```
[] G & Share
       main.c
                                                                                           Output
       1 #include <stdio.h>
                                                                                           Reader 1 reads 0
       2 #include <pthread.h>
                                                                                           Reader 2 reads 0
                                                                                           Writer 1 writes 1
       3 #include <semaphore.h>
       4 #include <unistd.h>
                                                                                           Reader 1 reads 1
       5 sem_t wrt; int readcount=0; pthread_mutex_t mutex; int data=0;
                                                                                           Reader 2 reads 1
       6 void* reader(void* id){while(1){pthread_mutex_lock(&mutex); readcount++;
                                                                                           Writer 1 writes 2
5
        7 if(readcount==1) sem_wait(&wrt); pthread_mutex_unlock(&mutex);
                                                                                           Reader 1 reads 2
       8 printf("Reader %ld reads %d\n",(long)id,data); sleep(1);
                                                                                           Reader 2 reads 2
       9 pthread_mutex_lock(&mutex); readcount--; if(readcount==0) sem_post(&wrt);
                                                                                           Writer 1 writes 3
       10 pthread_mutex_unlock(&mutex); sleep(1);}return NULL;}
                                                                                           Reader 1 reads 3
0
      11 void* writer(void* id){while(1){sem_wait(&wrt); data++;
                                                                                           Reader 2 reads 3
      12 printf("Writer %ld writes %d\n",(long)id,data); sem_post(&wrt); sleep(2);}return Writer 1 writes 4
              NULL;}
                                                                                           Reader 1 reads 4
      13 int main(){pthread_t r1,r2,w1; sem_init(&wrt,0,1); pthread_mutex_init(&mutex,NULL
                                                                                           Reader 2 reads 4
                                                                                           Writer 1 writes 5
0
      14 pthread_create(&r1,NULL,reader,(void*)1); pthread_create(&r2,NULL,reader,(void*)2 Reader 1 reads 5
                                                                                           Reader 2 reads 5
JS
      15 pthread_create(&w1,NULL,writer,(void*)1); pthread_join(r1,NULL); pthread_join(r2
                                                                                           Writer 1 writes 6
              ,NULL); pthread_join(w1,NULL);}
                                                                                           Reader 1 reads 6
TS
                                                                                           Reader 2 reads 6
                                                                                           Writer 1 writes 7
                                                                                           Reader 1 reads 7
```