Code 10

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

#define SIZE 5
int buffer[SIZE], 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("Producer produced: %d\n", item++);
      in = (in+1)%SIZE;
      sem_post(&mutex);
      sem_post(&full);
      sleep(1);
   }
}

void *consumer(void *arg) {
   int item;
   while(1) {
      sem_wait(&full);
```

```
        sem_wait(&mutex);
        item = buffer[out];
        printf("Consumer consumed: %d\n", item);
        out = (out+1)%SIZE;
        sem_post(&mutex);
        sem_post(&empty);
        sleep(2);
    }
}

int main() {
    pthread_t p, c;
    sem_init(&empty,0,SIZE);
    sem_init(&full,0,0);
    sem_init(&mutex,0,1);

    pthread_create(&p,NULL,producer,NULL);
    pthread_create(&c,NULL,consumer,NULL);

    pthread_join(p,NULL);
    pthread_join(c,NULL);

    return 0;
}
```