《操作系统》课程中英文简介

Operating Systems

课程代码：070843A **Course Code：**070843A

课程名称：操作系统 **Course Name：**Operating Systems

学时：48 **Periods：**48

学分：3 **Credits：**3

考核方式：考试 **Assessment：**Examination

先修课程： **Preparatory Courses：**

计算机导论 Introduction to Computer Science

程序设计基础 Programming Foundation

数据结构 Data Structure

计算机原理与汇编语言 Computer Organization & Assemble Language

《操作系统》是为计算机科学与技术专业本科生开设的学科基础课。通过本门课程的教学，使学生了解操作系统发展史、特征和分类；了解文件系统；掌握操作系统内部结构和工作原理；掌握处理机调度和内存管理方法；熟练运用C语言探索多进程，多线程编程的乐趣。教学过程中，通过理论讲解和编程实践等手段，使学生能够深入理解和运用操作系统，并为学生进一步学习后续课程，以及今后的学习与工作打好基础。

本课程是一门理论+实践的课程，它阐述了（1）操作系统的基本概念、基本功能及研究操作系统的方法；（2）进程、线程等基本概念；（3）同步、互斥、死锁等问题；（4）进程间通信，包括管道、信号、共享内存、消息队列等；（5）操作系统计算机软硬件资源的管理过程和方法，包括处理机调度、进程管理、存储管理等；（6）文件系统等内容。在实践上主要涉及使用C语言进行多进程、多线程编程，锁与信号量等同步互斥的运用，进程间通信（管道、信号、共享内存、消息队列等）编程等。

"Operating Systems" is a course for undergraduates majoring in computer science and technology. By learning this course, the students should understand the history, properties and types of operating systems; understand file systems; master the internal structure and principles of operating systems; master processor scheduling and memory management methods; proficiently use C language to explore programming of multiprocessing and multithreading. In the teaching process, through theoretical explanations and programming practices, students can deeply understand and use the operating system, and lay a solid foundation for students to further study follow-up courses, as well as future study and work. In the teaching process of theoretical explanations and programming practices, students would deeply understand and use the operating system, and lay a solid foundation for students to further courses, as well as future study and work.

This course is a theoretical and practical course. It explains (1) the basic concepts, basic functions and researching methods of operating system; (2) basic concepts of processes and threads; (3) synchronization, mutual exclusion, deadlock and other issues; (4) inter-process communication, including pipes, signals, shared memory, message queues, etc.; (5) management methods of computer software and hardware resource, including processor scheduling, process management, storage management, etc.; (6) file system and other contents. In practice, this course mainly involves the use of C language for multiprocessing and multithreading programming, the use of synchronization and mutual exclusion such as locks and semaphores, and the programming of inter-process communication (pipes, signals, shared memory, message queues, etc.).