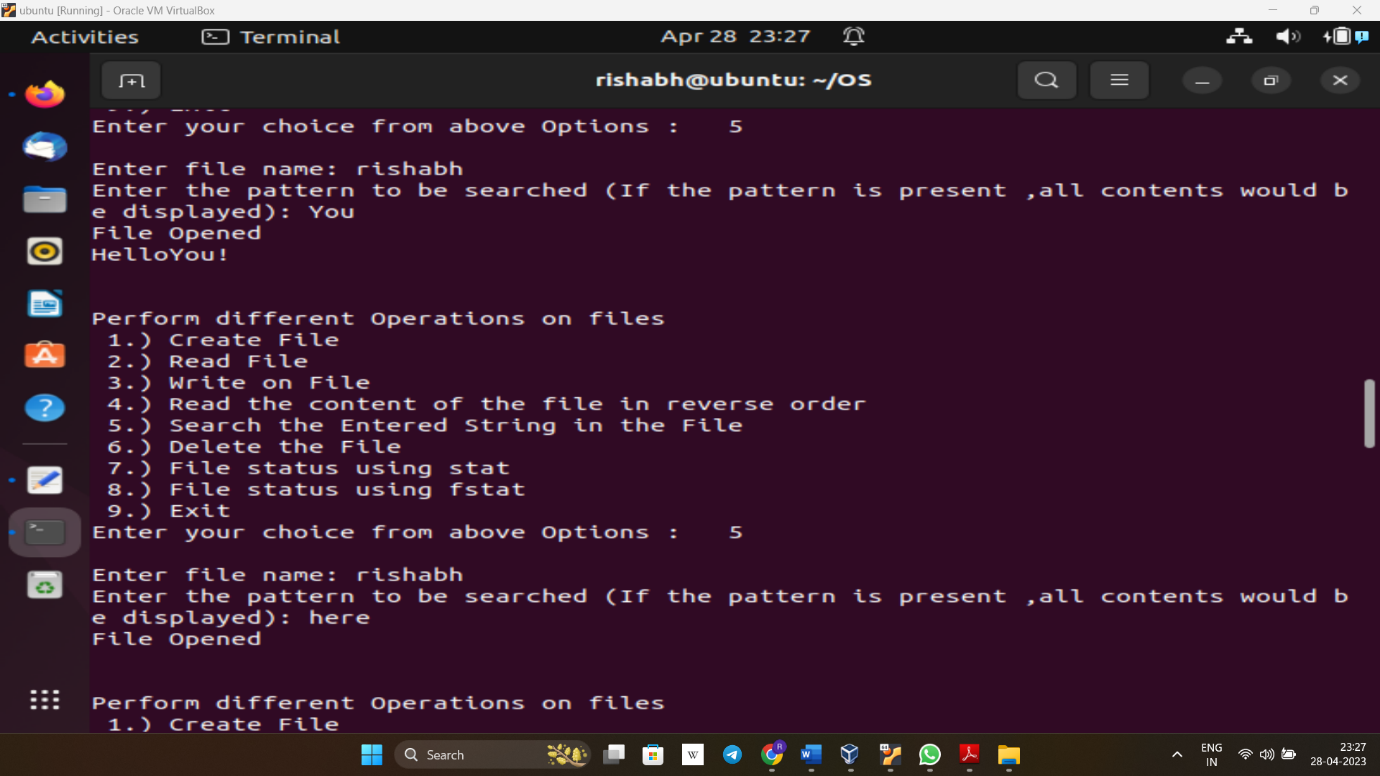
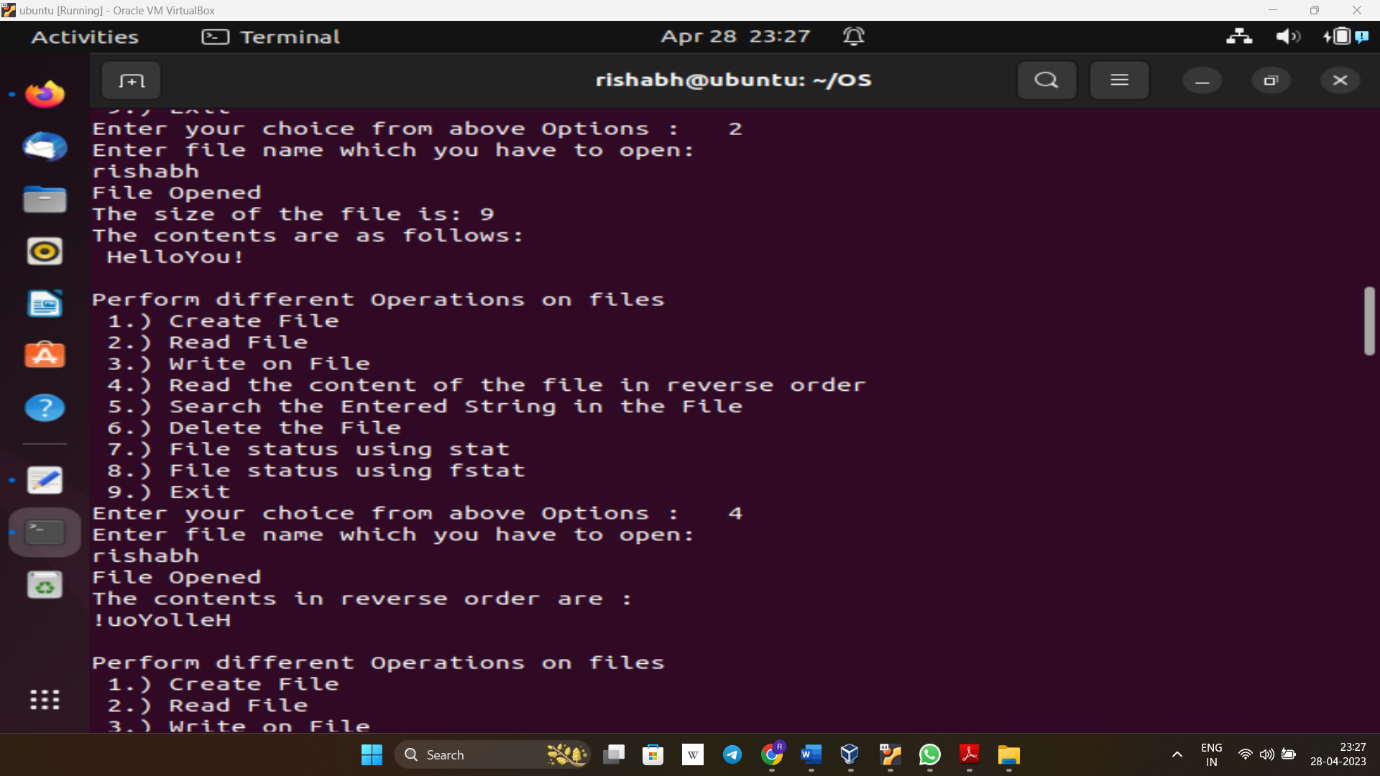
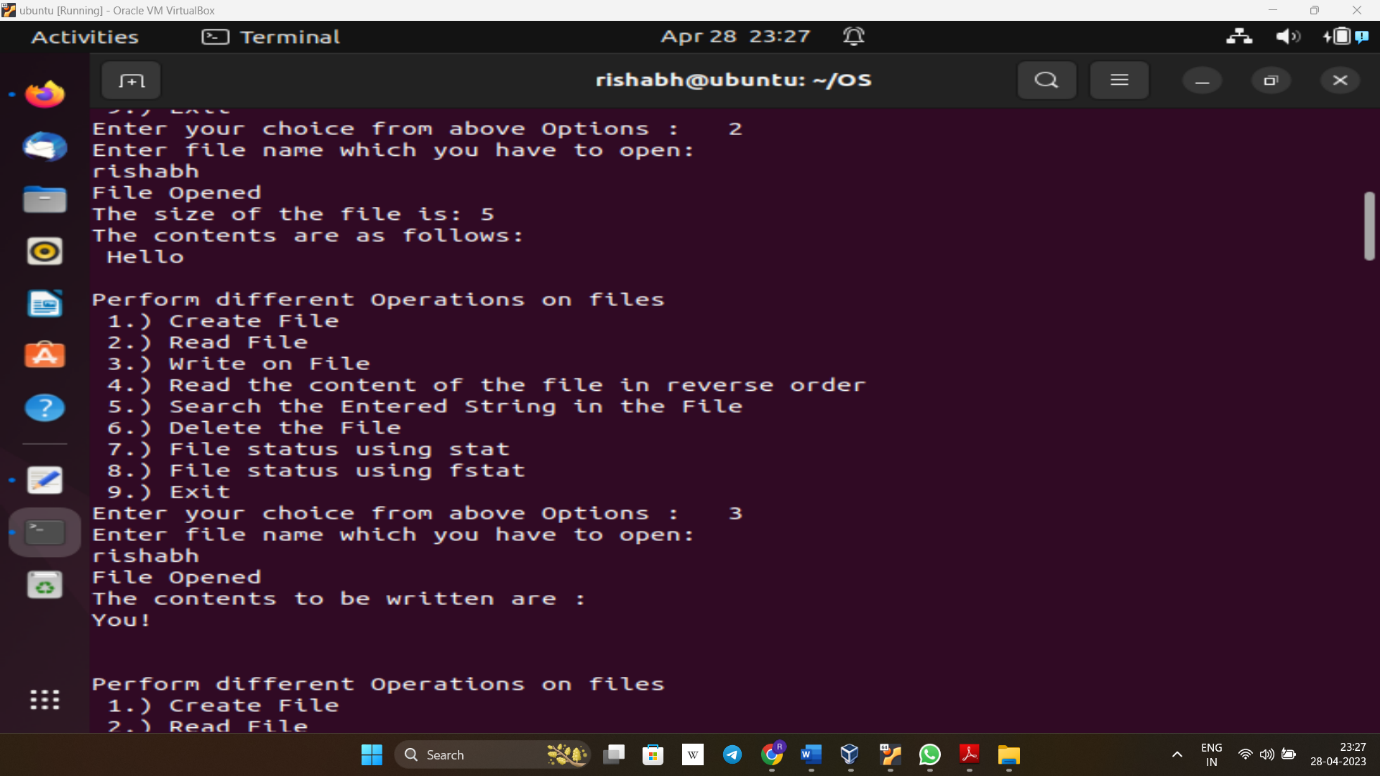
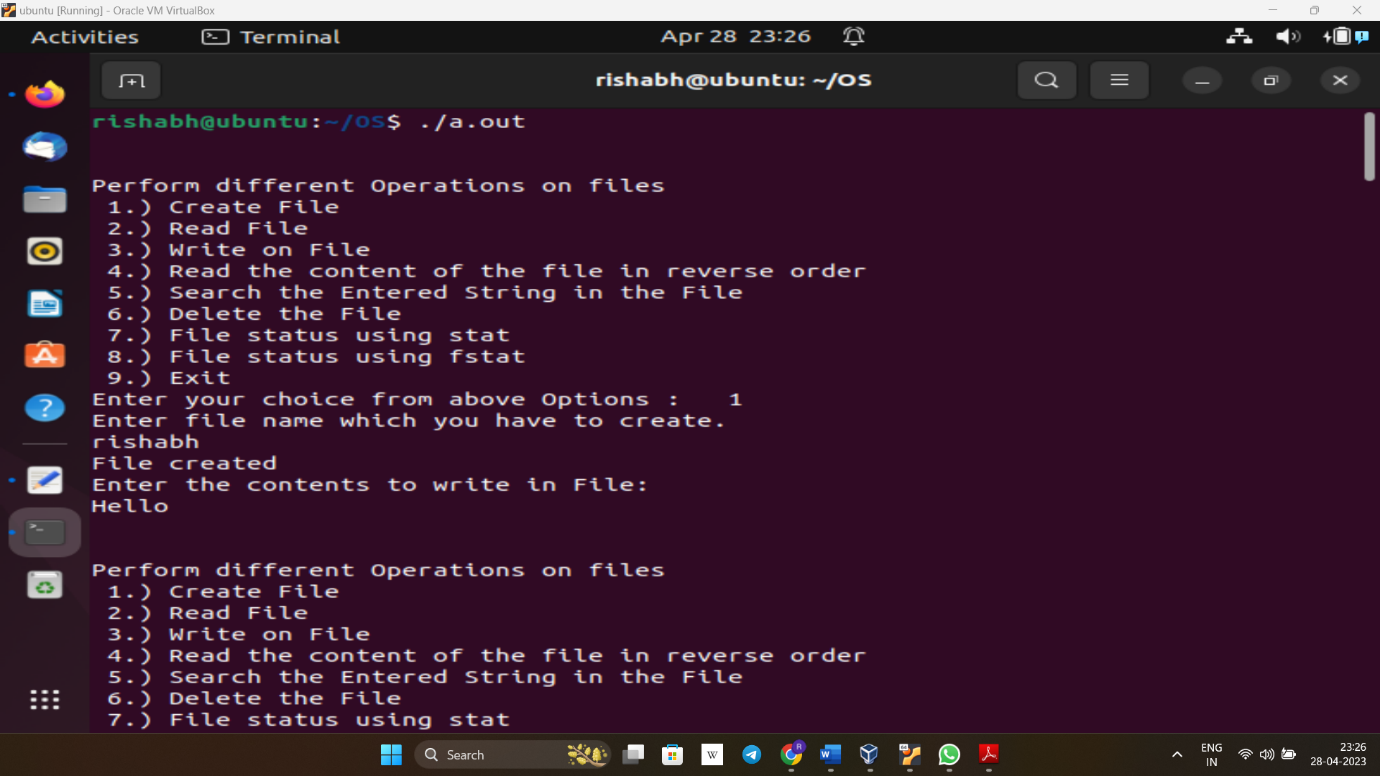
                exit(0);

            }

        }

    }

}

**OUTPUT:**