EXP 20

```
#include<stdio.h>
#include<pthread.h>
#include<semaphore.h>

sem_t mutex;
sem_t db;
int readercount=0;
pthread_t reader1,reader2,writer1,writer2;
void *reader(void *);
void *writer(void *);
main()
{
sem_init(&mutex,0,1);
sem_init(&db,0,1);
while(1)
{
pthread_create(&reader1,NULL,reader,"1");
pthread_create(&reader2,NULL,reader,"2");
pthread_create(&writer1,NULL,writer,"1");
pthread_create(&writer2,NULL,writer,"2");
}
}
void *reader(void *p)
{
printf("prevoius value %dn",mutex);
sem_wait(&mutex);
printf("Mutex acquired by reader %dn",mutex);
readercount++;
if(readercount==1) sem_wait(&db);
sem_post(&mutex);
```

```
printf("Mutex returned by reader %dn",mutex);
printf("Reader %s is Readingn",p);
//sleep(3);
sem_wait(&mutex);
printf("Reader %s Completed Readingn",p);
readercount--;
if(readercount==0) sem_post(&db);
sem_post(&mutex);
}

void *writer(void *p)
{
printf("Writer is Waiting n");
sem_wait(&db);
printf("Writer %s is writingn ",p);
sem_post(&db);
//sleep(2);
}
```