



操作系统查漏补缺

East China University of Science and Technology

目录

1	计算机系统概述	2
1.1	操作系统的基本概念	2
1.2	操作系统发展历程	2
1.3	操作系统运行环境	2
1.4	操作系统结构	3
2	进程与线程	4
2.1	进程与线程	4
2.2	处理机调度	4

Chapter 1

计算机系统概述

1.1 操作系统的基本概念

多任务处理和多重处理之间的区别: 多任务处理指的是操作系统允许多个进程都处在运行的状态 (这里指的是广义的运行), 多重处理基于多核CPU, 指的是同一时刻能有多个程序在CPU上执行.

数据库系统是应用软件, 数据库管理系统是系统软件.

1.2 操作系统发展历程

实时操作系统不存在交互功能, 而分时操作系统是一定具有人际交互功能的.

多任务操作系统具有并发和并行的特点, 这里的并行指的是处理机和设备之间并行.

多道程序操作系统相比于单道程序操作系统的优点: CPU利用率提升, 吞吐量提升, 设备的利用率升高 (引入了中断处理).

1.3 操作系统运行环境

通道技术是一种特殊的处理器技术, 是独立于CPU的, 可以并行.

中断处理程序一定是在内核态执行的.

中断处理程序一定是 os 程序, 中断程序有可能是 os 程序, 也有可能是用户程序.

用户态到核心态的转换是由硬件实现的.

在核心态下才能执行的指令举例: 关中断, 清内存, 置时钟, 修改用户管理权限, 修改段表或者页表, 分配系统资源.

中断处理过程中最重要的两个寄存器是 PC 和 PSW(程序状态字寄存器), 这两个寄存器由中断隐指令自动保存, 另外寄存器的指令由操作系统保存.

1.4 操作系统结构

微内核相比于宏内核的优势: 可移植性, 可扩展性, 可靠性. 劣势: 执行效率低.

Chapter 2

进程与线程

2.1 进程与线程

进程是一个具有独立功能的程序关于某个数据集合的一次运行活动。

线程包含CPU现场, 可以独立执行程序. 而且同一个进程下的各个进程之间切换CPU线程信息相比于进程切换来说需要保存的更少.

同一个进程内的线程是**没有自己的地址空间**的. 他们共享进程的地址空间, 只有一丁点自己的资源. 基于数据段进行通信.

进程的封闭性: 封闭性的意思是程序的执行结果**只取决于进程本身**. 程序失去并发性的意思就是并发进程共享变量, 其执行结果与速度有关.

C语言程序中的内容和相关的数据结构在进程结构中的哪一段: 全局赋值变量和常量放在正文段, 函数调用的实参传递值和未赋值的局部变量放在栈段, 要求`malloc()`动态分配的区域放在堆段. 进程的优先级放在PCB中.

降低进程优先级和升高进程优先级的时机: 应当提高优先级的事件有进程长期处于就绪队列, 进程刚完成IO操作进入就绪队列; 应当降低优先级的事件有进程的时间片刚用完. 不用调整优先级的事件有进程从就绪态转为运行态.

线程也有不能共享的东西: 栈指针, 进程中的线程共享进程的全部资源, 但是进程中的栈指针对其他的线程是透明的, 不能和其他线程共享.

2.2 处理机调度

作业和进程的区别: 作业是用户提交的, 以用户的任务为单位; 进程是系统自动生成的, 以操作系统控制为单位.

进程在临界区的时候处理机是可以进行进程调度的, 只要不破坏临界资源的使用规则.(非常坑的一道题)

在考虑优先级调度算法的时候, 一般来说, **IO**型作业的优先权高于作业的优先权, 这是由于**IO**操作需要及时的完成, 他们有很长时间保存所要输入输出的数据, 所以, **IO**繁忙型作业应该有更高的优先级.