

# 处理机管理——生产&消费模型 设计方案

1851201 周子龙

## 一、 生产&消费模型的提出

生产消费模型是一个在进程同步的问题，最早由荷兰计算机科学家 Dijkstra 在阐述 semaphore 机制的时候提出。其背景为：一群消费者和生产者共享一块共享区，生产者生产物品，并将其放置在共享区中，消费者从共享区中消费物品。如果共享区没有物品供消费者消费，那么当消费者发出消费请求的时候必须要等待生产者生产物品，同样的，如果生产者生产时共享区已经放满，则必须要等待消费者消费，以清出空间给生产者存放。如何管理生产者和消费者之间的关系，使得二者不会发生冲突，造成数据丢失等情况便十分重要。

## 二、 生产&消费模型设计约束

1. 有界缓冲区内设有 N 个存储单元（整型数）；
2. 每个生产者和消费者对有界缓冲区进行操作后，即时显示有界缓冲区的全部内容，当前指针位置和生产者/消费者进程的标识符；
3. 生产者和消费者各有两个以上；
4. 多个生产者或多个消费者之间须有共享对缓冲区进行操作的函数代码。

## 三、 问题分析

### 1. 临界区的确定

在本模型中，临界区即消费者和生产者代码中对共享区（有界缓冲区）发生访问的代码段。

### 2. 制约条件的确定

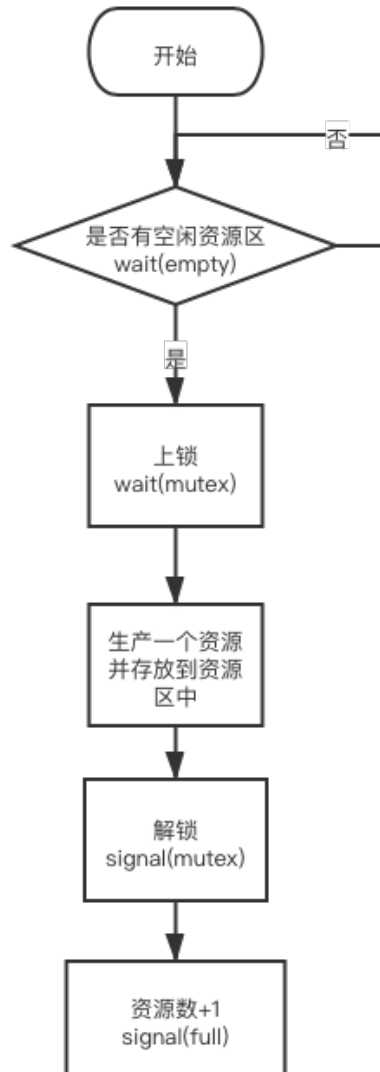
- a) 所有对共享区的访问均为互斥的。共享区每次只能供一个生产者或者消费者访问，以防止并发访问造成的不可预知的错误。
- b) 当共享区为空时，停止消费者对共享区的访问，直到某个生产者生产了一个资源。
- c) 当共享区为满时，停止生产者对共享区的访问，直到某个消费者消费了一个资源。

### 3. 信号量的确定

- a) full: 资源信号量, 判断共享区是否有资源, 初始值为 0。
- b) empty: 资源信号量, 判断共享区是否有空间, 初始值为 N。
- c) mutex: 互斥信号量, 用于对共享区的互斥访问, 初始值为 1。

#### 四、 生产&消费的逻辑

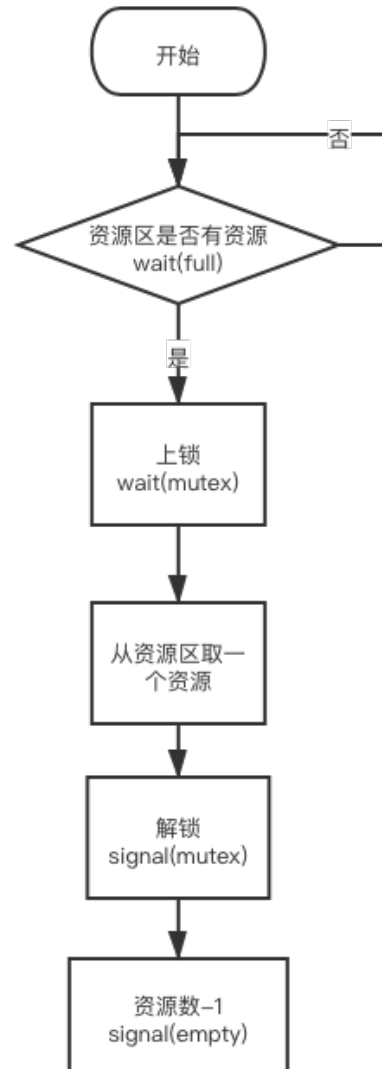
##### 1. 生产者



使用代码描述为:

```
wait(empty)
wait(mutex)
//produce something
signal(mutex)
signal(full)
```

## 2. 消费者



使用代码描述为：

```
wait(full)
wait(mutex)
//consume something
signal(mutex)
signal(empty)
```

## 五、 实现

### 1. 共享区的实现

在本问题中，共享区大小有上限，为  $N$  个。我们设置一个循环数组队列来充当共享区，并有两个指针指向队列的头和尾。

### 2. 各个进程的实现

首先，我们在系统堆区初始化各个信号量，然后进入 `main` 函数，进行对各个子进程（生产者&消费者）的创建。共享区的数量和生产者、消费者的数量由宏定义实现，可以方便的调整数量。在每一次生产、消费活动后，将在子进程中打印共享区的状态，包括：各个区域的资源状态、队列指针的状态。

此外，在本问题中，我们假设生产者和消费者的生产、消费时间均为  $2s$ 。

## 六、 运行

本程序的执行情况如下图：

```
zhouzilong — 01处理机管理_生产&消费_1851201_周子龙_执行程序 — -bash — 80...
shread areas:20, consumers:2, producer:5
produced item, value:1
buffer status:
current in-->1 out-->0
No.0 No.1 No.2 No.3 No.4 No.5 No.6 No.7 No.8 No.9
1 0 0 0 0 0 0 0 0 0
No.10 No.11 No.12 No.13 No.14 No.15 No.16 No.17 No.18 No.19
0 0 0 0 0 0 0 0 0 0

consumed item, value:1
buffer status:
current in-->1 out-->1
No.0 No.1 No.2 No.3 No.4 No.5 No.6 No.7 No.8 No.9
1 0 0 0 0 0 0 0 0 0
No.10 No.11 No.12 No.13 No.14 No.15 No.16 No.17 No.18 No.19
0 0 0 0 0 0 0 0 0 0

produced item, value:2
buffer status:
current in-->2 out-->1
No.0 No.1 No.2 No.3 No.4 No.5 No.6 No.7 No.8 No.9
1 2 0 0 0 0 0 0 0 0
No.10 No.11 No.12 No.13 No.14 No.15 No.16 No.17 No.18 No.19
0 0 0 0 0 0 0 0 0 0

consumed item, value:2
buffer status:
current in-->2 out-->2
No.0 No.1 No.2 No.3 No.4 No.5 No.6 No.7 No.8 No.9
1 2 0 0 0 0 0 0 0 0
No.10 No.11 No.12 No.13 No.14 No.15 No.16 No.17 No.18 No.19
0 0 0 0 0 0 0 0 0 0

produced item, value:3
buffer status:
current in-->3 out-->2
```

## 七、 边界性测试

### 1. 只有生产者，没有消费者

预期实现：共享区满后不继续生产，当线程时间使用完后程序正常结束。

运行结果：

```
zhouzilong — OS — 80x24
Last login: Tue May 12 16:34:35 on ttys002
zhouzilongdeMacBook-Pro:~ zhouzilong$ /Users/zhouzilong/Library/Developer/Xcode/
DerivedData/Build/Products/Debug/OS ; exit;
shread areas:1, consumers:0, producer:1
produced item, value:1
buffer status:
current in-->0 out-->0
No.0
1

logout
Saving session...
...copying shared history...
...saving history...truncating history files...
...completed.
Deleting expired sessions...12 completed.

[进程已完成]
```

## 2. 只有消费者，没有生产者

预期实现：消费者无法消费，当线程时间使用完后程序正常结束。

运行结果：

```
zhouzilong — OS — 80x24
Last login: Tue May 12 16:37:13 on ttys001
zhouzilongdeMacBook-Pro:~ zhouzilong$ /Users/zhouzilong/Library/Developer/Xcode/
DerivedData/Build/Products/Debug/OS ; exit;
shread areas:1, consumers:1, producer:0
logout
Saving session...
...copying shared history...
...saving history...truncating history files...
...completed.

[进程已完成]
```

## 3. 生产者数量大于消费者

预期实现：当共享区满后，生产者停止生产，等待消费者消费，当有空间释放后继续生产。

运行结果：

```
zhouzilong — OS — -bash — 58x42
shread areas:2, consumers:1, producer:2
produced item, value:1
buffer status:
current in-->1 out-->0
No.0    No.1
1       0

consumed item, value:1
buffer status:
current in-->1 out-->1
No.0    No.1
1       0

produced item, value:2
buffer status:
current in-->0 out-->1
No.0    No.1
1       2

produced item, value:3
buffer status:
current in-->1 out-->0
No.0    No.1
3       2

consumed item, value:2
buffer status:
current in-->1 out-->0
No.0    No.1
3       2

produced item, value:4
buffer status:
current in-->0 out-->0
No.0    No.1
3       4

^Z
[1]+  Stopped                  /Users/zhouzilong/Library/De
veloper/Xcode/DerivedData/Build/Products/Debug/OS
logout
There are stopped jobs.
```

#### 4. 消费者数目大于生产者

预期实现：当共享区空后，消费者停止消费，等待生产者生产，当有



资源后继续消费。

运行结果:

```
zhouzilong — OS — -bash — 58x42
shread areas:2, consumers:2, producer:1
produced item, value:1
buffer status:
current in-->1 out-->0
No.0    No.1
1       0

consumed item, value:1
buffer status:
current in-->1 out-->1
No.0    No.1
1       0

produced item, value:2
buffer status:
current in-->0 out-->1
No.0    No.1
1       2

consumed item, value:2
buffer status:
current in-->0 out-->0
No.0    No.1
1       2

produced item, value:3
buffer status:
current in-->1 out-->0
No.0    No.1
3       2

consumed item, value:3
buffer status:
current in-->1 out-->1
No.0    No.1
3       2

produced item, value:4
buffer status:
current in-->0 out-->1
No.0    No.1
3       4
```