

生产者消费者问题实验报告

一、 问题描述

通过编写**多进程程序**实现典型的生产者和消费者问题。实验要求如下：

- 完成 Windows 版本和 Linux 版本
- 一个大小为 3 的缓冲区，初始为空
- 2 个生产者
 - 随机等待一段时间，往缓冲区添加数据，
 - 若缓冲区已满，等待消费者取走数据后再添加
 - 重复 6 次
- 3 个消费者
 - 随机等待一段时间，从缓冲区读取数据
 - 若缓冲区为空，等待生产者添加数据后再读取
 - 重复 4 次
- 显示每次添加和读取数据的时间及缓冲区的状态

二、 实验过程

实验中共设置了三种信号量，分别为：

- 空闲缓冲区个数：idle，初值为 3
- 产品个数：product，初值为 0
- 互斥访问缓冲区（互斥信号量）：mutex，初值为 1

本实验共分三个程序，分别为 MainProgram、Producer 和 Customer。

MainProgram:

- 创建信号量
- 创建互斥体
- 创建共享内存区
- 创建生产者进程
- 创建消费者进程

Producer:

- 申请空闲缓冲区
- 申请互斥信号量
- 往缓冲区存放数据
- 释放产品个数
- 释放互斥信号量

Customer:

- 申请产品个数
- 申请互斥信号量
- 从缓冲区取出数据
- 释放空闲缓冲区
- 释放互斥信号量

以上为实验的整体结构，下面说明一些实验细节。

共享文件使用了 20 个字节的空間，即 5 个 int 型数据，其中第一个数据负责存放生产者的索引，即当前应该往哪一个缓冲区放数据，第二个数据为消费者索引，即应该从哪个缓冲区取数据，最后三个数据空间为缓冲区。

生产者进程在申请空闲缓冲区之前，会先生成一个随机的等待时间，并打印到屏幕上以进行观察，成功申请到两个信号量进入到放数据的步骤后，会先生成一个 100 以内的随机数，并将这个随机数作为数据存放到缓冲区中，在存放前后会分别打印当时缓冲区的状态，存放时会打印当前的索引值，进程 ID 和数据大小。

消费者进程与生产者进程类似，但是有一些不同。消费者进程不生成数据，它会根据索引值从相应的缓冲区中取出数据，并将该缓冲区置为 0，同样也会在取数据的前后分别打印缓冲区状态，取数据时会打印索引值，进程 ID 和数据大小。

在 Windows 和 Linux 环境下的程序基本相同，只是在进行系统调用时的调用方式不同，比如 Windows 环境下申请和释放信号量的过程可以分别用 WaitForSingleObject 和 ReleaseMutex/ReleaseSemaphore 一个函数解决，在 Linux 环境下则需要自己设置好参数，在调用函数根据设置的参数修改信号量。以及 Linux 环境下并没有互斥体一说（我看书后是这么觉得的，如果说错了请见谅），是将互斥体看成大小为 1 的信号量。

其余的区别就是在不同环境下对于同样的功能调用方式的不同，如 Windows 下的 CreateProcess 和 Linux 下的 fork，以及在创建和初始化信号量与共享主存时的一些方式上的差别（我觉得最主要的差别就是 Windows 的系统调用所需要的参数比 Linux 需要的多很多）。

三、 实验结果

Windows

```
选择C:\WINDOWS\system32\cmd.exe
-----
Product_Space:0 0 0
-----
Product_Space:0 0 0
Customer:5468 WaitTime:0ms
-----
Producer:10328 put the data:45
0
Product_Space:45 0 0
-----
Product_Space:45 0 0
Producer:10328 exit
-----
0
Customer:10720 take the data:45
-----
Product_Space:0 0 0
-----
Customer:10720 WaitTime:0ms
-----
Product_Space:0 0 0
-----
Producer:12172 put the data:64
1
Product_Space:0 64 0
-----
选择C:\WINDOWS\system32\cmd.exe
-----
Product_Space:0 64 0
Producer:12172 WaitTime:0ms
-----
1
Customer:5468 take the data:64
-----
Product_Space:0 0 0
-----
Customer:5468 exit
Product_Space:0 0 0
-----
Producer:12172 put the data:45
2
Product_Space:0 0 45
-----
Producer:12172 exit
Product_Space:0 0 45
-----
2
Customer:10720 take the data:45
-----
Product_Space:0 0 0
-----
Customer:10720 exit
MainProgram exit
C:\Users\Administrator\Desktop\OS Program>
```

Linux

```
root@tos211-vpc: /home/tos/Desktop
root@tos211-vpc:/home/tos/Desktop# ./MainProgram.exe
Producer:3539 WaitTime:3s
-----
Producer:3540 WaitTime:3s
-----
Customer:3541 WaitTime:3s
-----
Customer:3542 WaitTime:3s
-----
Customer:3543 WaitTime:3s
-----
Product_Space:0 0 0
-----
Producer:3539 put the data:86
-----
0
Product_Space:86 0 0
-----
Producer:3539 WaitTime:2s
-----
Product_Space:86 0 0
-----
Producer:3540 put the data:86
-----
```

```
root@tos211-vpc: /home/tos/Desktop
1
Product_Space:86 86 0
-----
Producer:3540 WaitTime:2s
-----
Product_Space:86 86 0
-----
0
Customer:3542 take the data:86
-----
Product_Space:0 86 0
-----
Customer:3542 WaitTime:1s
-----
Product_Space:0 86 0
-----
1
Customer:3543 take the data:86
-----
Product_Space:0 0 0
-----
Customer:3543 WaitTime:1s
-----
Product_Space:0 0 0
```