山东大学 软件 学院

操作系统课程设计 课程实验报告

|  |  |  |  |
| --- | --- | --- | --- |
| 学号：201600301079 | 姓名： 崔玉峰 | | 班级： 2016级软件4班 |
| 实验题目：Laboratory 3: Synchronization Using Semaphores | | | |
| 实验学时：4 | | 实验日期： 2018/10/23 | |
| 实验目的：  In this laboratory session, you are required to write a test program for the producer/consumer problem using semaphores for synchronization. After completing the session, you will  • have a understanding in Nachos of  – how semaphores are implemented, and  – how the producer/consumer problem is implemented using semaphores  • know how to create concurrent threads in Nachos, and  • know how to test and debug programs in Nachos.  The work of this laboratory session will prepare you for the programming task in Assignment 2. | | | |
| 硬件环境：  PC | | | |
| 软件环境：  虚拟机：VMWARE  操作系统：ubuntu 16.04.5 32位 | | | |
| 实验步骤与内容：  实验内容主要是完善代码，实现生产者消费者同步问题   1. 首先查看 ring.h 和 ring.cc   ring定义了一个环形的缓冲区，可以用作生产者消费者问题中的共享内存区域。  Ring.Put(slot \*message); //向缓冲区放入一个消息  Ring.Gut(slot \*message); //从缓冲区读出一个消息，放入message  O0U`6BAB~DHAB{0%WX]L$D8  slot定义了缓冲区中的数据类型，存储进程号thread\_id和值value  **W3F4{B(E(P6D`_EB_(3Y_J8**   1. **查看main.cc**   本实验的入口程序，主要功能是初始化主线程，调用prodcons++.cc中的ProdCons()方法来模拟生产者消费者问题。   1. 查看prodcons++.cc   ProdCons()方法:  **初始化信号量，共享缓冲区并且创建两个生产者进程，两个消费者进程。并查看运行情况，**  **LUEO$Z(PTOQY@YO5R2DA{BO**  Producer(\_int which)方法：  生产者进程要执行的方法,创建消息实例，并将其加到共享缓冲区内，需要补全信号量，同步伪代码如下：  P(empty)  P(mutex)  ring->put(message)  P(mutex)  P(full)  Consumer(\_int which)方法：  消费者进程执行的方法， 创建两个文件输出执行情况，不断从共享缓冲区中取出消息，并不断写到文件中去。同步伪代码如下：  P(full)  P(mutex)  ring->get(message)  P(mutex)  P(empty)   1. 查看synch.cc中Semaphore   Semaphore信号量的实现，可设置信号量的名字和初始值，定于了两个方法P()和V()均为原语  `7WH%R`~FI$T{FQJI4)D]0K   1. 程序具体补全：   补全prodcons++.cc中的相关方法来模拟生产者和消费者问题。  ProdCons()方法  创建三个信号量nempty 初始值是缓存区的长度  nfull 初始值是0  mutex 初始值是1  创建缓冲区ring = new Ring();  分别创建两生产者进程，两个消费者进程 通过Fork()方法初始化。  mmexport1540536110066    Producer()方法  具体实现情况如下：  mmexport1540536102916  Customer（）方法：  mmexport1540536106361     1. 执行结果：   编译发现报错添加#include <unistd.h> #include <fcntl.h> 后正常运行执行情况如下：    mmexport1540537221642mmexport1540537216759  mmexport1540537218928 | | | |
| 结论分析与体会：  通过本次实验了解了nachos线程创建执行的逻辑，并且学习到了如何定义和使用信号量来实现线程的同步，同样也学习到了信号量的实现，PV操作的实现。  消费者生产者问题是最经典的线程同步问题，用信号量进行同步的代码相对简单，已完成本次实验要求的全部情况，并且测试通过。 | | | |