

h. Producer Consumer problem

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

#define SIZE 5

int buffer[SIZE];
int in = 0, out = 0;
sem_t empty, full, mutex;

void* producer(void* arg) {
    int item = 1;
    while (1) {
        sem_wait(&empty);
        sem_wait(&mutex);

        buffer[in] = item;
        printf("Produced: %d\n", item);
        in = (in + 1) % SIZE;
        item++;

        sem_post(&mutex);
        sem_post(&full);
        sleep(1);
    }
}

void* consumer(void* arg) {
    while (1) {
        sem_wait(&full);
        sem_wait(&mutex);

        int item = buffer[out];
        printf("Consumed: %d\n", item);
        out = (out + 1) % SIZE;

        sem_post(&mutex);
        sem_post(&empty);
        sleep(2);
    }
}
```

```

}

int main() {
    pthread_t prod, cons;

    sem_init(&empty, 0, SIZE);
    sem_init(&full, 0, 0);
    sem_init(&mutex, 0, 1);

    pthread_create(&prod, NULL, producer, NULL);
    pthread_create(&cons, NULL, consumer, NULL);

    pthread_join(prod, NULL);
    pthread_join(cons, NULL);

    return 0;
}

```

```

Produced: 1
Produced: 2
Produced: 3
Consumed: 1
Produced: 4
Consumed: 2
Produced: 5
Consumed: 3
Produced: 6
Consumed: 4
Produced: 7
Consumed: 5
...

```