# 操作系统引论

## 操作系统的目标和功能

方便性，有效性（提高系统资源利用率，提高系统吞吐量），可扩展性，开放性

## 操作系统发展过程

单道批处理，多道批处理，分时系统，实时系统，集群，微机操作发展

操作系统的基本特征（并发【引入进程】，共享，异步，虚拟）

## 操作系统的主要功能

* 处理机管理（进程控制，进程同步，进程通信，调度）
* 存储器管理（内存分配，内存保护，地址映射，内存扩充）
* 设备管理（设备分配管理）
* 文件管理（文件存储空间，目录管理，读写保护）
* 操作系统和用户接口（用户，程序）
* 现代操作系统新功能（网络，多媒体等）

# 进程描述和控制

程序并发引入进程，并发会导致程序 间断性，失去封闭性，不可再现性

## 进程描述

* 进程定义：进程是程序的一次执行；进程是一个程序机器数据在处理机上顺序活动；进程是具有独立功能，系统调度和分配资源的的独立单位。系统进程和用户进程
* 进程的独特性：动态，并发，独立，异步
* 程序和进程：动静，程序是构成进程的组成部分，进程由PCB，程序和数据组成。
* 进程的基本状态：（创建，终止）就绪，阻塞，运行
* 挂起操作：终端用户需要，父进程需要，对换需要，负荷调节保证实时任务执行

进程管理中的数据结构

进程控制快PCB作用：作为独立运行基本单位标志，提供进程管理所需信息，进程调度所需信息，和其他进程进行同步和通信

**进程控制块信息**

* + 进程标识符（外部标识符PID，内部标识符端口）
  + 处理机状态（通用寄存器，指令计数器，程序状态字PSW，用户栈指针）
  + 进程调度信息：进程状态，进程优先级，进程调度所需其他信息，事件
  + 进程控制信息：程序和数据地址，进程同步和通信机制，资源清单，链接指针
  + 进程组织方式：线性方式，链式方式，索引方式

## 进程创建

引起进程创建的事件：用户登录，作业调度，提供服务，应用请求

进程创建过程：

1. 申请空白PCB
2. 为新进程分配资源
3. 初始化进程控制块
4. 如果进程就绪队列接纳新进程，将新进程插入就绪队列

进程的终止：

1. 正常结束 2.异常结束3.外界干预

进程阻塞和唤醒

1. 请求系统服务未满足
2. 启动某种操作阻塞当前进程
3. 新数据尚未到达
4. 无工作可做：系统进程

## 进程同步

基本概念：互斥竞争，同步协作

临界资源（一次只允许一个进程使用的资源）进入区，临界区，退出区，剩余区

同步机制原则：空闲让进，忙则等待，有线等待，让权等待

**进程间通信**

共享存储器 管道通信，消息传递系统，客户服务器socket方式

## 线程的基本概念

线程的引入是为了简化进程内通信，以最小开销提高进程内的并发程度。

# 处理机调度

## 处理机调度的层次

高级调度（作业调度）

中级调度（和挂起有关）

低级调度（进程调度）：方式分为抢占式和非抢占式，抢占式采用优先级原则，短进程优先，时间片原则。

进程调度的工作任务：保存处理机现场，按照某正算法选取进程，分配处理机资源

## 作业与作业调度

作业不仅包含程序和数据，还配有一份还说明书JCB，三个阶段收容阶段，运行阶段，完成阶段。

主要算法有先来先服务，短作业优先，优先级调度算法，高响应比优先 （等待时间+要求服务时间/要求服务时间）

## 死锁

* 计算机中资源问题，消耗性资源(数据消息)，可抢占式（cpu，内存不引起死锁），不可抢占式（光驱打印机），可重用的（外设资源）
* 计算机系统中的死锁：竞争不可抢占资源，竞争消耗性资源，进程间顺序推进不当引起
* 产生死锁的四个条件:互斥条件，请求和保存，不可抢占，循环等待

预防死锁：就是破坏死锁的四个条件

破坏请求和保存1.所有进程运行前一次性申请整个运行过程所需全部资源（优点：简单缺点：资源浪费，产生饥饿）2.运行初期所需资源，在逐步释放

破坏不可抢占：当一个程序保存不可抢占资源，提出新需求不满足话，必须释放已经保持的资源

破坏循环等待：对资源线性排序，赋值不同的序号，按照序号提出

避免死锁的策略：系统安全（安全序列是指一个进程序列{P1，…，Pn}是安全的，即对于每一个进程Pi(1≤i≤n），它以后尚需要的资源量不超过系统当前剩余资源量与所有进程Pj (j < i )当前占有资源量之和。），银行家算法 （可用资源，最大需求，已分配资源，需求矩阵资源）

# 存储器管理

## 程序装入

程序装入链接步骤：编译（源程序到目标模块）-链接（目标到装入模块）-装入（到内存）

程序装入：绝对装入，可重定位装入，运行时装入

程序的链接：静态链接，装入时动态链接，运行时动态链接

## 连续分配存储管理方式

静态分区：分区大小相同，多个小分区适量中等分区，少量大丰区（有内部碎片无外部碎片），动态分区：分区大小和分区数可变，根据作业大小多少动态划分。

基于顺序搜索动态分区：首次适应（顺序找到一个就使用），循环首次适应（类似哈希让内部碎片分配更加均匀），最佳适应算法（先找小分区，尽量不用大分区），最坏适应算法（直接找大分区，最后造成大分区越来越少）

基于索引搜索动态分区：快速适应算法，哈希算法

动态可重定向分区：动态运行时装入，内存地址=相对地址+起始地址

## 分段式管理存储

分段式管理中，作业地址空间被划分为若干段，每个段是一组完整的逻辑信息，连续的地址空间，各段长度不等。

段号+段内地址，实现了从逻辑段到物理地址映射。(信息共享是主要优点)

分页分段区别:

* 页是信息的物理单位
* 页的大小固定且由系统固定
* 分页的用户程序地址是一维空间
* 通常段比页大，段表比页表端，可以缩短查询时间，提高访问速度
* 分页是系统管理需求，分段是用户应用的需要。一条指令或一个才作数可能会跨越分页出，但是不能跨越两个分段处。

## 分页是管理

页号+位移量，页表大多驻留在内存中实现从页面好到物理地址的映射，访问两次内存，页表和页信息，

多级页表：页表太多，需要简化

# 虚拟存储器

离散型，多次性（一个程序分多次调入内存，只需当前运行部分），虚拟性，对换性

## 页面置换算法

抖动：刚被换出的页面很快又要被访问，需要重新调入

先进先出fifo，在内存中驻留时间最久的页面淘汰

最近未使用 last recently use

最少使用 Last frequently use

clock时钟算法（改进型，增加对修改位M的判断）

页面缓冲算法（空闲页面链表，修改页面链表）

# 文件系统

## 文件的逻辑结构

顺序文件，记录寻址，索引文件，索引顺序文件，直接文件和哈希文件。

文件共享：实际上就是快捷方式，符号链接实现文件共享，有向无环图DAG

## 文件目录

简单文件目录 （单级文件目录（查找满，不重名，不利于共享），两级文件目录）

树形结构目录