

云南大学数学与统计学院 上机实践报告

课程名称：操作系统实验	年级：2015 级	上机实践成绩：
指导教师：李源	姓名：刘鹏	
上机实践名称：生产者与消费者问题	学号：20151910042	上机实践日期：2017-10-25
上机实践编号：No.04	创建时间：	最近一次修改时间：20:24

一、实验目的

1. 掌握进程的概念，明确进程的含义；
2. 认识并了解并发执行的实质。

二、实验内容

编制一段程序，实现软中断通信：

使用系统调用 `fork()` 创建两个子进程，再用系统调用 `signal()` 让父进程捕捉键盘上来的中断信号（即按 `Del` 键），当父进程接受到这两个软中断的其中一个后，父进程用系统调用 `kill()` 向两个子进程分别发送整数值为 16 和 17 软中断信号，子进程获得对应软中断信号后，分别输出下列信息后终止：

```
Child process 1 is killed by parent!!  
Child process 2 is killed by parent!!
```

父进程调用 `wait()` 函数等待两个子进程终止后，输出以下信息后终止：

```
Parent process is killed!!
```

三、实验要求

生产者、消费者实验

三个生产者向两个消费者提供消息，它们共享一个有界缓冲池，缓冲池有四个缓冲区，生产者向其中投放消息，消费者从中取得消息。假定这些生产者和消费者互相等效，只要缓冲池未满，生产者可将消息送入缓冲池；只要缓冲池未空，消费者可从缓冲池取走一个消息。每个缓冲设有一个锁标记，生产者在生产产品以前要先判断缓冲区是否已上锁，消费者在消费产品之前也要进行判断。

要求，在 **Microsoft Visual C++** 上试运行，查看程序执行过程及结果，画出执行流程图。

四、实验内容与步骤

程序代码：

```
1  #include<iostream>
2  #include <windows.h>
3  #include "stdio.h"
4
5  using namespace std;
6
7  bool lock[4] = { false,false,false,false };
8  int buffer[4] = { 0,0,0,0 };
9  void display() {
10     cout << "-----" << endl;
11     cout << "缓冲区 0 1 2 3" << endl;
12     for (int i = 0; i<4; i++)
13     {
14         cout << "" << buffer[i] << " ";
15     }
16     cout << endl;
17 }
18
19 //生产者生产进程函数
20
21 DWORD WINAPI Producer(LPVOID lpParameter)
22 {
23     while (true) {
24         for (int j = 0; j<4; j++)
25         {
26             if (buffer[j] == 0)
27             {
28                 if (lock[j] == false)
29                 {
30                     lock[j] = true;
31                     if (buffer[j]<1)
32                     {
33                         ++buffer[j];
34                         cout << "生产一个资源，放入缓冲区" << j << "中" << endl;
35                         lock[j] = false;
36                         break;
37                     }
38                 }
39                 if (j == 3) {
40                     cout << "找不到空缓冲区，等待中。。" << endl;
41                     Sleep(2000);
42                 }
43             }
44         }
45     }
```

```
46     return 0;
47 }
48
49 //消费者消费进程函数
50 DWORD WINAPI Customer(LPVOID lpParameter)
51 {
52     while (true) {
53         for (int n = 0; n<4; n++)
54         {
55             if (buffer[n] == 1)
56             {
57                 if (lock[n] == false)
58                 {
59                     lock[n] = true;
60                     if (buffer[n] >= 1)
61                     {
62                         --buffer[n];
63                         cout << "消费一个资源，从缓冲区" << n << "中取出" << endl;
64                         lock[n] = false;
65                         break;
66                     }
67                 }
68             }
69             if (n == 3)
70             {
71                 cout << "找不到满缓冲区，等待中。。" << endl;
72                 Sleep(2000);
73             }
74         }
75     }
76 }
77
78
79 //主函数，创建 5 个进程，其中 3 个生产者进程，2 个消费者进程
80 int main(int argc, char* argv[])
81 {
82     display();
83     HANDLE handle[3];
84     DWORD dw1, dw2, dw3, dw4, dw5; //创建 5 个进程
85     handle[0] = CreateThread(NULL, 0, Producer, NULL, 0, &dw1);
86     handle[1] = CreateThread(NULL, 0, Producer, NULL, 0, &dw2);
87     handle[2] = CreateThread(NULL, 0, Producer, NULL, 0, &dw3);
88     handle[3] = CreateThread(NULL, 0, Customer, NULL, 0, &dw4);
89     handle[4] = CreateThread(NULL, 0, Customer, NULL, 0, &dw5);
90     display();
91     Sleep(3000);
92     display();
93     return 0;
94 }
```

流程设计:

结果分析:

五、实验总结

六、参考文献