

“操作系统”课程实验教学探讨

陈渝

清华大学计算机系

yuchen@tsinghua.edu.cn chyyuu@gmail.com

2010年7月24日

报告内容

- 前言
- 已有现状
- 实验课程设计
- 效果和存在的问题
- 小结

前言

- 对操作系统实验教学的理解
 - 计算机科学与计算机工程相结合
 - 原理和实验教学内容并行进行
 - 原理-->实验-->原理
 - 强调动手编程实践

前言

- 对操作系统实验教学的理解
 - 实验需求
 - 理解硬件
 - 循序渐进
 - 阅读代码
 - 把握全局
 - 功能完善
 - 改进创新

已有现状

- 当前国外的现状
 - MIT: xv6-2.0和JOS
 - 7千行以下, C语言, 支持X86 SMP架构
 - 清华: ucore
 - 基于jos/xv6/linux/rtems, 5千行左右, C语言, 支持X86, 组件化
 - Harvard: OS161-1.4.1
 - 1万1千行代码, C语言, 支持MIPS架构
 - Columbia: Linux
 - 部分Linux核心代码, C语言
 - Berkeley: Nachos
 - 1万行左右, java语言, 模拟MIPS架构

已有现状

- 当前国内的现状
 - Ucore
 - 清华
 - xv6和JOS
 - 北大
 - Linux
 - 国防科大、浙大、西邮、清华
 - MINIX
 - 上海交大，南开
 - Nachos
 - 南开，山大
 - Solaris Windows WRK Wince RTEMS ..

已有现状

- 清华大学的OS课程
 - 操作系统原理：本科大三下，160人左右
 - 原理与实验
 - 操作系统实践：本科大四上，60人左右
 - 侧重创新型实践，Linux....
 - 高级操作系统：研究生课程，40人左右
 - 操作系统前沿+研究型实验，SOSP, OSDI, EuroSys...

实验课程设计

- 目标
 - 对原理知识的补充和完善
 - 讲课内容和实验内容同步
 - 让学生对操作系统设计有一个全局的理解
 - 操作系统要小巧且覆盖面全
 - 适合不同层次学生的需求
 - 存在高中低三类学生

实验课程设计

- 设计思路

- 方便且利用理解细节

- 大量采用开源软件
 - 实验环境: **Windows/Linux**
 - **IDE工具: Eclipse**
 - 源码阅读工具: **Kscope**
 - 源码文档自动生成工具: **Doxygen**
 - 编译环境: **gcc, make, Binutils**
 - 真实/虚拟运行环境: **X86机器或QEMU**
 - 调试工具: **改进的QEMU + GDB**

实验课程设计

- 设计思路
 - 采用小巧全面的操作系统**ucore**并进行改进，需要覆盖操作系统的关键点，为此增加：
 - I/O管理/中断管理
 - 虚存管理/页表/缺页处理/页替换算法
 - 进程管理/调度器算法
 - 信号量实现和同步互斥应用
 - 基于链表/**FAT**的文件系统
 - 完整代码量控制在**5000**行左右
 - 提供实验讲义和源码分析文档

实验课程设计

- 设计思路

- 利用互联网进行广泛的交流：

- 建立邮件列表：答疑、交流、总结

- 代码和实验讲义公开

- 开发邮件列表：

- https://groups.google.com/group/ucore_dev

- 论坛：<https://groups.google.com/group/oscourse>

- 源码：`svn checkout http://ucore.googlecode.com/svn/trunk/ ucore-read-only`

实验课程设计

- 设计思路
 - 差异化教学
 - 很好的学生：鼓励创新
 - 中差学生：鼓励根据各自层次选择合适的实验方式

实验课程设计

• 实验内容

- 1 OS启动、中断与设备管理: 200~1617行代码
- 2 内存管理: 1800~3000行代码
- 4 进程管理: 3500~4100行代码
- 5 处理器调度: 4300~5000行代码
- 6 同步互斥与死锁: 5100~5400行代码
- 7 文件系统: 5500~5900行代码
- 8 Web/TCP/IP/net driver: 6000~8000行代码

效果和存在的问题

- 好的方面

- 理论和实验能够较好地结合起来，不再感到**OS**课是一个只要死记硬背的课程了
- 理解了一个**OS**的全局设计于实现，而不是一个一个分离的知识点
- 掌握了许多**OS**原理上没有涉及或涉及不够的东西，比如中断/系统调用的实现，**X86**的段页机制，进程上下文如何切换的，内核态和用户态的具体区别是什么
- 这是大学期碰到的最复杂的软件，学习了分析和设计大型系统软件的方法

效果和存在的问题

• 存在的问题

- 对Linux和相关软件和X86保护模式中中断等不熟悉
 - 上课讲解相关内容，提高学生分析代码和编程能力
- 大三下课程繁重，在OS实验上花费时间多，对其他课程有一定的影响
 - 只要求完成部分实验，其它实验提供参考答案来理解
- 对抄袭没有有效的监管手段
 - 实验成绩30%，考试成绩(包含原理和实验内容)70%
- 好学生吃不饱，差学生感觉是“下地狱”
 - 实验要求和参考答案一起给
 - 进一步完善实验文档、辅助参考文档和辅助工具
 - 让3%的好学生参加OS科研，让差学生理解参考答案

小节

- 个人体会

- 一个可实际运行的，代码基小且实验点覆盖操作系统各个关键知识点的微型实验OS平台对学生学好操作系统课程有极大的促进作用
- 吸取国内外先进经验并进行再加工
- 广泛使用丰富的开源软件
- 教师讲课内容与实验内容紧密结合
- 学生需要进行差异化实验教学
- 建立开放的实验交流环境，实现知识复用

谢谢！

欢迎访问、下载使用相关文档和实验软件并加入**OS Course**讨论组进行交流！

开发邮件列表：https://groups.google.com/group/ucore_dev

论坛：<https://groups.google.com/group/oscourse>

源码：`svn checkout http://ucore.googlecode.com/svn/trunk/ ucore-read-only`