**山东大学 计算机科学与技术 学院**

**《 操作系统课程设计 》教学大纲**

编写人：韩芳溪 审定人：杨兴强

编制时间：2017.6.7 审定时间：2017.9.16

**一、课程基本信息：**

|  |  |  |  |
| --- | --- | --- | --- |
| 课程名称 | 操作系统课程设计（双语） | | |
| 英文名称 | Operating System curriculum design | | |
| 课程编码 | sd01320190 | | |
| 开课单位 | 计算机科学与技术学院 | | |
| 课程类别 | □通识教育必修课程 □通识教育核心课程  □通识教育选修课程 □学科基础平台课程  √专业基础课程 □专业必修课程 □专业选修课程 | | |
| 课程性质 | √必修 □选修 | | |
| 学分 | 2 | 学时 | 16+32 |
| 适用专业 | 计算机科学与技术 | | |
| 先修课程 | 操作系统，C++语言 | | |
| 课程网站 | http://course.sdu.edu.cn/G2S/Template/View.aspx?action=view&courseType=0&courseId=1487 | | |

**二、课程描述**

操作系统课程设计是计算机科学与技术专业的实践必修课，以一个教学用的操作系统Nachos的设计和实现为例，阐述操作系统软件中具有普遍意义的方法和结构。课程要求学生在分析Nachos系统源代码的基础上，按指定要求，设计编程完善系统的功能。

课程由课堂教学和实验两部分构成：课堂教学介绍Nachos操作系统的整体架构，引领学生分析nachos系统代码；实验部分由学生自己动手设计编程实现课程目标。

课程的特色是：通过编程实验，在培养学生动手能力的同时，加深学生对操作系统运行机制的理解。

The operating system curriculum is a practice compulsory course of Computer Science and Technology. It expounds methods and structures with an implementation of an operating system (Nachos). Students should program to finish the Nachos, based on the analysis of Nachos operating- system.

This curriculum is comprised of two parts: the classroom teaching introduces the architecture of Nachos and lead the students to analyze the source code; experiments will train the programming abilities of students.

It is the feature of this curriculum that experiments will strengthen both the students’ programming abilities and understanding for the operating system mechanisms.

**三、课程教学目标和教学要求**

【教学目标】

通过本课程，我们希望学生应获得如下能力：

1. 分析大型程序代码的能力。

2. 程序设计能力。

3. 系统能力。

【教学要求】

1. 将计算机硬件系统、操作系统和应用软件系统的知识贯穿起来，获得系统能力和综合应用能力。（综合应用）
2. 学生能够Nachos的现有框架，设计针对课程目标的综合解决方案。（设计/开发解决方案）
3. 鼓励学生积极参与课堂讨论，锻炼学生表达与理解能力。（沟通）
4. 学生分组，共同完成课程设计目标。（个人和团队）
5. 学生一方面研究数据结构和算法在操作系统中的应用，另一方面研究操作系统代码和运行机制，提升创新研究能力。（研究）
6. 要求学生熟练使用C++语言、IDE以及调试环境。（使用现代工具）

**四、课程教学内容及学时分配**

**第一章 Nachos虚拟机（4学时授课+2学时上机）**

本章目标：

1. 描述Nachos的基本组成
2. 说明Nachos虚拟机的模拟方法
3. 熟悉Nachos的安装过程及调试方法

第一节 Nachos的硬件设备

【教学目标】通过了解Nachos机器的各种硬件设备，形成Nachos计算机的整体轮廓。

【教学要求】有选择地介绍Nachos硬件。

【教学内容】计时器，中断，磁盘，串口，指令系统。

【教学和学习建议】讲授、讨论。

【难点重点】整体把握Nachos的硬件架构。

第二节 Nachos虚拟机

【教学目标】掌握各种硬件的软件模拟方法

【教学要求】针对每种Nachos硬件，说明相应的软件模拟方法，分析虚拟机器与真实机器的异同，以及对后续系统实现的影响

【教学内容】结合machine包中的相关代码，介绍Nachos对时钟、中断、分时、磁盘的实现。

【教学和学习建议】讲授、讨论。

【难点重点】分析是否足够真实地模拟了硬件环境

**实验一 Nachos安装与调试（2学时）**

【教学目标】熟悉Nachos的安装及调试过程，了解Nachos对计算机硬件的模拟方法。

【教学要求】熟悉Nachos的安装过程，了解相关的makefile文件结构与内容，熟悉程序调试环境gdb，熟悉交叉编译的环境设置，了解Nachos实现的中断控制器、定时器等设备的工作过程。

【教学内容】安装Nachos及交叉编译器，熟悉Nachos中makefile以及gdb使用，利用gdb跟踪几个模拟设备的工作过程。

【教学和学习建议】上机、讨论。

【难点重点】对Nachos的安装过程，基于gdb的调试过程；交叉编译器的理解

1. **线程管理（4学时授课+6学时上机）**

本章的目标：

1. 描述Nachos线程的概念及实现
2. Nachos线程上下文切换
3. Nachos线程之间的同步

第一节 线程包

【教学目标】Nachos线程的管理。

【教学要求】了解Nachos线程包的各项功能。

【教学内容】线程创建、线程切换、调度状态、执行模式，上下文切换。

【教学和学习建议】授课、讨论。

【难点重点】从整体上把握线程包的功能及其之间的关联。

第二节 Nachos线程创建

【教学目标】了解Nachos线程创建的过程。

【教学要求】分析源代码，理解Nachos线程的概念及其创建过程。

【教学内容】结合main.cc，thread.cc，system.cc，分析系统初始化及线程的创建方法。

【教学和学习建议】授课、讨论。

【难点重点】Nachos内核初始化以及线程的创建过程。

第三节 Nachos线程调度及上下文切换

【教学目标】理解Nachos线程调度及线程切换程序的运行过程。

【教学要求】分析Nachos中线程调度算法、实现过程，以及上下文切换的具体实现。

【教学内容】结合scheduler.cc及switch.s，分析其设计实现方法。

【难点重点】线程切换的工作过程及实现方法。

第四节 Nachos线程同步

【教学目标】理解Nachos中信号量工作原理及线程同步机制。

【教学要求】分析Nachos中信号量及管程的实现方法，并利用其实现进程之间的同步。

【教学内容】结合synch.cc及synchlist.cc，分析Nachos中信号量及管程的构造方法，并利用信号量实现生产者-消费者模型。

【教学和学习建议】授课、讨论。

【难点重点】信号量及管程的工作机制及实现方法

实验二 Nachos中线程的上下文切换（2学时）

【教学目标】了解Nachos中线程上下文切换的工作机理

【教学要求】通过gdb调试器跟踪Nachos中内核线程的切换过程

【教学内容】将nachos装入到gdb，在gdb中适当位置设置断点，利用gdb提供的命令跟踪Nachos中线程的切换过程

【教学和学习建议】上机、讨论。

【难点重点】上下文的切换过程

实验三 信号量同步（4学时）

【教学目标】了解Nachos中利用信号量实现线程同步的工作机理

【教学要求】利用信号量机制完成生产者-消费者问题，并跟踪其运行过程

【教学内容】分析main.cc、prodcons++.cc，ring.cc、ring.h与synch.cc文件，利用Nachos实现的同步接口，编程实现生产者-消费者问题，并跟踪其运行过程。

【教学和学习建议】上机、讨论。

【难点重点】信号量、同步

1. 文件系统管理（4学时授课+10学时实验）

本章目标：

1. 了解文件系统的工作原理
2. 了解文件系统管理的实现、相关数据结构的组织及其关联
3. 完善Nachos文件系统，使其能够支持文件的扩展功能

第一节 文件管理包

【教学目标】Nachos文件系统的管理。

【教学要求】了解Nachos文件系统管理包的各项功能。

【教学内容】文件创建、文件系统结构等。

【教学和学习建议】授课、讨论。

【难点重点】从整体上把握文件管理包的功能以及类之间的关联。

第二节 文件系统的功能与实现

【教学目标】了解Nachos文件系统的管理功能，相关的数据结构。

【教学要求】了解Nachos文件系统的现有功能，实现方法

【教学内容】分析fstest.cc、synchdisk.h、openfile.h、filesys.h、directory.h、filehdr.h等文件，了解Nchos文件系统的设计、实现方法及相关的数据结构（如文件目录表、FCB、inode、Bit map等）

【教学和学习建议】授课、讨论。

【难点重点】文件系统相关数据结构及其之间的数据关联

实验四 Nachos文件的组织与结构（2学时）

【教学目标】通过跟踪调试nachos文件系统，掌握Nachos文件系统的工作机理

【教学要求】了解Nachos中文件系统所实现的功能、实现方法及文件系统的相关数据结构

【教学内容】在分析fstest.cc、synchdisk.h、openfile.h、filesys.h、directory.h、filehdr.h等文件的基础上，结合hexdump -C filename”或“od –c filename”等命令，了解Nchos文件系统的实现细节及相关的数据结构（如文件目录表、FCB、inode、Bitmap等）。

【教学和学习建议】上机、讨论。

【难点重点】文件系统的磁盘布局及相关的数据结构

实验五 可扩展文件的设计与实现（8学时）

【教学目标】对nachos中基本文件系统功能进行完善，设计实现文件的动态扩展功能

【教学要求】修改文件系统的相关代码，以支持根据文件的大小动态分配磁盘块数。

【教学内容】在实验四的基础上， 阅读修改OpenFile::WriteAt、FileHeader::AppendAllocate等方法，使文件系统具有根据文件大小动态扩展的功能。

【教学和学习建议】上机、讨论。

【难点重点】将文件系统的相关数据结构进行整合，实现文件的可扩展功能。

**第四章 应用程序与内存管理（4学时授课+14学时上机）**

本章目标：

1. 理解用户程序和操作系统的关系
2. 理解应用程序的加载过程及执行过程。
3. 完善Nachos，使其支持多进程管理。
4. 实现相应的系统调用。

第一节 内存管理包

【教学目标】Nachos内存管理

【教学要求】了解Nachos内存管理包的各项功能。

【教学内容】Nachos应用程序及内存管理。

【教学和学习建议】授课、讨论。

【难点重点】从整体上把握内存管理包的全部功能。

第二节 Nachos应用程序

【教学目标】创建Nachos应用程序的方法及应用程序调用内核提供服务的方法。

【教学要求】如何基于交叉编译程序建立Nachos能够运行的MIPS应用程序。

【教学内容】交叉编译环境的建立，应用程序结构，应用程序转换器coff2noff.c；

【教学和学习建议】授课、讨论。

【难点重点】Nachos应用程序的建立过程及系统调用

第三节 Nachos地址空间的管理

【教学目标】理解操作系统加载应用程序的过程，并为其分配内存空间的方法。

【教学要求】结合代码分析Nachos对存储器的管理方法。

【教学内容】分析addrspace.cc、progtest.cc等文件，理解Nachos如何将一个用户程序加载到内存，并为其分配相应的内存空间。

【教学和学习建议】授课、讨论。

【难点重点】应用程序的加载过程、内存空间的分配及虚实地址的转换方法。

第三节 Nachos对应用程序的执行过程

【教学目标】理解Nachos启动执行应用程序的过程。

【教学要求】结合代码说明Nachos启动应用程序及解析执行应用程序指令的方法。

【教学内容】分析machine/machine.cc、userprog/progtest.cc等文件，介绍Nachos将一个用户程序加载到内存并为其分配相应的内存空间后，如何启动该应用程序并解析执行其指令的过程。

【教学和学习建议】授课、讨论。

【难点重点】Nachos的CPU对应用程序指令的解析执行过程

第四节 Nachos多进程并发技术的设计与实现

【教学目标】理解操作系统对多道程序设计技术的设计与实现方法。

【教学要求】修改Nachos代码，提供相应的系统调用接口，使Nachos具备多进程并发执行的功能。

【教学内容】分析修改addrspace.cc及相关程序，实现系统调用exec()及exit()，使Nachos能够为每道程序分配独立的内存空间，并基于页表实现虚实地址的转换，在进程正常退出时能够回收其释放的内存空间。

【教学和学习建议】授课、讨论。

【难点重点】内存空间的分配及内存保护措施。

实验六 用户程序与系统调用分析（2课时）

【教学目标】理解用户程序加载执行过程，系统调用的工作原理。

【教学要求】理解如何创建用户进程，用户进程是如何利用系统调用与Nachos系统内核进行交互，Nachos系统调用是如何实现的。

【教学内容】 创建Nachos应用程序的方法；在分析addrspace.cc、progtest.cc等文件的基础上，理解Nachos何加载并执行应用程序的方法；理解内核与应用程序中系统调用的交互过程。

【教学和学习建议】上机、讨论。

【难点重点】应用程序的加载过程，为应用程序分配内存空间的方法，解析执行用户程序机器指令的过程。

实验七 地址空间扩展（6课时）

【教学目标】理解操作系统多道程序设计技术的工作原理、内存分配方法及虚实地址的转换方法。

【教学要求】理解页表生成机制，修改nachos空间分配代码，实现对Nachos对多进程空间分配的支持。

【教学内容】熟悉页式内存管理方法，分析修改addrspace.cc，为每道程序可分配独立的内存空间；

【教学和学习建议】上机、讨论。

【难点重点】虚实地址的转换及地址分配方法。

实验八 用户程序与系统调用实现（6课时）

【教学目标】理解多道程序设计技术的实现，系统调用的实现方法。

【教学要求】扩展nachos系统调用相关代码，实现exec()和exit()系统调用，使Nachos支持多线程并发执行。

【教学内容】在实验七的基础上，实现exec()及exit()系统调用，使Nachos可同时加载多个应用程序并正常退出，使Nachos可支持多道程序设计技术，并对实验七的设计方案的合理性进行验证。

【教学和学习建议】上机、讨论。

【难点重点】多道程序设计技术的实现方法。

**五、教学要求对应关系**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 个人与团队 | 使用现代工具 | 综合应用 | 研究 | 沟通 | 设计/开发解决方案 |
| 第一章 |  | X | X |  | X | X |
| 第二章 | X | X | X |  | X | X |
| 第三章 | X | X | X | X | X | X |
| 第四章 | X | X | X | X | X | X |

**六、考核及成绩评定方式**

【考核方式】：实验报告、实验演示、面试

【成绩评定】：实验报告（30%）、实验演示+面试（70%）

【考试大纲】

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **考试目的** | | 1.考察学生掌握和应用知识的能力**√** 2.评价教师教学质量 3.分级教学 4.招生 | | | | | | | | |
| **考试对象** | | 年级： 专业：计算机科学与技术 | | | | 卷面总分 | | | 100 | |
| **考试方式** | | 1.闭卷2.开卷 3.上机**√** 4.综述 5.论文 6.设计**√** 7.其它（ 面试 ）**√** | | | | | | | | |
| **考试方法** | | 1.笔试 2.口试**√** 3.实际操作**√** 4.其它（ ） | | | | | | | | |
| **试卷来源** | | 1.试题库 2.试卷库 3.校内统一命题 4.校外教师命题 5.任课教师命题**√** | | | | | | | | |
| **试题难易度** | | 1.较容易（ 30 ）% 2. 中等难度（ 50 ）% 3.较大难度（ 20 ）% | | | | | | | | |
| **编**  **题**  **计**  **划** | 题 类  量 型  内 %  容 | | | 记忆 | 理解  分析 | | 综合  应用 | 提高  扩展 | | 合计 |
| 第一章 | | | 5 | 0 | | 0 | 0 | | 5 |
| 第二章 | | |  | 5 | | 5 | 0 | | 10 |
| 第三章 | | |  | 15 | | 15 | 10 | | 40 |
| 第四章 | | |  | 10 | | 20 | 15 | | 45 |
| 合 计 | | | 5 | 30 | | 40 | 25 | | 100 |
| **阅卷方法：** | | | 1.微机阅卷 2.流水阅卷 3.任课教师阅卷**√** | | | | | | | |
| **记分方式：** | | | 1.百分制 2.五级制**√** 3.二级制 | | | | | | | |
| 备注：设计报告成绩占总成绩的30%，实验演示及面试成绩占总成绩的70%。  总成绩 = 考核成绩 + 平时成绩 | | | | | | | | | | |

**七、教材及参考资料**

【教材】

1．操作系统课程设计指导书（Nachos, C++）

2. Operating System Concepts, ninth Edition, Abraham Silberschatz, Wiley, 2012

【参考书】

1. Linux内核设计与实现，Robert Love, 译 陈莉君

2. 深入理解Linux内核，Daniel P. Bovet, 译 陈莉君

3. Linux内核的设计艺术，新设计团队，机械工业出版社

4. Linux内核完全解析0.11，赵炯

5．几个参考网站

C++:http://homes.cs.washington.edu/~tom/nachos/

C++:http://people.cs.uchicago.edu/~odonnell/OData/Courses/CS230/NACHOS/reading-code.html

C++:http://www.cs.duke.edu/~narten/110/nachos/main/main.html

Java: http://http.cs.berkeley.edu/~tea/nachos/nachos.ps

Java:http://www-inst.eecs.berkeley.edu/~cs162/sp14/

https://github.com