```c
/*
Name - Atharva Paliwal
Experiment no.2 - Problem Statement 2
*/

#include<unistd.h>
#include<stdio.h>
#include<fcntl.h>
#include<stdlib.h>
#include <sys/types.h>

int main()
{
int fd,option,sz,temp,n;
char buffer[80];
char str[50]="Welcome";
char d,name[50];

do{
    printf("\nOperation?\n");
    printf("1.Create 2.Open 3.Read 4.Write 5.Close 6.lSeek 7.Unlink 8.reverse
    9.resource allocate\n");
    scanf("%d",&option);
    switch(option)
    {
    case 1: ;
                fd=open("ras.txt",O_RDONLY | O_CREAT);
                if(fd!=-1)
                {
                    printf("\nFile is created\n");
                }
            break;
    case 2: ; fd=open("am.txt",O_CREAT | O_RDWR);
                if(fd<0)
                    {
                        perror("r1");
                        exit(1);
                    }
                else
                    {
                        printf("File opened at %d",fd);
                    }
            break;
    case 3: ; sz=read(fd,buffer,sizeof(str));
                if(sz==-1 || sz==0)
                {
                printf("not allowed\n");
                }
                else{
                 printf("Called read.returned that"  " %d bytes were read.\n",sz);
                 buffer[sz]='\0';
                 printf("\nas follows:%s\n",str); }
            break;
    case 4: ;  fd=write(fd,str,sizeof(str));
                printf("Called write" "It returned %d\n",sz);
            break;
    case 5: ;  int close(int fd);
            break;
    case 6:if(lseek(fd,10,SEEK_SET) < 0)
            {
            printf("lseek implemented");
            return 1;
            }
            break;
    case 7:   unlink("ras.txt");
            break;

    case 8: printf("Enter the file\n");
        scanf("%s",name);
```

```
69          fd=open(name,O_RDONLY);
70          if(fd<0){
71              printf("error");}
72          n=lseek(fd,0,SEEK_END);
73          while(n>0)
74          {
75              read(fd,&d,1);
76              printf("%c",d);
77              lseek(fd,-2,SEEK_CUR);
78              n--;
79          }
80          break;
81
82
83       case 9:
84
85
86          temp=getpid();
87          printf("\n%d\n",temp);
88          char tt2[100];
89          sprintf(tt2,"ls /proc/%d/fd",temp);
90          system(tt2);
91          break;
92
93   default:printf("option invalid");
94                      break;
95      }
96      }while(option!=0);
97      }
98
```