山东大学 软件 学院

操作系统 课程实验报告

|  |  |  |  |
| --- | --- | --- | --- |
| 学号：201400301291 | 姓名： 雷超 | | 班级： 四班 |
| 实验题目：进程互斥实验 | | | |
| 实验学时：2 | | 实验日期： 2016-05-03 | |
| 实验目的：  进一步研究和实践操作系统中关于并发进程同步与互斥操作的一些经典问题的解法，加深对于非对称性互斥问题有关概念的理解。观察和体验非对称性互斥问题的并发控制方法。进一步了解Linux操作系统中IPC进程同步工具的用法，训练解决对该类问题的实际编程，调试和分析能力。 | | | |
| 硬件环境：  00:00.0 Host bridge: Intel Corporation Xeon E3-1200 v3/4th Gen Core Processor DRAM Controller (rev 06)  00:01.0 PCI bridge: Intel Corporation Xeon E3-1200 v3/4th Gen Core Processor PCI Express x16 Controller (rev 06)  00:02.0 VGA compatible controller: Intel Corporation 4th Gen Core Processor Integrated Graphics Controller (rev 06)  00:03.0 Audio device: Intel Corporation Xeon E3-1200 v3/4th Gen Core Processor HD Audio Controller (rev 06)  00:14.0 USB controller: Intel Corporation 8 Series/C220 Series Chipset Family USB xHCI (rev 05)  00:16.0 Communication controller: Intel Corporation 8 Series/C220 Series Chipset Family MEI Controller #1 (rev 04)  00:1a.0 USB controller: Intel Corporation 8 Series/C220 Series Chipset Family USB EHCI #2 (rev 05)  00:1b.0 Audio device: Intel Corporation 8 Series/C220 Series Chipset High Definition Audio Controller (rev 05)  00:1c.0 PCI bridge: Intel Corporation 8 Series/C220 Series Chipset Family PCI Express Root Port #1 (rev d5)  00:1c.1 PCI bridge: Intel Corporation 8 Series/C220 Series Chipset Family PCI Express Root Port #2 (rev d5)  00:1c.2 PCI bridge: Intel Corporation 8 Series/C220 Series Chipset Family PCI Express Root Port #3 (rev d5)  00:1c.4 PCI bridge: Intel Corporation 8 Series/C220 Series Chipset Family PCI Express Root Port #5 (rev d5)  00:1d.0 USB controller: Intel Corporation 8 Series/C220 Series Chipset Family USB EHCI #1 (rev 05)  00:1f.0 ISA bridge: Intel Corporation HM86 Express LPC Controller (rev 05)  00:1f.2 SATA controller: Intel Corporation 8 Series/C220 Series Chipset Family 6-port SATA Controller 1 [AHCI mode] (rev 05)  00:1f.3 SMBus: Intel Corporation 8 Series/C220 Series Chipset Family SMBus Controller (rev 05)  01:00.0 3D controller: NVIDIA Corporation GM107M [GeForce GTX 860M] (rev a2)  08:00.0 Network controller: Intel Corporation Wireless 3160 (rev 93)  09:00.0 Ethernet controller: Realtek Semiconductor Co., Ltd. RTL8111/8168/8411 PCI Express Gigabit Ethernet Controller (rev 10)  0a:00.0 Unassigned class [ff00]: Realtek Semiconductor Co., Ltd. RTS5249 PCI Express Card Reader (rev 01) | | | |
| 软件环境：  Linux EVA 4.6.2-gentoo-EVOLUTION x86\_64 Intel(R) Core(TM) i7-4710HQ CPU @ 2.50GHz GenuineIntel GNU/Linux  dev-libs/gmp-6.0.0a  dev-libs/mpfr-3.1.3\_p4  dev-libs/mpc-1.0.2-r1  sys-devel/gcc-4.9.3 , 5.3.0 , 5.4.0  sys-libs/glibc-2.22-r4  sys-devel/clang-3.5.0-r100  app-editors/emacs-24.5-r1  sys-kernel/gentoo-sources-4.6.2  sys-kernel/linux-headers-4.6  sys-apps/systemd-226-r2  ABI\_X86=”32 64” | | | |
| 实验步骤与内容：  理发师问题：理发店的理发厅有3个理发椅子和3个理发师，有一个4个座位的沙发。还有一个等待厅，可以容纳13个顾客。如果顾客满，则不进入。  在理发店内，理发师一有空就为沙发上等待时间最长的人理发，空出的座位为等待时间最长的顾客。理完发后，可以向任何一位理发师缴费，但是只有一本现金登记手册，在任意一刻只能记录一个顾客的付款。理发师没有顾客的时候就在椅子上睡眠直到被顾客唤醒。理发师的时间就用于理发，收款，睡眠上。 | | | |
| 结论分析与体会：  理发师问题相对前面几个实验稍微复杂一点，但是还是可以解决的。对于顾客进程而言，查看店内有没有人使用互斥锁，使用P（）操作所具有的消息队列来达成对于先到的顾客进行排队的机制，满足等待时间最长的顾客先被服务；当顾客进入沙发时，若沙发没有人，则进行尝试唤醒理发师；若有人，判定是否满，若满则二次判定是否等待厅的人数是否满。  当理发师理发完毕的时候，进行调度，确定是谁应该进入沙发，进行理发，并且查看收账的消息队列中是否有客人进行缴费操作。若这些都没有就进入沉睡，直到被唤醒。 | | | |

附件：