**操作系统实习**

Lab2 线程调度 实习说明

本实习希望通过修改Nachos系统平台的底层源代码，达到“扩展调度算法”的目标。

【实习内容】

**Exercise 1 调研**

调研Linux或Windows中采用的进程/线程调度算法。具体内容见课堂要求。

**Exercise 2 源代码阅读**

仔细阅读下列源代码，理解Nachos现有的线程调度算法。

* code/threads/scheduler.h和code/threads/scheduler.cc
* code/threads/switch.s
* code/machine/timer.h和code/machine/timer.cc

**Exercise 3 线程调度算法扩展**

扩展线程调度算法，实现基于优先级的抢占式调度算法。

**\*Challenge 1 线程调度算法扩展**（至少实现一种算法）

可实现“时间片轮转算法”、“多级队列反馈调度算法”，或将Linux或Windows采用的调度算法应用到Nachos上。

【实习建议】

**1. 数据结构的修改和维护**

线程核心管理机制的升级首先依赖于对线程数据结构的修改。例如实现“时间片轮转”线程调度机制时，必须首先在线程数据结构中增加“已使用时间片计数”这样一个变量。同时在“线程创建”、“时钟中断”、“线程切换”等代码中，增加对这个数据成员的维护性代码。

**2. 线程管理机制所依赖的细节技术处理**

线程管理机制的运行过程非常复杂，实践过程中应把握好关键的处理步骤：

* 时钟中断处理：在Nachos中，时钟中断处理包含了引起线程调度的代码，修改线程调度机制必须调整这部分代码。
* 线程上下文切换：在发生线程切换时，必须妥善保存线程的上下文。由于修改了线程的底层数据结构，因此上下文切换的代码也必须修改。
* 线程调度：这是本实习的核心代码，应认真阅读后再进行修改。