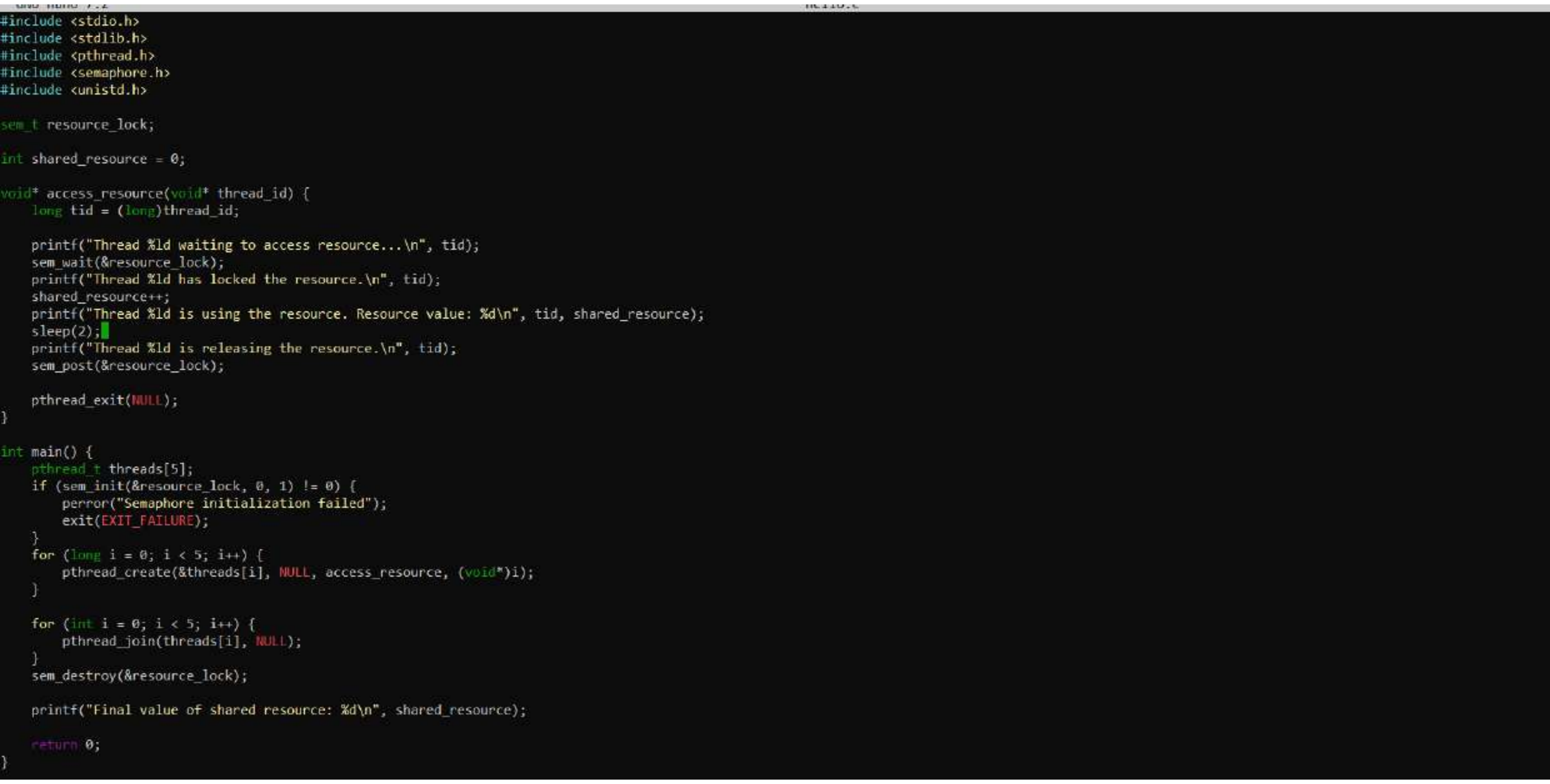
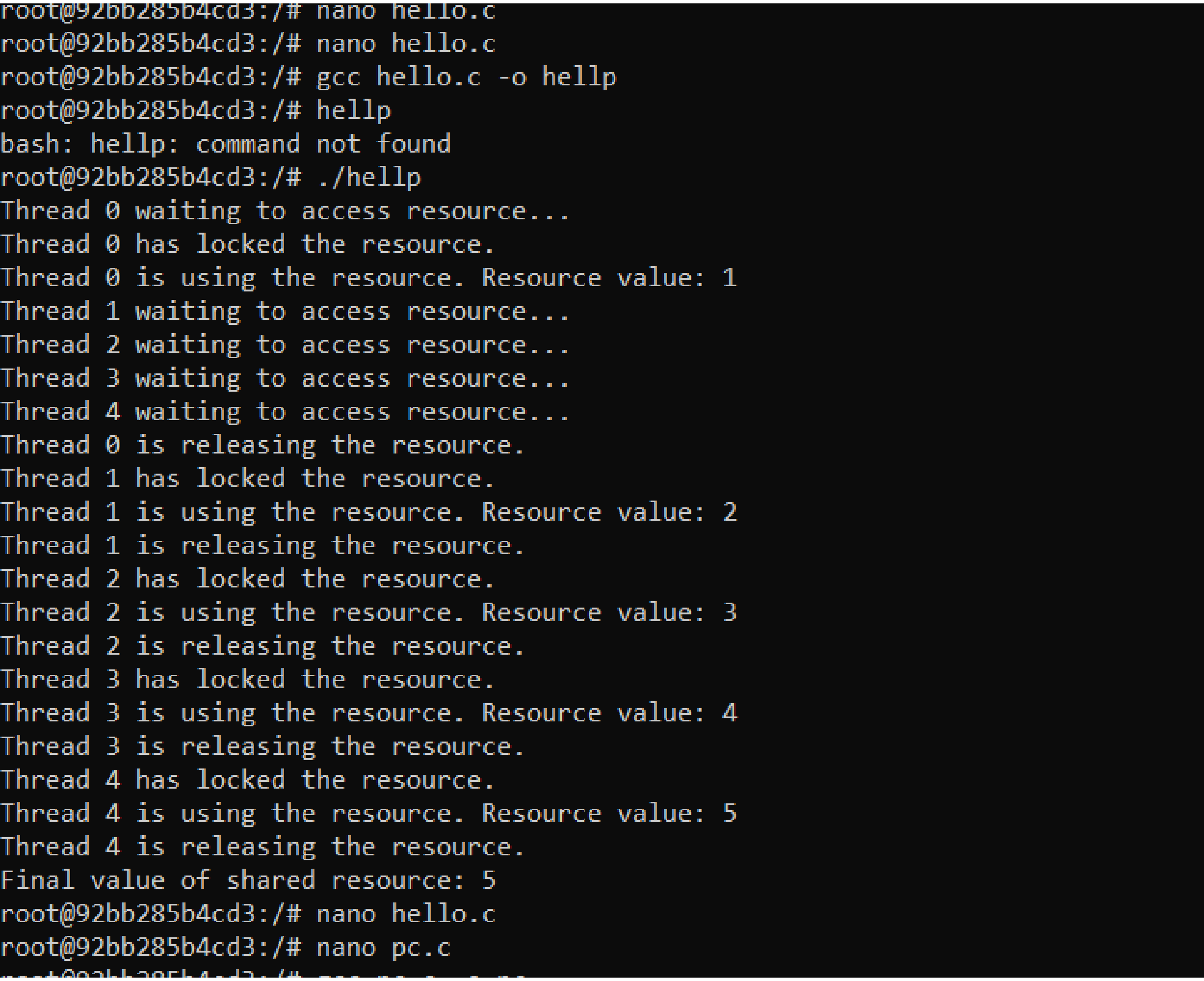
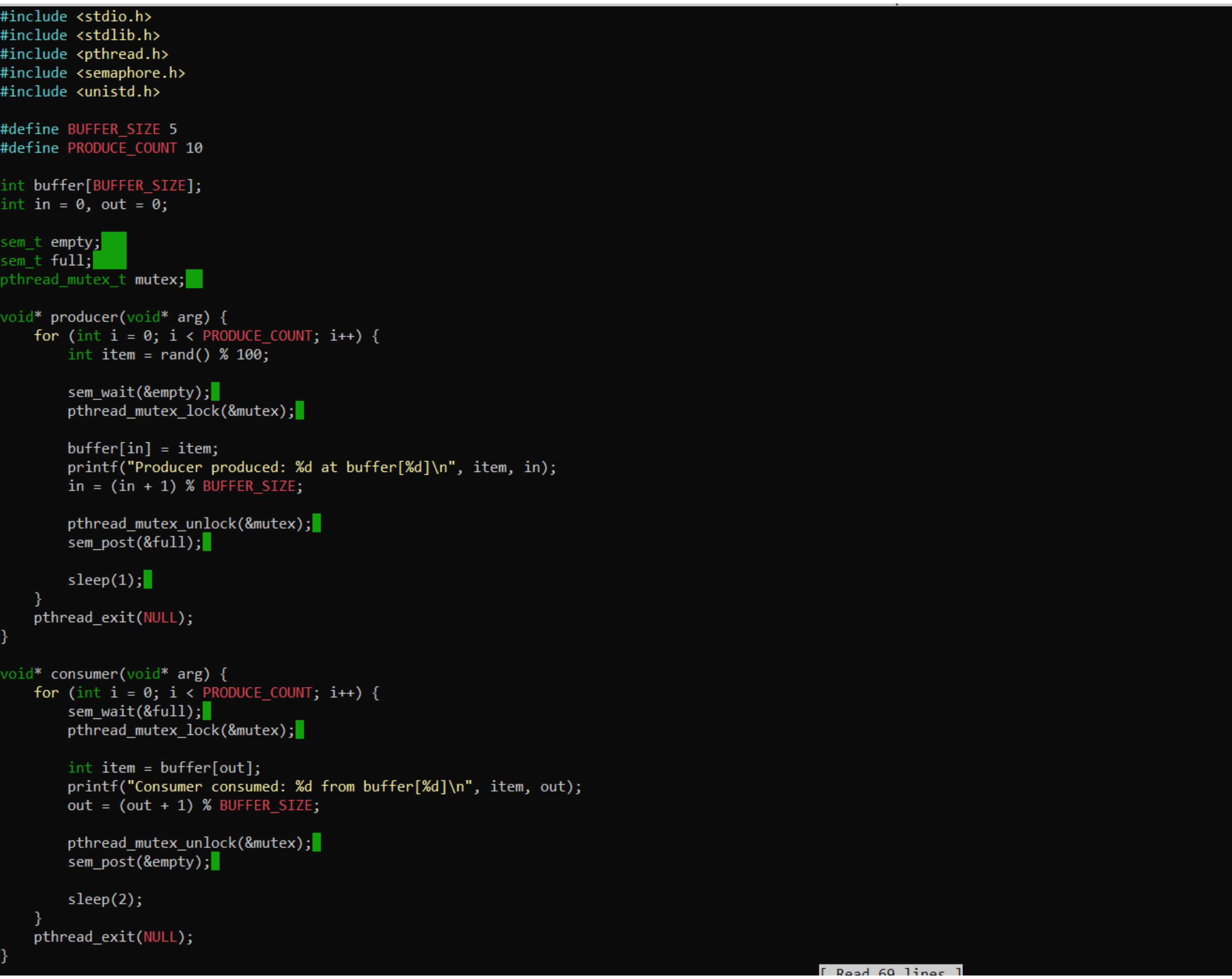


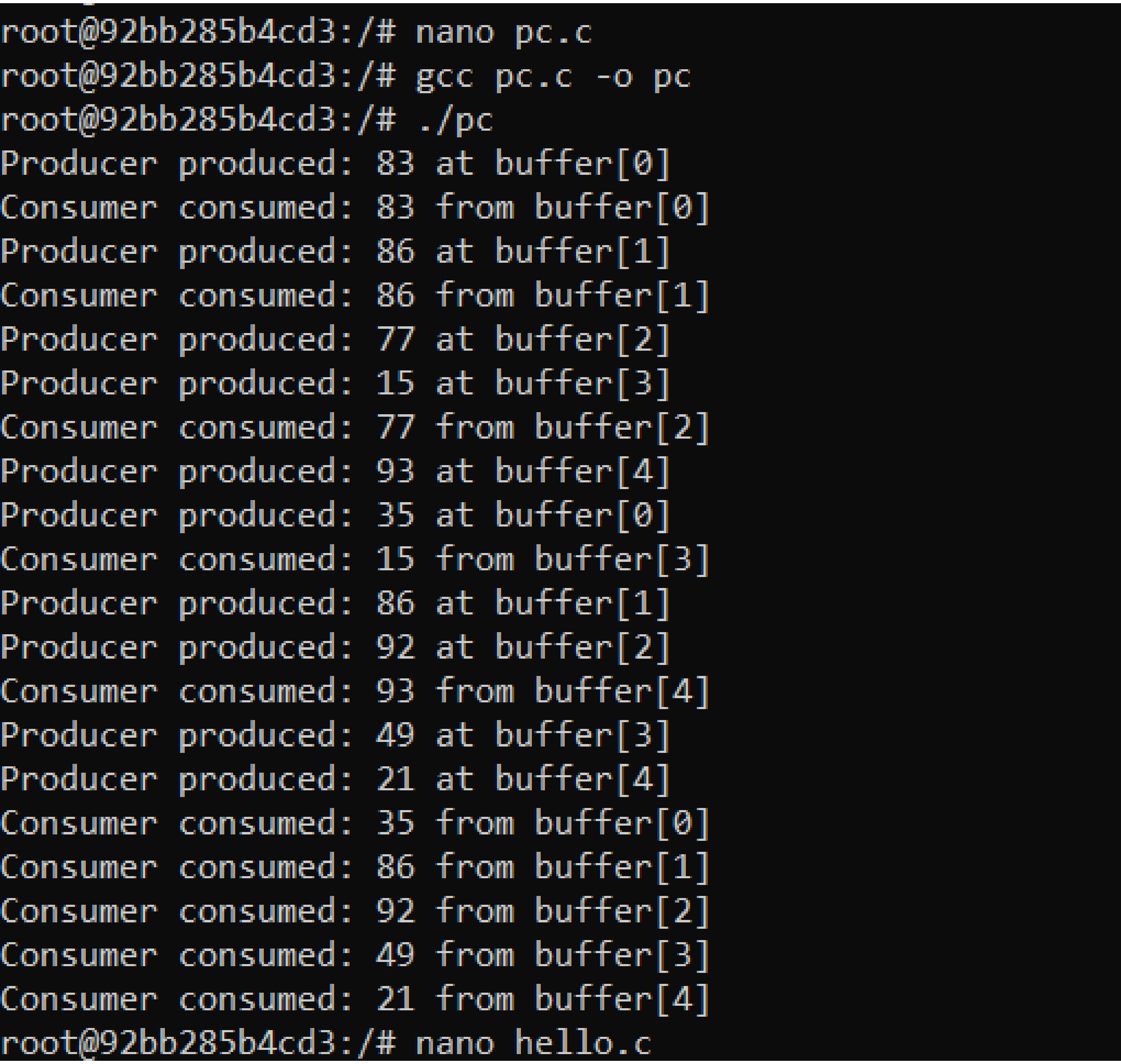
```c
#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>
#include <semaphore.h>
#include <unistd.h>

sem_t resource_lock;

int shared_resource = 0;

void* access_resource(void* thread_id) {
    long tid = (long)thread_id;

    printf("Thread %ld waiting to access resource...\n", tid);
    sem_wait(&resource_lock);
    printf("Thread %ld has locked the resource.\n", tid);
    shared_resource++;
    printf("Thread %ld is using the resource. Resource value: %d\n", tid, shared_resource);
    sleep(2);
    printf("Thread %ld is releasing the resource.\n", tid);
    sem_post(&resource_lock);

    pthread_exit(NULL);
}

int main() {
    pthread_t threads[5];
    if (sem_init(&resource_lock, 0, 1) != 0) {
        perror("Semaphore initialization failed");
        exit(EXIT_FAILURE);
    }
    for (long i = 0; i < 5; i++) {
        pthread_create(&threads[i], NULL, access_resource, (void*)i);
    }

    for (int i = 0; i < 5; i++) {
        pthread_join(threads[i], NULL);
    }
    sem_destroy(&resource_lock);

    printf("Final value of shared resource: %d\n", shared_resource);

    return 0;
}
```



```
root@92bb285b4cd3:/# nano hello.c
root@92bb285b4cd3:/# nano hello.c
root@92bb285b4cd3:/# gcc hello.c -o hellp
root@92bb285b4cd3:/# hellp
bash: hellp: command not found
root@92bb285b4cd3:/# ./hellp
Thread 0 waiting to access resource...
Thread 0 has locked the resource.
Thread 0 is using the resource. Resource value: 1
Thread 1 waiting to access resource...
Thread 2 waiting to access resource...
Thread 3 waiting to access resource...
Thread 4 waiting to access resource...
Thread 0 is releasing the resource.
Thread 1 has locked the resource.
Thread 1 is using the resource. Resource value: 2
Thread 1 is releasing the resource.
Thread 2 has locked the resource.
Thread 2 is using the resource. Resource value: 3
Thread 2 is releasing the resource.
Thread 3 has locked the resource.
Thread 3 is using the resource. Resource value: 4
Thread 3 is releasing the resource.
Thread 4 has locked the resource.
Thread 4 is using the resource. Resource value: 5
Thread 4 is releasing the resource.
Final value of shared resource: 5
root@92bb285b4cd3:/# nano hello.c
root@92bb285b4cd3:/# nano pc.c
root@92bb285b4cd3:/# gcc pc.c -o pc
```



```c
#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>
#include <semaphore.h>
#include <unistd.h>

#define BUFFER_SIZE 5
#define PRODUCE_COUNT 10

int buffer[BUFFER_SIZE];
int in = 0, out = 0;

sem_t empty;
sem_t full;
pthread_mutex_t mutex;

void* producer(void* arg) {
    for (int i = 0; i < PRODUCE_COUNT; i++) {
        int item = rand() % 100;

        sem_wait(&empty);
        pthread_mutex_lock(&mutex);

        buffer[in] = item;
        printf("Producer produced: %d at buffer[%d]\n", item, in);
        in = (in + 1) % BUFFER_SIZE;

        pthread_mutex_unlock(&mutex);
        sem_post(&full);

        sleep(1);
    }
    pthread_exit(NULL);
}

void* consumer(void* arg) {
    for (int i = 0; i < PRODUCE_COUNT; i++) {
        sem_wait(&full);
        pthread_mutex_lock(&mutex);

        int item = buffer[out];
        printf("Consumer consumed: %d from buffer[%d]\n", item, out);
        out = (out + 1) % BUFFER_SIZE;

        pthread_mutex_unlock(&mutex);
        sem_post(&empty);

        sleep(2);
    }
    pthread_exit(NULL);
}
```



```
root@92bb285b4cd3:/# nano pc.c
root@92bb285b4cd3:/# gcc pc.c -o pc
root@92bb285b4cd3:/# ./pc
Producer produced: 83 at buffer[0]
Consumer consumed: 83 from buffer[0]
Producer produced: 86 at buffer[1]
Consumer consumed: 86 from buffer[1]
Producer produced: 77 at buffer[2]
Producer produced: 15 at buffer[3]
Consumer consumed: 77 from buffer[2]
Producer produced: 93 at buffer[4]
Producer produced: 35 at buffer[0]
Consumer consumed: 15 from buffer[3]
Producer produced: 86 at buffer[1]
Producer produced: 92 at buffer[2]
Consumer consumed: 93 from buffer[4]
Producer produced: 49 at buffer[3]
Producer produced: 21 at buffer[4]
Consumer consumed: 35 from buffer[0]
Consumer consumed: 86 from buffer[1]
Consumer consumed: 92 from buffer[2]
Consumer consumed: 49 from buffer[3]
Consumer consumed: 21 from buffer[4]
root@92bb285b4cd3:/# nano hello.c
```