Q1. Write a program using file operations that demonstrates copying of data from input file and write into output file, untill reaches end of file data.

|  |
| --- |
| //write a code for take input in one file and print the output in unother file    #include<stdio.h>  #include<fcntl.h>  #include<unistd.h>    int main()  {  int fd,fd1,len;  char buf[300];    fd = open("/home/ashruti/linux/Day-1/hello.txt",O\_RDONLY);    if(fd<0)  {  printf("File is not opened");  return 0;  }    len = read(fd,buf,300);  fd1 = open("/home/ashruti//linux/Day-1/output.txt",O\_CREAT|O\_WRONLY,0755);    if(fd<0)  {  printf("File is not opened");  return 0;  }    write(fd1,buf,len);  close(fd);  close(fd1);  return 0;  } |

Q2. Write a program that demonstrates repositioning of file offset using SEEK\_SET, SEEK\_CUR and SEEK\_END.

|  |
| --- |
| #include<stdio.h>  #include<fcntl.h>  #include<unistd.h>      int main()  {  int fd,len;  char write\_buf[50] = "hi, how are you!";  char read\_buf[50];    fd = open("linuxabc.txt", O\_CREAT| O\_RDWR , 777);  len = write(fd, write\_buf, 50);  printf("return value from write optn=%d\n",len);    lseek(fd,0,SEEK\_SET);  read(fd,read\_buf,len);  printf("data from read option =%s",read\_buf);    lseek(fd,0,SEEK\_END);  read(fd,read\_buf,len);  printf("\ndata from read option =%s",read\_buf);    lseek(fd,-5,SEEK\_CUR);  read(fd,read\_buf,len);  printf("\ndata from read option =%s",read\_buf);      close(fd);  return 0;  } |

Q3. Write program that returns “ls -l ” kind of structure of information from an existing file or opend file.

|  |
| --- |
| #include<stdio.h>  #include<fcntl.h>  #include<unistd.h>  #include<sys/stat.h>  #include<sys/types.h>    int main()  {    struct stat st,st1;  int fd;  stat("Q1.c",&st);  printf("File size: %lu\n",st.st\_size);  printf("File inode: %lu\n",st.st\_ino);    fd=open("que2.c",O\_RDONLY);  fstat(fd,&st1);  printf("File size: %lu\n",st1.st\_size);  printf("File inode: %lu\n",st1.st\_ino);    return 0;  } |

Q4. Write a program that implements all file operations(open/creat/write/read/lseek/close)

|  |
| --- |
| #include<stdio.h>  #include<stdlib.h>  #include<fcntl.h>  #include<unistd.h>    int main()  {  int fd,len;  char r\_buf[50];  creat("Q4.txt",0755);  char buf[50]="Q4 is all system calls in one program";  fd =open("Q4.txt",O\_WRONLY);  len=write(fd,buf,50);    lseek(fd,0,SEEK\_SET);  read(fd,r\_buf,len);  printf("data from read option =%s",r\_buf);    close(fd);    fd=open("Q4.txt",O\_RDONLY);    read(fd,r\_buf,len);  printf("%s\n",r\_buf);  close(fd);  return 0;  } |

Q5 . Write a program that creates a file with a 4K bytes free space. (Such files are called files with holes.)

|  |
| --- |
| // file with a 4k byte free space    #include<sys/types.h>  #include<sys/stat.h>  #include<fcntl.h>  #include<unistd.h>  #include<stdlib.h>    int main()  {  char buf1 = "LAB";  char buf2 = "OS Linux";  int fd;  if((fd =creat("file.gol" , 0666))<0)  {  error("creation error");  exit(1);  }    if(write(fd,buf1,sizeof(buf1)) < 0)  {  error("Writing error");  exit(2);  }    if(lseek(fd,4096, SEEK\_SET) < 0)  {  error("Positioning error");  exit(3);  }    if(write(fd,buf2,sizeof(buf2)) < 0)  {  error("Writing error");  exit(2);  }  } |