**西安电子科技大学**

**操作系统课程设计**

**(2020年度)**

**实**

**验**

**报**

**告**

**实验名称：** Priority-scheduling Task1

**班 级：** 1803011

**姓 名：** 张帅豪

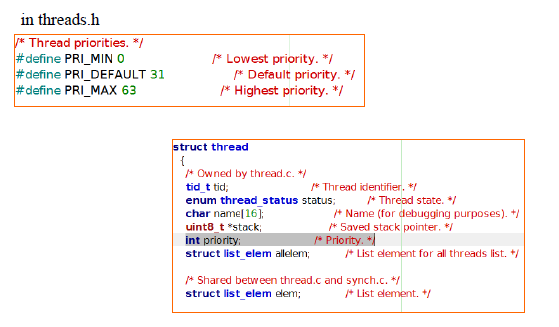
**学 号：** 18030100101

**一、实验内容**

1. 原始Pintos系统中对于线程的调度，没有考虑优先级问题，采用的是最为简单的FCFS（先来先服务策略）。
2. 本实验要求为Pintos建立优先级调度机制，确保任何时刻CPU上运行的都是最高优先级线程。

**二、分析及设计**

1. 对threads.h和threads.c进行修改，为结构体thread添加新变量priority，设置优先级变量。



1. 在线程生成的时候，即在thread.c中修改thread\_create（）函数，在其中的变量声明中，也要加上优先级变量，同时声明为函数设置优先级的函数thread\_set\_priority（）。

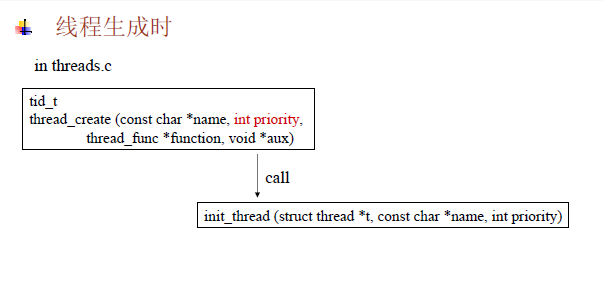
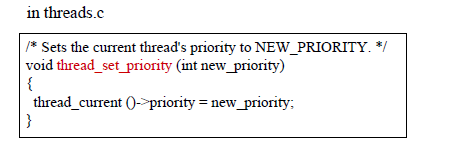
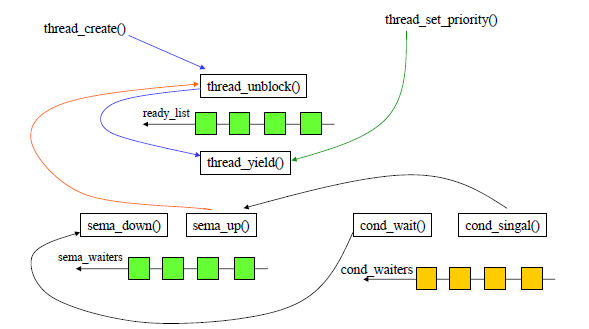


图 1 线程生成时



1. 在工作过程中，需要对到来的进程进行优先级判断，并修改线程执行的队列，各个函数调用关系如下。



**三、详细实现**

1、在thread.c中修改函数

**（1）修改函数thread.create（）**



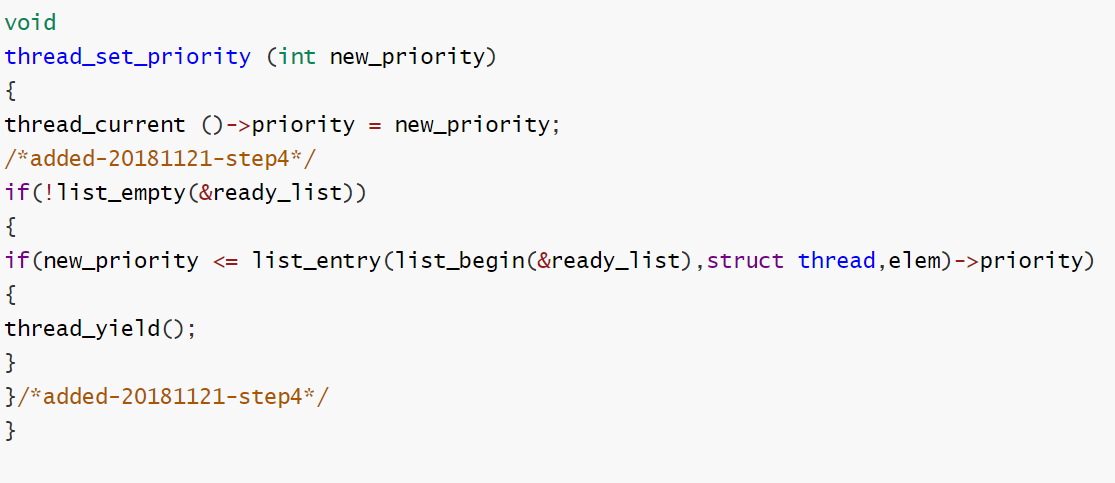
**（2）修改函数thread\_yield（）**



**（3）修改函数thread\_unblock（）**



**（4）修改函数thread\_set\_priority（）**



**（5）定义函数thread\_less（）和thread\_priority\_check（）**



2、在thread.h中修改

**（1）声明函数函数thread\_less（）和thread\_priority\_check（）**

3、在synch.c中修改

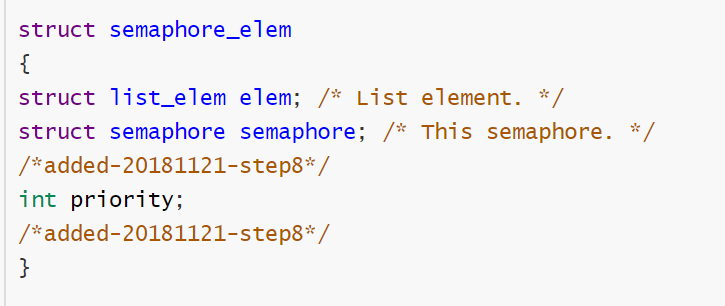
**（1）修改函数sema\_down（）**



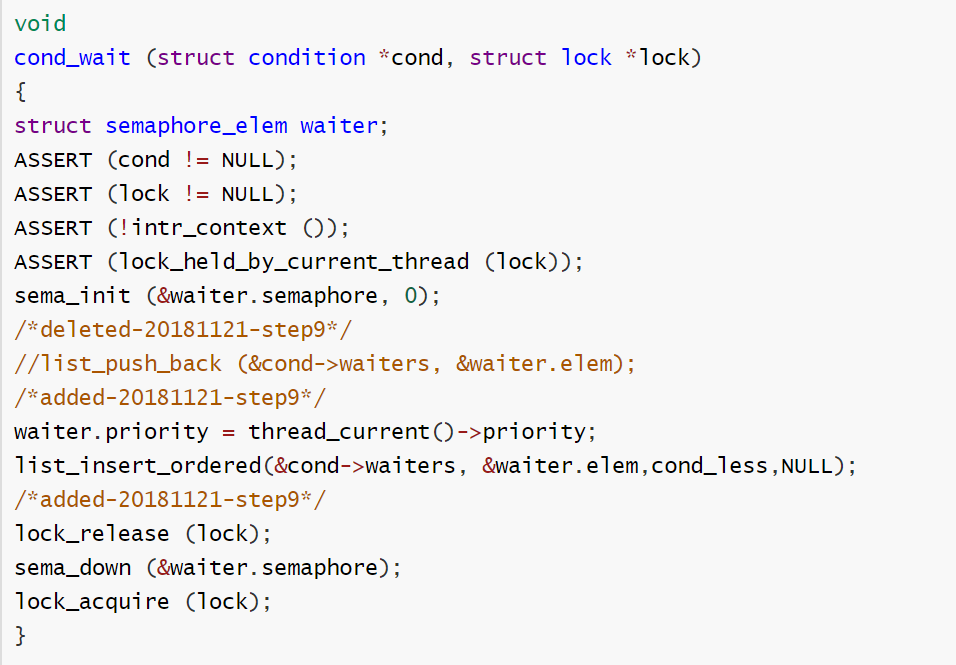
**（2）修改函数sema\_up（）**



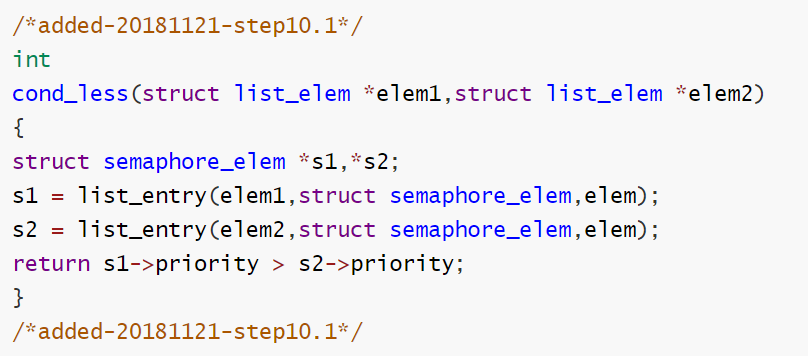
**（3）修改结构体semaphore\_elem**



**（4）修改函数cond\_wait（）**

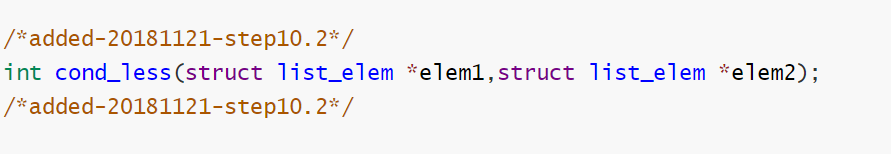


**（5）定义函数cond\_less（）**



4、在synch.h中修改

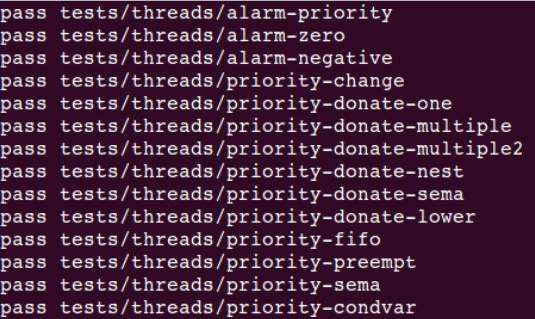
**（1）声明函数cond\_less（）**



**四、实验结果**

在/home/xd/os/pintos/src/thread下执行make check指令，比对正确结果中的pass项，发现实验成功。

具体运行结果如下：



**五、心得体会**

通过这次实验，我深入了解了系统线程对系统的影响，可以通过亲手修改设计，然系统对线程进行不同的cpu占用，通过重新设定系统的优先级，是系统的处理效率更高，对Pintos系统修改实现优先级调度机制，让系统总是使最高优先级的线程进行。同时通过这次亲手实践，让我把上学期学到的操作系统的理论知识得到了实践，对操作系统的实现和运行有了更为深刻的理解。