

```

#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>
#include <semaphore.h>
#include <stdint.h>
#include <unistd.h>

#define R 5
#define W 5
int readcount;
int writecount;
//sem_t x;
//sem_t y,z;
pthread_mutex_t x;
sem_t wsem,y; //mutual exclusion
int s=5;
void *reader1(void *a);
void *writer1(void *a);
//void *reader2(void *a);
//void *writer2(void *a);
int main()
{
    int i,op;

    pthread_t thread_read[R],thread_write[W];
    //sem_init(&x,0,1);
    pthread_mutex_init(&x,NULL);           //initialized to default value
    sem_init(&wsem,0,1);
    sem_init(&y,0,1);
    // sem_init(&rsem,0,1);
    // sem_init(&z,0,1);

    do{
        printf("Menu : 1. Readers have priority 2. exit ");
        scanf("%d",&op);
        switch(op)
        {
            case 1:readcount=0;
                for(i=0;i<W;i++)
                {
                    pthread_create(&thread_write[i],NULL, *writer1,(void *) (intptr_t) i);
                }
                for(i=0;i<R;i++)
                {
                    pthread_create(&thread_read[i],NULL, *reader1,(void *) (intptr_t) i);
                }

```

```

        for(i=0;i<W;i++)
        {
            pthread_join(thread_write[i],NULL);
        }
        for(i=0;i<R;i++)
        {
            pthread_join(thread_read[i],NULL);
        }
        break;
/*    case 2:readcount=0;
        writecount = 0;
        for(i=0;i<W;i++)
        {
            pthread_create(&thread_write[i],NULL, *writer2,(void *) (intptr_t) i);
        }
        for(i=0;i<R;i++)
        {
            pthread_create(&thread_read[i],NULL, *reader2,(void *) (intptr_t) i);
        }
        for(i=0;i<W;i++)
        {
            pthread_join(thread_write[i],NULL);
        }
        for(i=0;i<R;i++)
        {
            pthread_join(thread_read[i],NULL);
        }
        break;*/

    case 2: break;
}

}while(op!=2);
}

```

```

void *reader1(void *a)
{
    //int r=(int)a;
    int r = (intptr_t) a;
    int i=0;
    while (i<5){
        //sleep(rand() % 10);
        //sem_wait(&x);
        pthread_mutex_lock(&x);
        readcount++;
        if(readcount == 1)
            sem_wait(&wsem);
        //sem_post(&x);
        pthread_mutex_unlock(&x);
        printf("\t\tReader %d is reading : %d \n",r,s);
        //sem_wait(&x);
    }
}

```

```

        pthread_mutex_lock(&x);
        readcount--;
        if(readcount == 0)
            sem_post(&wsem);
        //sem_post(&x);
        pthread_mutex_unlock(&x);
        sleep(rand() % 10);
        i++;
    }
}

void *writer1(void *a)
{
    int w = (intptr_t) a;
    //int w=(int)a;
    int i=0;
    while (i<2){
        //sleep(rand() % 10);
        sem_wait(&wsem);
        s+=5;
        printf("Writer %d is writing : %d \n",w,s);
        sem_post(&wsem);
        sleep(rand() % 10);
        i++;
    }
}

```

OUTPUT

Menu : 1. Readers have priority 2. exit 1

Writer 1 is writing : 10

Writer 0 is writing : 15

Writer 3 is writing : 20

Writer 4 is writing : 25

Reader 0 is reading : 25

Reader 1 is reading : 25

Reader 2 is reading : 25

Reader 3 is reading : 25

Reader 4 is reading : 25

Writer 2 is writing : 30

Writer 2 is writing : 35

Reader 3 is reading : 35

Writer 1 is writing : 40

Reader 0 is reading : 40

Writer 4 is writing : 45

Reader 1 is reading : 45

Writer 0 is writing : 50

Reader 2 is reading : 50

Writer 3 is writing : 55

Reader 4 is reading : 55

Reader 3 is reading : 55

Reader 3 is reading : 55

Reader 1 is reading : 55

Reader 0 is reading : 55

Reader 2 is reading : 55

Reader 2 is reading : 55

Reader 2 is reading : 55

Reader 4 is reading : 55

Reader 4 is reading : 55

Reader 1 is reading : 55

Reader 3 is reading : 55

Reader 0 is reading : 55

Reader 0 is reading : 55

Reader 1 is reading : 55

Reader 4 is reading : 55

Menu : 1. Readers have priority 2. exit