

# 操作系统

January 11, 2015

## Contents

<b>1</b>	<b>TODO 操作系统概述</b>	<b>4</b>
1.1	操作系统的地位, 作用, 发展历史, 特征, 功能 . . . . .	4
1.2	操作系统的分类及各类操作系统的特点 . . . . .	4
1.3	常用的操作系统 . . . . .	4
1.4	<b>TODO</b> 习题 . . . . .	4
<b>2</b>	<b>TODO 进程管理</b>	<b>5</b>
2.1	进程的概念, 特性 . . . . .	5
2.2	进程的状态及其转换 (具有挂起状态) . . . . .	5
2.3	PCB 的作用, 内容, 组织方式 . . . . .	5
2.4	各种进程控制的原子 (创建, 终止, 阻塞, 唤醒, 挂起) . . .	5
2.5	线程的概念, 进程与线程的联系与区别 . . . . .	6
2.6	进程同步的概念 . . . . .	6
2.7	临界资源, 临界区的概念 . . . . .	6
2.8	同步机制原则 . . . . .	6
2.9	同步与互斥的各种方法 (软件, 硬件, 信号量 (集)) . . . .	7
2.10	经典同步问题 . . . . .	7
2.11	<b>TODO</b> 调度方式 (抢占, 非抢占式), 类型 (高, 中, 低) . .	8
	2.11.1 方式 . . . . .	8
	2.11.2 <b>TODO</b> 类型 . . . . .	8
2.12	各种调度算法 (FCFS, SJF, RR, 优先权法, 多级反馈轮转) .	8
2.13	死锁的概念, 产生原因, 必要条件, 处理方法 . . . . .	8
2.14	银行家算法, 死锁定理 . . . . .	9
2.15	<b>TODO</b> 习题 . . . . .	9

<b>3</b>	<b>TODO</b>	存储器管理	<b>9</b>
3.1		连续分配，动态分配算法，回收算法	9
3.1.1		连续存储管理	9
3.1.2		动态分配算法	9
3.1.3		回收算法	9
3.2		地址重定位的概念，紧凑	10
3.2.1		概念	10
3.2.2		紧凑	10
3.3		页式，段式，段页式管理方式，数据结构，地址变换过程	10
3.3.1		分页式	10
3.3.2		段式	10
3.3.3		区别	11
3.3.4		段页式	11
3.4		可重入码的概念	11
3.5	<b>TODO</b>	虚存的概念，虚存的实现方式	11
3.5.1		概念	11
3.5.2		实现方式	11
3.5.3	<b>TODO</b>	流程	11
3.6		各种页面置换算法及实现方式（FIFO，LRU，CLOCK）	11
3.6.1		OPT 最优算法	11
3.6.2		FIFO 先进先出算法	11
3.6.3		LRU 最近最久未使用算法	12
3.6.4		CLOCK 时钟算法	12
3.7		工作集的概念	12
3.8	<b>TODO</b>	习题	12
<b>4</b>	<b>TODO</b>	文件系统	<b>12</b>
4.1		文件的概念	12
4.2		文件的逻辑结构，物理结构，存取方法	12
4.2.1		逻辑结构	12
4.2.2		物理结构	12
4.2.3		存取方法	13
4.3		文件控制块，索引节点，树型目录结构	13
4.3.1		文件控制块	13
4.3.2		索引节点	13
4.3.3		树型目录结构	14
4.4		建立目录的过程，目录查询技术	14
4.4.1		建立目录的过程	14
4.4.2		目录查询技术	14
4.5		文件共享与保护措施	14

4.5.1	文件共享	14
4.5.2	保护措施	14
4.6	文件的打开过程，打开的作用，读写过程	15
4.6.1	打开过程	15
4.6.2	打开的作用	15
4.6.3	读写过程	15
4.7	外存分配方式（连续式，链接式，索引式）	15
4.7.1	连续式	15
4.7.2	链接式	15
4.7.3	索引式	15
4.7.4	组合空间分配	15
4.8	<b>TODO</b> 文件存储空间管理（表，链，位示图，成组链接）	15
4.8.1	空闲表	15
4.8.2	<b>TODO</b> 空闲链表	16
4.8.3	位示图法	16
4.8.4	<b>TODO</b> 成组链接法	16
4.9	改善文件系统性能的方法	16
4.9.1	文件缓存	16
4.9.2	提前读取	16
4.9.3	减少磁臂移动	16
4.10	文件系统应该具备的主要功能	16
4.11	<b>TODO</b> 习题	16
<b>5</b>	<b>TODO</b> 设备管理	<b>16</b>
5.1	I/O 控制方式（程序，中断，直接存储器存取，通道）	16
5.1.1	程序	16
5.1.2	中断驱动	17
5.1.3	DMA	17
5.1.4	通道	17
5.2	<b>TODO</b> 输入输出硬件组织	17
5.2.1	I/O 设备	17
5.2.2	<b>TODO</b> I/O 控制器	17
5.2.3	I/O 通道	17
5.2.4	总线系统	17
5.3	<b>TODO</b> 输入输出软件组织及各层次的处理过程	17
5.3.1	中断处理程序	17
5.3.2	设备驱动程序	17
5.3.3	设备独立性软件	17
5.3.4	用户层的 I/O 软件	17
5.4	<b>TODO</b> 缓冲区的管理方式	17

5.4.1	单缓冲和双缓冲 . . . . .	17
5.4.2	循环缓冲 . . . . .	17
5.4.3	缓冲池 . . . . .	17
5.5	<b>TODO</b> 设备分配流程 . . . . .	17
5.5.1	设备分配中的数据结构 . . . . .	17
5.5.2	设备分配时应考虑的因素 . . . . .	18
5.5.3	独占设备的分配程序 . . . . .	18
5.6	SPOOLing 技术 . . . . .	18
5.6.1	概念 . . . . .	18
5.6.2	特点 . . . . .	18
5.7	<b>TODO</b> 设备独立性 (无关性) 及实现 . . . . .	18
5.8	<b>TODO</b> 磁盘存储器管理 . . . . .	18
5.8.1	磁盘性能 . . . . .	18
5.8.2	磁盘调度 . . . . .	18
6	安全与保护 . . . . .	18
6.1	口令的身份认证 . . . . .	18
6.2	访问控制技术 . . . . .	19
6.2.1	访问控制矩阵 (访问表) . . . . .	19
6.2.2	访问权限表 . . . . .	19
6.2.3	保护域 . . . . .	19
6.3	没有绝对安全的系统 . . . . .	19
7	Task . . . . .	19
7.1	<b>TODO</b> 试卷的完成 . . . . .	19
7.2	<b>TODO</b> 页面置换算法 . . . . .	19
7.3	<b>TODO</b> 磁盘调度算法 . . . . .	19
7.4	<b>TODO</b> 相应的题目 . . . . .	19
	知识提纲	

## 1 **TODO** 操作系统概述

1.1 操作系统的地位, 作用, 发展历史, 特征, 功能

1.2 操作系统的分类及各类操作系统的特点

1.3 常用的操作系统

1.4 **TODO** 习题

- 计算机操作系统方便用户, 管理和控制计算机软硬件资源的系统软件。

## 2 TODO 进程管理

### 2.1 进程的概念，特性

进程是对正在运行的程序的一种抽象，是资源分配和独立运行的基本单位。

### 2.2 进程的状态及其转换（具有挂起状态）

运行状态 - \* 阻塞状态 \*（静止阻塞，活动阻塞） - \* 就绪状态 \*（活动就绪，静止就绪）

- 特别的，无法从阻塞状态直接转换成运行状态

### 2.3 PCB 的作用，内容，组织方式

#### 1. 作用

- 为了保存进程的各个状态，建立专用数据结构，即为 PCB

#### 2. 内容

- 进程标识，位置信息，状态信息，进程的优先级，进程现场保护区，资源清单，队列指针等

#### 3. 组织方式

- 线性表
- 链接表
- 索引表

### 2.4 各种进程控制的原语（创建，终止，阻塞，唤醒，挂起）

原语指由若干条指令组成，不可被打断。

1. 创建 申请空间 - 查找 PCB 表 - 分配 PCB 空间 - 分配资源 - 信息填入 PCB - PCB 插入就绪队列
2. 终止 找到 PCB - 没有，进入异常处理 - 有，撤销 PCB 链中该进程的名字以及所有的子进程
3. 阻塞 中断 CPU - 将 CPU 状态存入 PCB - 进程状态转变为阻塞 - 插入等待队列 - 执行调度程序
4. 唤醒 唤醒条件满足 - PCB 状态设置就绪，脱离阻塞队列 - 插入就绪队列

## 2.5 线程的概念，进程与线程的联系与区别

1. 概念 线程是进程中的一个实体，是被系统独立调度和分配的基本单位
2. 联系区别
  - (a) 调度 传统操作系统中，拥有资源的基本单位和独立调度的基本单位都是进程，而在引入线程的操作系统中，则把线程作为调度和分配的基本单位，而把进程作为拥有资源的基本单位，使传统进程的两个属性分开
  - (b) 并发性 不仅进程之间可以并发执行，一个进程的多个线程也可以并发执行
  - (c) 拥有资源 线程不拥有系统资源，但可以访问隶属进程的其他资源
  - (d) 系统开销

## 2.6 进程同步的概念

异步环境下并发执行的进程：

1. 同步 因直接制约而需要相互等待，相互合作，以达到各进程按相互协调的速度执行的过程，成为进程的同步
2. 互斥 因间接制约而导致交替执行的过程称为进程的互斥

## 2.7 临界资源，临界区的概念

某段时间内只能允许一个进程使用的资源称为临界资源，必须互斥执行的程序段称为某临界资源的临界区

- 进入区
  - 临界区
- 退出区

## 2.8 同步机制原则

- 空闲让进
- 忙则等待
- 有限等待
- 让权等待（不占用 CPU 资源），忙等待则占用 CPU 资源

## 2.9 同步与互斥的各种方法（软件，硬件，信号量（集））

### 1. 硬件方式

- 中断禁用
- 专用机器指令

### 2. 软件方式

- Dekker 算法，Peterson 算法无法实现让权等待

### 3. 信号量, P/V 操作

## 2.10 经典同步问题

### 1. 生产者消费者问题

#### (a) 生产者

- i. 有无空位
- ii. 锁资源
- iii. 释放资源
- iv. 增加可使用资源

#### (b) 消费者

- i. 有无可使用资源
- ii. 锁资源
- iii. 释放资源
- iv. 增加空位

### 2. 读者写者问题

#### (a) 读者

- i. 锁资源
- ii. 判断读者数量，以此判断是否锁写信号
- iii. 释放锁

#### (b) 写者

- i. 要求写信号
- ii. 写资源
- iii. 释放写信号

### 3. 哲学家就餐问题

- 左右两边同时可以拿叉子的时候，才拿
- 奇数号（1,3）的哲学家先拿左边的叉子，偶数号（0,2,4）的科学家先拿右边的叉子。

## 2.11 TODO 调度方式（抢占，非抢占式），类型（高，中，低）

### 2.11.1 方式

抢占方式即新来的优先级高，则立即执行，对于原本的进程，保存其状态

### 2.11.2 TODO 类型

## 2.12 各种调度算法（FCFS, SJF, RR, 优先权法，多级反馈轮转）

1. FCFS 先来先服务
2. RR 时间片轮转
  - 分配时间片段给任何一个进程
3. 短进程优先
  - 时间短的进程优先执行
4. 高响应比优先
  - 公式：优先权 = （等待时间 + 要求服务时间） / 要求服务时间
  - 需要计算响应比，会增加系统开销
5. 优先级法 进程的类型，运行时间，作业的优先级。
6. 多级反馈队列调度算法
  - (a) 放在不同等级的队列上进行执行
  - (b) 在每一个队列上执行时间片轮转

## 2.13 死锁的概念，产生原因，必要条件，处理方法

1. 概念 某个进程提出资源申请的时候，有关进程无法在外力的协助下，永远分配不到必需的资源而无法继续运行。
2. 产生原因
  - (a) 系统提供的资源不能满足每个进程的使用需求



(b) 在多道程序运行时，进程推进顺序不合法。

### 3. 必要条件

- (a) 互斥条件
- (b) 不剥夺条件
- (c) 请求且保持条件
- (d) 环路等待条件

## 2.14 银行家算法，死锁定理

1. 死锁定理 当且仅当系统资源分配图是不可完全简化时，系统中有死锁。
  - 特别的，可以完全简化的时候，系统成功结束，没有简化出来的进程就是参与死锁的进程
2. 银行家算法
  - (a) 安全状态 安全状态是指分配的资源，满足每个进程的资源的最大需求，则每个进程顺利完成。
  - (b) 简单理解 简单理解为：进程执行前，判断系统是否有足够的进程分配给该进程使得该进程能够成功执行完毕

## 2.15 TODO 习题

# 3 TODO 存储器管理

## 3.1 连续分配，动态分配算法，回收算法

### 3.1.1 连续存储管理

依次分配内存即可

### 3.1.2 动态分配算法

1. 首次适应算法 取第一个满足大小的空间
2. 最佳适应算法 取最接近的空间
3. 最差适应算法 取最不接近的空间（远大于所需量的空间）

### 3.1.3 回收算法

回收以后，指针如何移动的问题。书上没有，参考相关图

## 3.2 地址重定位的概念，紧凑

### 3.2.1 概念

把用户程序装入内存时，对有关指令的逻辑地址部分的修改称为地址重定位。

### 3.2.2 紧凑

经过一段时间的使用之后，内存中存在大量的碎片，于是需要紧凑处理。移动各用户分区中的程序，使他们集中于内存的一端，使碎片集中于另一端，从而将空闲的碎片连成一个较大的分区，供需求作业使用，为此，必须采用动态重定位技术。

## 3.3 页式，段式，段页式管理方式，数据结构，地址变换过程

### 3.3.1 分页式

1. 概念 通过查页表补全块号，构成实际地址
2. 分配与回收 位图法，链表法
3. 地址重定位 页内地址的长度由页大小决定，逻辑地址由硬件分成的两部分页号  $p$  和页内地址  $w$ 。例如：页大小 1KB，则逻辑地址的低 10 位 ( $2^{10} = 1\text{KB}$ ) 为页内地址

Table 1: 逻辑地址  
页号  $p$ (6 位) 页内地址  $w$  (10 位)

4. 具体过程

### 3.3.2 段式

1. 概念 以段为基本单位分配内存，每一段必须分配连续的内存空间，单各段之间不要求连续。

段号  $S$  段内位移  $W$

2. 地址重定位 检查段号，查段表，得出内存初始地址，与段内位移相加，得出物理地址

### 3.3.3 区别

1. 页是信息的物理单位，段是信息的逻辑单位。页减少外部碎片。
2. 分页式存储管理的作业地址空间是一维的，分段是二维的。
3. 页的大小固定，段的长度不定。
4. 分页的优点体现在内存空间的管理上，分段体现在地址空间的管理上

### 3.3.4 段页式

#### 1. 管理方式

- (a) 先分段作业地址空间进行段式管理
- (b) 后分页每段内在分成若干大小固定的页
- (c) 物理块大小
- (d) 分为三个部分

段号 (s) 段内页号 (p) 页内位移 (d)

### 3.4 可重入码的概念

也称纯代码，是一种允许多个进程同时访问的代码（函数式编程）

### 3.5 TODO 虚存的概念，虚存的实现方式

#### 3.5.1 概念

把内存与外存统一起来，形成一个存储器。

#### 3.5.2 实现方式

请求页式存储管理

#### 3.5.3 TODO 流程

### 3.6 各种页面置换算法及实现方式（FIFO, LRU, CLOCK）

#### 3.6.1 OPT 最优算法

#### 3.6.2 FIFO 先进先出算法

最先淘汰那些驻留在内存时间最长的页面。

### 3.6.3 LRU 最近最久未使用算法

一直未被使用的页面最先淘汰

### 3.6.4 CLOCK 时钟算法

判断页面的 R 位如果 R 位为 0，置换页面；如果 R 位为 1，清除 R 位，前移指针

## 3.7 工作集的概念

一个进程当前正在使用的页面的集合

## 3.8 TODO 习题

# 4 TODO 文件系统

## 4.1 文件的概念

存储在外部存储介质上的，具有符号名的一组相关信息的集合

## 4.2 文件的逻辑结构，物理结构，存取方法

### 4.2.1 逻辑结构

1. 记录式文件 记录式文件在逻辑上总是被看成一组顺序的记录集合，是一种有结构的文件组织，又分成定长记录文件和变长记录文件
2. 流式文件 流式文件又称为无结构文件，文件内部不划分记录，文件内部是由一组相关信息组合成的有序字符流，这种文件的长度直接按字节计算

### 4.2.2 物理结构

#### 1. 顺序文件

(a) 概念 把一个逻辑上连续的文件存放在连续编号的物理块中。

(b) 缺点

- 建立时就要确定其长度
- 不便于文件的动态扩充
- 可能出现外部碎片

#### 2. 链接文件

- (a) 概念 逻辑上连续的文件，分散地存放在不同的物理块中。
- (b) 特点
  - i. 仅适于对信息的顺序访问，不利于对文件的随机存取
  - ii. 每个物理块上增加了一个连接字，为信息管理添加了麻烦

### 3. 索引文件

- (a) 概念 系统为每个文件建立一个索引表，其中的表项指出该文件的各个物理块号。
- (b) 特点 具有多种优点，但是需要内存来提高速度

### 4. 多级索引文件 盘块号指向盘块号，从而扩大文件索引范围

#### 4.2.3 存取方法

1. 顺序存取方式 按照逻辑顺序一次读取文件内容
2. 随机存取方式 根据需要存取文件中的任何块和记录，先对文件进行选点 (seek)，到达指令位置再开始读写。

## 4.3 文件控制块，索引节点，树型目录结构

### 4.3.1 文件控制块

文件的文件目录项又称为文件控制块。(FCB)

1. 有关文件存取控制信息
2. 文件结构信息
3. 文件使用信息
4. 文件管理信息

### 4.3.2 索引节点

1. 概念 为了减少检索文件访问的物理块数，UNIX 文件系统把文件目录项中的文件名和其他管理信息分开，后者单独组成定长的数据结构，称为 \* 索引节点 \*，也称 i 节点。文件目录项组成的目录文件与普通文件一样对待，存储在文件存储器中。
2. i 节点表 文件存储器上的每一个文件，都有一个外存文件控制块（又称外存索引节点）i 节点与之对应，这些 i 节点被集中存放在文件存储器的 i 节点区。在使用的时候，内存会针对其建立一个内存 i 节点表（活动文件控制块或者活动 i 节点表）。

3. 使用 i 节点表的流程 使用一个文件，如果在活动 i 节点表中找不到其对应的 i 节点时，就申请一个内存 i 节点，把外存 i 节点的内容复制到内存 i 节点中，随之就使用这个内存 i 节点来控制文件的读写。通常，最后一个用户关闭此文件后，内存 i 节点的内容被写到外存 i 节点中，然后，释放内存 i 节点。

#### **4.3.3 树型目录结构**

1. 方便用户查找文件，又可以把不同类型和用途的文件分类
2. 允许文件重名
3. 利用树型结构关系，可以更方便的制定保护文件的存取权限，有利于文件的保护
4. 不能直接支持文件或目录的共享

### **4.4 建立目录的过程，目录查询技术**

#### **4.4.1 建立目录的过程**

建立文件控制块 FCB，标识目录文件，赋予目录文件信息，写入磁盘

#### **4.4.2 目录查询技术**

1. 线性检索法 在单级目录中，利用用户提供的文件名，用顺序查找法从文件目录表中找到指名文件的目录项
2. Hash 方法 windows 中的建立索引

### **4.5 文件共享与保护措施**

#### **4.5.1 文件共享**

#### **4.5.2 保护措施**

1. 口令
2. 加密
3. 访问控制

## 4.6 文件的打开过程，打开的作用，读写过程

### 4.6.1 打开过程

- 在目录上寻找该文件
- 活动 i 节点表增加该文件，（为该文件建立内存，从磁盘读入）
- PCB 获取 inode

### 4.6.2 打开的作用

减少磁盘 I/O，加快运行速度

### 4.6.3 读写过程

读写都在内存中进行，通过 inode 确认文件（一个文件有唯一的一个 inode），在最后一个用户完成操作后，写入磁盘

## 4.7 外存分配方式（连续式，链接式，索引式）

### 4.7.1 连续式

每一个文件占据了一个完整且连续的磁盘区域，但是会产生碎片

### 4.7.2 链接式

没有外部碎片，但是磁盘读写速度较低，需要来回移动磁头寻找磁道，必须为指针分配空间

### 4.7.3 索引式

每个文件都有一个索引块。

### 4.7.4 组合空间分配

一般盘块号为 4byte，盘块大小为 1K，由此计算出最多存放 256 个盘块号。

## 4.8 TODO 文件存储空间管理（表，链，位示图，成组链接）

### 4.8.1 空闲表

系统为外存上的所有空闲区建立一张空闲表，分配方法与内存的动态分配相似

#### **4.8.2 TODO 空闲链表**

空闲表块链和区链

#### **4.8.3 位示图法**

#### **4.8.4 TODO 成组链接法**

1. 设置空闲盘块号栈
2. 文件区中的所有空闲盘块

### **4.9 改善文件系统性能的方法**

#### **4.9.1 文件缓存**

#### **4.9.2 提前读取**

#### **4.9.3 减少磁臂移动**

#### **4.10 文件系统应该具备的主要功能**

1. 对文件存储器空间进行组织和分配
2. 负责文件的存储并对存入的文件进行保护和检索的系统
3. 具体为用户建立文件，存入，读出文件，控制文件存取，撤销文件等。

#### **4.11 TODO 习题**

## **5 TODO 设备管理**

依据习题进行筛选

### **5.1 I/O 控制方式（程序，中断，直接存储器存取，通道）**

#### **5.1.1 程序**

不再使用



### **5.1.2 中断驱动**

### **5.1.3 DMA**

### **5.1.4 通道**

## **5.2 TODO 输入输出硬件组织**

### **5.2.1 I/O 设备**

### **5.2.2 TODO I/O 控制器**

#### **1. TODO 概念**

#### **2. TODO 功能**

### **5.2.3 I/O 通道**

### **5.2.4 总线系统**

## **5.3 TODO 输入输出软件组织及各层次的处理过程**

### **5.3.1 中断处理程序**

### **5.3.2 设备驱动程序**

### **5.3.3 设备独立性软件**

### **5.3.4 用户层的 I/O 软件**

## **5.4 TODO 缓冲区的管理方式**

### **5.4.1 单缓冲和双缓冲**

### **5.4.2 循环缓冲**

### **5.4.3 缓冲池**

## **5.5 TODO 设备分配流程**

### **5.5.1 设备分配中的数据结构**

#### **1. 设备控制表**

#### **2. 控制器控制表**

#### **3. 通道控制表**

#### **4. 系统设备表**

#### **5.5.2 设备分配时应考虑的因素**

#### **5.5.3 独占设备的分配程序**

### **5.6 SPOOLing 技术**

#### **5.6.1 概念**

SPOOLing 技术，是在多道程序设计中将一台独占设备改造为共享设备的一种行之有效的技术。

#### **5.6.2 特点**

1. 提高了 I/O 速度
2. 将独占设备改造为共享设备
3. 实现了虚拟设备功能

### **5.7 TODO 设备独立性 (无关性) 及实现**

### **5.8 TODO 磁盘存储器管理**

#### **5.8.1 磁盘性能**

#### **5.8.2 磁盘调度**

## **6 安全与保护**

### **6.1 口令的身份认证**

口令保护容易理解，也容易实现。

- 口令的选择策略，内部存储，有效期和限制登录尝试次数等方面加强口令机制的安全

## **6.2 访问控制技术**

### **6.2.1 访问控制矩阵（访问表）**

### **6.2.2 访问权限表**

### **6.2.3 保护域**

## **6.3 没有绝对安全的系统**

## **7 Task**

### **7.1 TODO 试卷的完成**

### **7.2 TODO 页面置换算法**

### **7.3 TODO 磁盘调度算法**

### **7.4 TODO 相应的题目**