20188068 孔天欣 180235

2．写代码实现生产者消费者问题

```java
import java.util.concurrent.Semaphore;

public class Cache {

    Semaphore mutex;
    Semaphore full;
    Semaphore empty;
    int size;
    int[] nums;
    int fill = 0;
    int use = 0;

    public Cache(int size) {
        mutex = new Semaphore(1);
        full = new Semaphore(0);
        empty = new Semaphore(size);
        this.size = size;
        nums = new int[size];
    }

    // 生产方法
    public void produce(int num) throws InterruptedException {
        empty.acquire();
        mutex.acquire();
        nums[fill] = num;
        fill = (fill + 1) % size;
        mutex.release();
        full.release();
    }

    // 消费方法
    public int consume() throws InterruptedException {
        full.acquire();
        mutex.acquire();
        int res = nums[use];
        use = (use + 1) % size;
        mutex.release();
        empty.release();
        return res;
    }

    public static void main(String[] args) {

        int cacheSize = 100;

        // 缓存队列
        Cache cache = new Cache(cacheSize);

        // 消费者
        Thread consumer = new Thread(() -> {
            try {
                while (true) {
                    int num = cache.consume();
                    System.out.println(num);
                }
            } catch (InterruptedException ignored) {
            }
        });

        // 生产者
        Thread producer = new Thread(() -> {
            for (int i = 0; i < 1000; i++) {
                try {
                    cache.produce(i);
                } catch (InterruptedException ignored) {
                }
            }
        });

        consumer.start();
        producer.start();
    }


}
```