OPERATING SYSTEM - PRACTICAL 3

NAME – Sakshi Soni

ROLL NO – 13

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 (Grep command)

f) Delete a file

g) To print file status using stat

h)To print file status using fstat
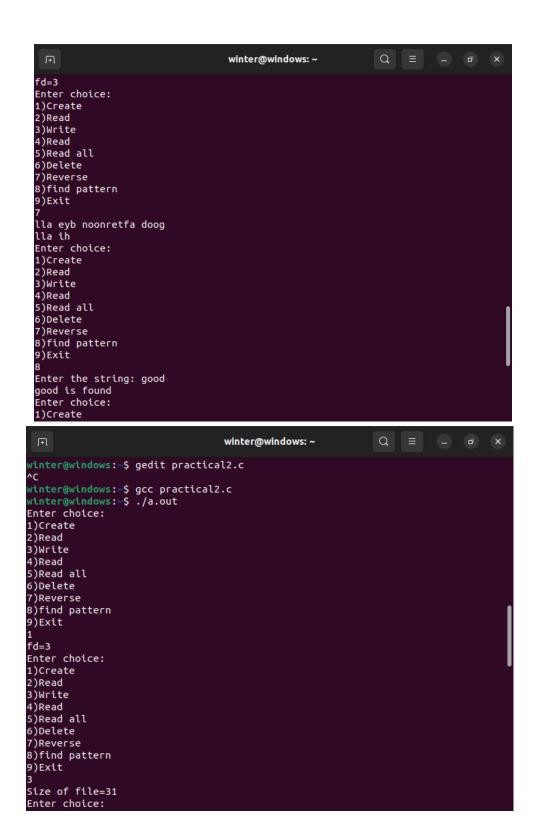
Using these system calls also write a program to,

1) Write a program that creates a file with a 4K bytes free space. Such files are called files with holes.

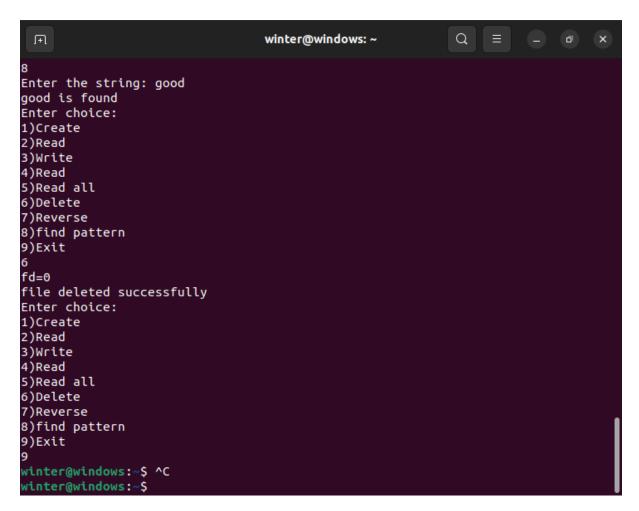2) Write a program that copies the contents of an existing file into another file. The names of the two

file should be read as an input from the command line.

PROGRAM AND OUTPUT –

```c
#include<fcntl.h>
#include<errno.h>
#include<unistd.h>
#include<string.h>
#include<sys/types.h>

#include<stdio.h>

#include<stdlib.h>
int main()
{
  int fd,c,n,n1;
  long sz,sz1;
  char c1[100],c2;
  char pattern[20],temp[100];
  while(1){
  printf("Enter choice: \n1)Create \n2)Read \n3)Write \n4)Read all \n5)Read all \n6)Delete
\n7)Reverse \n8)find pattern \n9)Exit \n");
  scanf("%d",&c);
  switch(c){
   case 1:
     fd=creat("f1.txt",0777);
     printf("fd=%d\n",fd);
     close(fd);
     break;
   case 2:
     fd=open("f1.txt",O_RDONLY|O_CREAT,0777);
     printf("fd=%d\n",fd);
     close(fd);
     break;
   case 3:
     fd=open("f1.txt",O_WRONLY|O_CREAT,0777);
     sz=write(fd,"hi all\n good afternoon bye all\n",strlen("hi all\n good afternoon bye all\n"));
     printf("Size of file=%ld\n",sz);
     close(fd);
     break;
   case 4:
     fd=open("f1.txt",O_RDONLY|O_CREAT,0777);
     sz=read(fd,c1,10);
     printf("Size of string read=%ld\n",sz);
     printf("fd=%d\n",fd);
     c1[sz]='\0';
     printf("%s\n",c1);
     close(fd);
     break;
   case 5:
     /*fd=open("f1.txt",O_RDONLY|O_CREAT,0777);
     do{
       sz=read(fd,c1,40);
     }while(sz!=-1);
```

```c
      printf("Size of string read=%ld\n",sz);
      printf("fd=%d\n",fd);
      c1[sz]='\0';
      printf("%s\n",c1);
      close(fd);*/
      fd=open("f1.txt",O_RDONLY|O_CREAT,0777);

      break;
    case 6:
      fd=unlink("f1.txt");
      printf("fd=%d\n",fd);
      printf("file deleted successfully\n");
      break;
    case 7:
      fd=open("f1.txt",O_RDONLY|O_CREAT,0777);
      n=lseek(fd,0,SEEK_END);
      for(int i=0;i<n;i++){
        sz1=read(fd,&c2,1);
        printf("%c",c2);
        n1=lseek(fd,-2,SEEK_CUR); //-2 is used as file pntr increments automatically so we
need to go back 2 pos every time
      }
      printf("\n");
      close(fd);
      break;
    case 8:
      printf("Enter the string: ");
      scanf("%s",pattern);
      fd=open("f1.txt",O_RDONLY,0777);
      n=lseek(fd,0,SEEK_END);
      n1=lseek(fd,0,SEEK_SET);
      read(fd,temp,n);
      if(strstr(temp,pattern)!=NULL)
        printf("%s is found\n",pattern);
      else
        printf("Pattern not found\n");
      close(fd);
      break;
    case 9:

      exit(0);
  }
```

```
fd=3
Enter choice:
1)Create
2)Read
3)Write
4)Read
5)Read all
6)Delete
7)Reverse
8)find pattern
9)Exit
7
lla eyb noonretfa doog
lla ih
Enter choice:
1)Create
2)Read
3)Write
4)Read
5)Read all
6)Delete
7)Reverse
8)find pattern
9)Exit
8
Enter the string: good
good is found
Enter choice:
1)Create
```

```
winter@windows:~$ gedit practical2.c
^C
winter@windows:~$ gcc practical2.c
winter@windows:~$ ./a.out
Enter choice:
1)Create
2)Read
3)Write
4)Read
5)Read all
6)Delete
7)Reverse
8)find pattern
9)Exit
1
fd=3
Enter choice:
1)Create
2)Read
3)Write
4)Read
5)Read all
6)Delete
7)Reverse
8)find pattern
9)Exit
3
Size of file=31
Enter choice:
```

```
                             winter@windows: ~        Q   ≡   —   ◻   ✕

8
Enter the string: good
good is found
Enter choice:
1)Create
2)Read
3)Write
4)Read
5)Read all
6)Delete
7)Reverse
8)find pattern
9)Exit
6
fd=0
file deleted successfully
Enter choice:
1)Create
2)Read
3)Write
4)Read
5)Read all
6)Delete
7)Reverse
8)find pattern
9)Exit
9
winter@windows:~$ ^C
winter@windows:~$
```

CONCLUSION - Programs to implement I/O System Calls of Linux has been executed.