**电 子 科 技 大 学**

**实 验 报 告**

**学生姓名： 学号： 指导教师：**

**实验地点： 实验时间：**

1. **实验室名称：**
2. **实验项目名称：shell设计（进程调度）**
3. **实验学时：4**
4. **实验原理：**

## 4.1 总体设计

系统总体架构如图1所示，最右边部分为进程与资源管理器，属于操作系统内核的功能。要求能够设计与实现一个简单的进程与资源管理器，具有如下功能：完成进程创建、撤销和进程调度；完成多单元 (multi\_unit)资源的管理；完成资源的申请和释放；完成错误检测和定时器中断功能。



图1 系统总体结构

图1中间绿色部分为驱动程序test shell, 设计与实现test shell，该test shell将调度所设计的进程与资源管理器来完成测试。Test shell的应具有的功能：

* + 从终端或者测试文件读取命令；
  + 将用户需求转换成调度内核函数（即调度进程和资源管理器）；
  + 在终端或输出文件中显示结果：如当前运行的进程、错误信息等。

图1最左端部分为：通过终端（如键盘输入）或者测试文件来给出相应的用户命令，以及模拟硬件引起的中断。

## 4.2 shell设计

shell应该具有的功能：

（1）读入指令

（2）根据指令，执行相应进程调度和资源管理操作

（3）要求的功能：创建/删除进程，申请/撤销资源，

1. **实验目的：**

设计和实现进程与资源管理，并完成Test shell的编写，以建立系统的进程管理、调度、资源管理和分配的知识体系，从而加深对操作系统进程调度和资源管理功能的宏观理解和微观实现技术的掌握。

1. **实验内容：**

实现一个简单的shell，实现进程的创建，删除，资源的请求和进程调度的功能。

1. **实验器材（设备、元器件）：**

## 7.1 硬件平台

本实验所需的硬件主要是PC计算机。

**7.2 软件平台**

1. 操作系统：Win 7、Windows10；
2. 编程语言：Python。
3. **实验步骤：**

1.类和全局变量

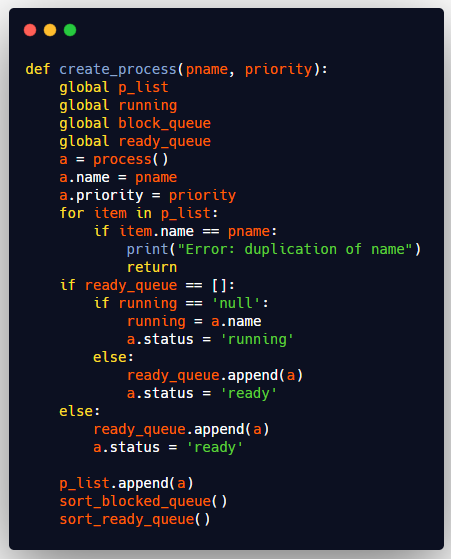
新建进程类和各种全局变量，方便对进程状态进行管理，相关代码如下：



process类的name属性表示进程名，r1，r2，r3，r4属性分别表示其目前占有的四类资源的数量，status的值可以为running、blocked和ready三种，分别代表了运行、阻塞和就绪三个状态。

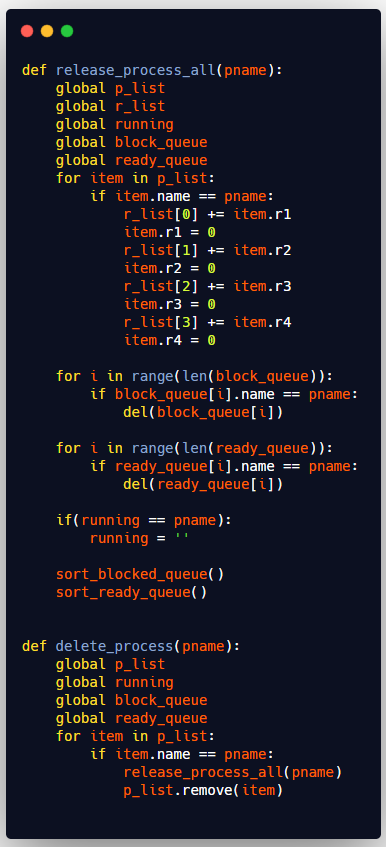
2.创建进程

新建进程类和各种全局变量，方便对进程状态进行管理，相关代码如下：



3.删除和释放进程

删除是将进程资源释放后，还要删除其在所有表中的记录，让此进程彻底消失，而释放资源之后释放该进程的资源，之后该进程会重新进入就绪队列。相关代码如下：



4.请求资源

请求资源是为当前正在运行的进程请求资源，如果资源不能得到满足，则进程进入阻塞队列，待有资源可以分配时，再进入就绪队列。否则可以将资源分给该进程，相关代码如下：

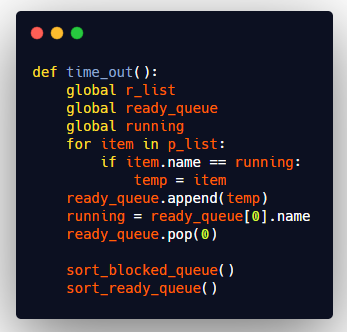
5.请求资源

请求资源是为当前正在运行的进程请求资源，如果资源不能得到满足，则进程进入阻塞队列，待有资源可以分配时，再进入就绪队列。否则可以将资源分给该进程，相关代码如下：



6.time\_out

time\_out是模拟当前进程的时间片已到，将该进程移入就绪队列，并将就绪队列头部进程移入running，相关代码如下：



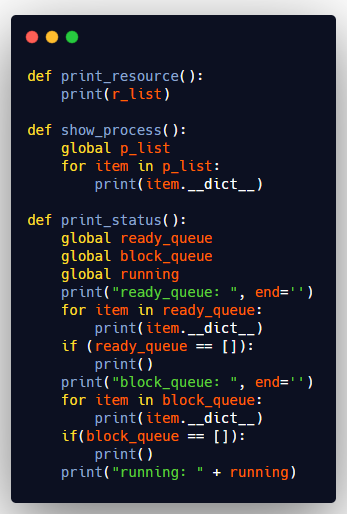
7.实现优先级

本实验中优先级有1、2、3，其中1的优先级最高，3的优先级最低。要实现按照优先级的先来先服务调度，只需要在原来的先来先服务调度算法中，将就绪队列和阻塞队列改造为优先队列，其方法是每次队列有更新时，对其进行排序，或者插入时，按照优先级插入。本实验代码采用第一种排序的方法，相关代码如下：



8.打印状态

为了方便验证结果，新增了三个打印状态的指令，分别可以打正运行进程running、阻塞队列blocked\_queue、就绪队列ready\_queue，资源状态表r\_list和进程状态表p\_list，相关代码如下:



**八、实验数据及结果分析：**

1.创建和删除进程：

在命令行输入相关指令，得到结果如图1：

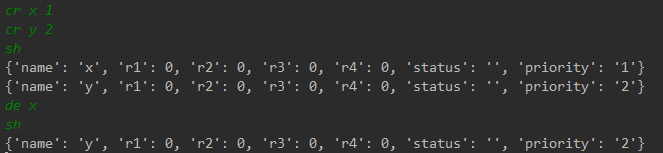


图 1. 创建和删除进程功能示意图

2.分配资源：

在命令行输入相关指令，得到结果如图2：

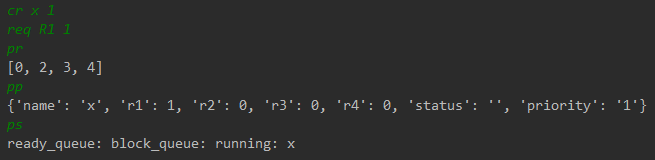


图 2. 分配资源功能示意图

无法分配资源的情况，测试如图3：

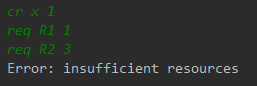


图 3. 无法分配资源功能示意图

3.删除和释放：

在命令行输入相关指令，得到结果如图4和图5所示（这里新加了再前面输出shell>字符的功能，然shell看起来更真实）：

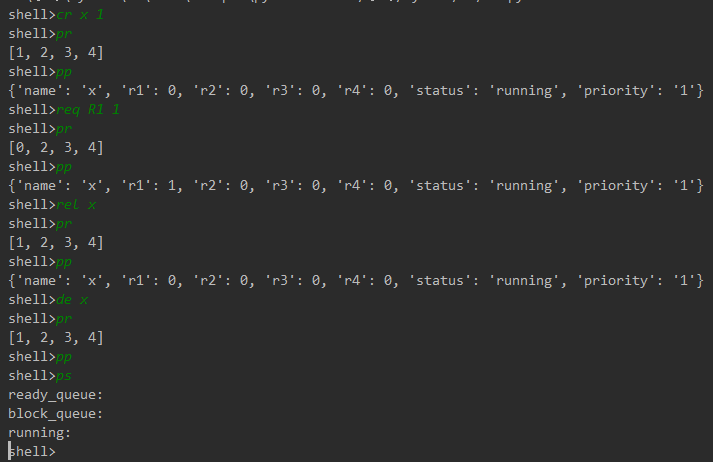


图 4. 删除进程功能示意图

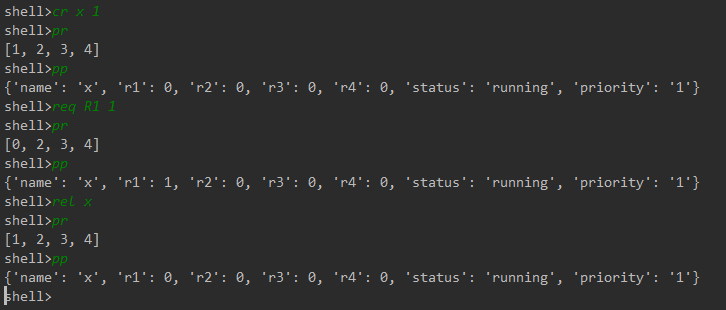


图 5. 释放进程功能示意图

4.time\_out功能测试：

在命令行输入相关指令，得到结果如图5：

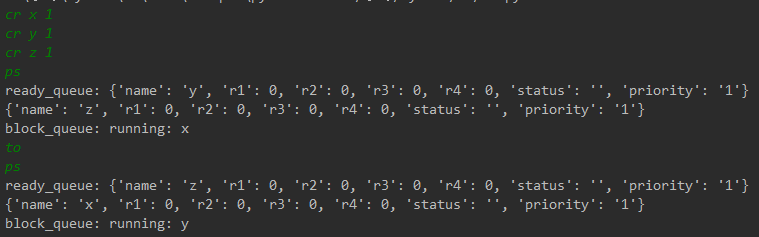


图 5. 创建和删除进程功能示意图

5.完整功能测试

在命令行输入相关指令，得到结果如图6和图7：

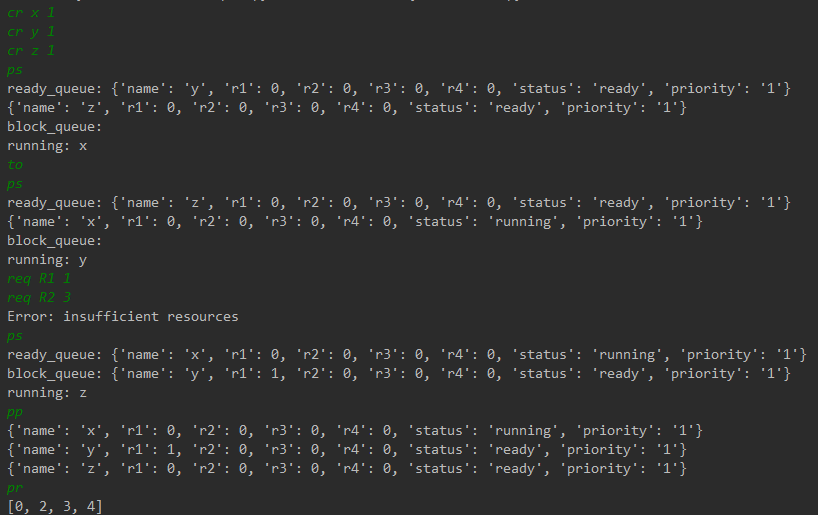


图 6. 完整功能测试图1

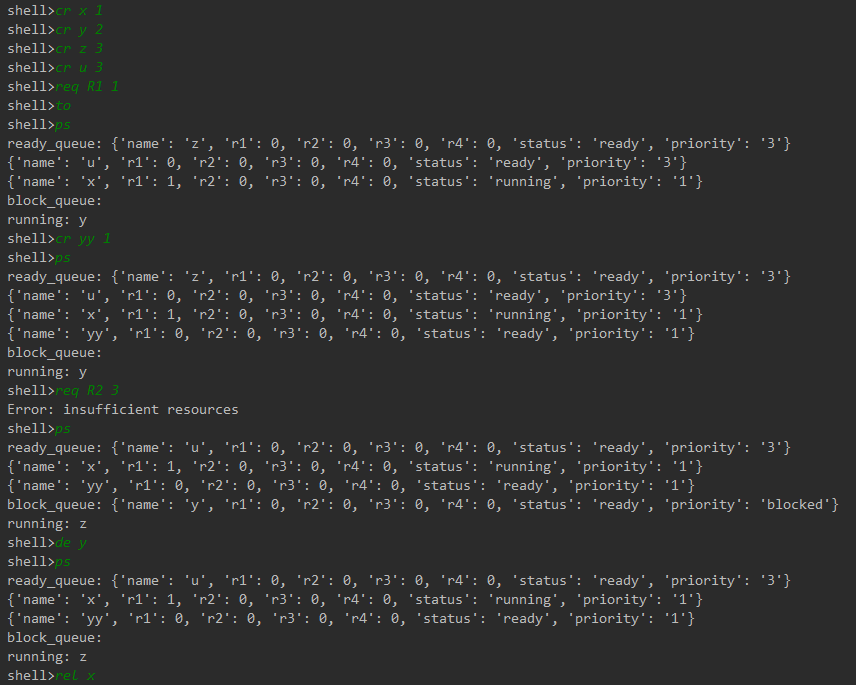


图 7. 完整功能测试图2

综上所述，所实现的shell能很好地完成进程创建、释放、删除、time\_out和分配资源及进程调度的功能。调度算法是按照优先级的先来先服务算法。实验结果正确，程序功能正常，实验成功。

1. **实验结论：**

计算机中的进程调度是一个复杂的过程，一般不会是简单的先来先服务算法。

1. **总结及心得体会：**

操作系统的设计是计算机科学中很有挑战性的一步。进程管理作为操作系统中重要的一步，其重要程度不言而喻。

1. **对本实验过程及方法、手段的改进建议：**

可以结合“计算机网络”这门课程中关于“应用层”的进程消息传递的内容，进一步深化相关知识。

**报告评分：**

**指导教师签字：**