**LSEEK**

#include<unistd.h>

#include<fcntl.h>

#include<sys/types.h

#include<sys/stat.h>

int main()

{

int n,f;

char buff[10];

f=open("/root/seeking",O\_RDWR);

read(f,buff,10);

write(1,buff,10);

lseek(f,5,SEEK\_CUR);//skips 5 characters from the current position

read(f,buff,10);

write(1,buff,10);

}

create seeking.txt separately

**File Locking**

**This program will write lock a sample.txt file. As it is write lock, when try to execute the same program in another window, it wont lock. Only when the first program (first window) releases lock, the second window will apply lock**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <errno.h>

#include <fcntl.h>

#include <unistd.h>

int main()

{

char buf[100];

struct flock fl;

fl.l\_type = F\_WRLCK;

fl.l\_whence = SEEK\_SET;

fl.l\_start = 0;

fl.l\_len= 0;

fl.l\_pid = getpid();

printf("P1 Applying\nwrite lock");

int fd = open("sample.txt", O\_WRONLY|O\_APPEND);

if (fd == -1)

{

perror("Unable to open file: ");

exit(EXIT\_FAILURE);

}

printf("Press Enter to try to get lock -");

getchar();

printf("Locking...\n");

if (fcntl(fd, F\_SETLKW, &fl) == -1)

{

perror("fcntl caused some error: ");

exit(EXIT\_FAILURE);

}

printf("Locked\n");

strcpy(buf,"we are appending file\n");

if (write(fd,buf,strlen(buf))<0)

{

perror("Problem in writing to file");

exit(1);

}

printf("Press Enter to release lock -");

getchar();

fl.l\_type = F\_UNLCK;

if (fcntl(fd, F\_SETLK, &fl) == -1)

{

perror("fcntl caused some error: ");

exit(EXIT\_FAILURE);

}

printf("Unlocked.\n");

close(fd);

return 0;

}

==========================================================

**This program will read lock a sample.txt file. As it is read lock, when try to execute the same program in another window, it will lock. This is because two reader shall operate on file simultaneously**

include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <errno.h>

#include <fcntl.h>

#include <unistd.h>

int main()

{

char buf[100];

struct flock fl;

fl.l\_type = F\_RDLCK;

fl.l\_whence = SEEK\_SET;

fl.l\_start = 0;

fl.l\_len= 0;

fl.l\_pid = getpid();

printf("Applying\nread lock");

int fd = open("sample.txt", O\_RDONLY);

if (fd == -1)

{

perror("Unable to open file: ");

exit(EXIT\_FAILURE);

}

printf("Press Enter to try to get lock -");

getchar();

printf("Locking...\n");

if (fcntl(fd, F\_SETLKW, &fl) == -1)

{

perror("fcntl caused some error: ");

exit(EXIT\_FAILURE);

}

printf("Locked\n");

int flag=read(fd,buf,100);

if(flag<0){

printf("Error occured while reading\n");

exit(1);

}

printf("Read file content:\n");

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

printf("Press Enter to release lock -");

getchar();

fl.l\_type = F\_UNLCK;

if (fcntl(fd, F\_SETLK, &fl) == -1)

{

perror("fcntl caused some error: ");

exit(EXIT\_FAILURE);

}

printf("Unlocked.\n");

close(fd);

return 0;

}