

# 操作系统复习@onevfall

## 1. zhj的知识点复习提问

### 进程管理部分

#### 1. 为什么要引入进程？

- 引入进程的目的是更好地**使多道程序并发执行，提高资源利用率和系统吞吐量。**

#### 2. 为什么要引入线程？

- 引入线程的目的则是**减小程序在并发执行时所付出的时空开销，提高操作系统的并发性能。**

#### 3. 从调度性、并发性、拥有的资源以及系统开销等方面，区别和比较进程和线程？

- **调度性**：线程是**独立调度**的基本单位，进程是**拥有资源**的基本单位。在同一进程中，线程的切换不会引起进程切换。在不同进程中进行线程切换，如从一个进程内的线程切换到另一个进程中的线程时，会引起进程切换。
- **并发性**：在引入线程的操作系统中，**不仅进程之间可以并发执行，而且多个线程之间可以并发执行**，从而使操作系统具有**更好的并发性**，提高了系统的吞吐量。
- **拥有资源**：不论是传统操作系统还是设有线程的操作系统，**进程都是拥有资源的基本单位，而线程不拥有系统资源，但线程可以访问其隶属进程的系统资源。**
- **系统开销**：由于**创建或撤销进程时**，系统都要为之**分配或回收资源**，如内存空间、I/O设备等，因此操作系统所付出的**开销远大于**创建或撤销线程时的开销。类似地，在**进行进程切换时**，涉及当前执行过程CPU环境的保存及新调度到进程CPU环境的设置，而**线程切换时**只需保存和设置少量寄存器内容，**开销很小**。此外，由于同一进程内的**多个线程共享进程的地址空间**，因此这些线程之间的同步与通信非常容易实现，即**通信开销少**。

#### 4. CPU调度算法有哪些？

- 先来先服务调度算法
- 短作业优先调度算法
- 优先级调度算法
- 时间片轮转调度算法
- 多级反馈队列调度算法

#### 5. 多级反馈队列调度算法的分析？

- 是**时间片轮转调度算法和优先级调度算法的综合与发展**，通过**动态调整进程优先级和时间片大小**，多级反馈队列调度算法可以**兼顾多方面的系统目标——为提高系统吞吐量和缩短平均周转时间而照顾短进程；为获得较好的I/O设备利用率和缩短响应时间而照顾I/O型进程；同时，也不必事先估计进程的执行时间。**
- 优势：1. 终端型作业用户：短作业优先；2. 短批处理作业用户：周转时间较短；3. 长批处理作业用户：经过前面几个队列得到部分执行，不会长期得不到处理。

#### 6. 如何利用 P/V 操作实现多个进程之间的同步和互斥？

- 实现同步：为临界资源设置一个互斥信号量mutex，初值为1；并在每个进程中，将临界区代码置于P(mutex)和V(mutex)原语之间。
- 实现互斥：前驱关系，为每个前驱关系设置一个同步信号量S12，其初值为0；将V(S12)置于前驱进程的尾部，P(S12)置于后驱进程的头部。

#### 7. 引起死锁的四个特征是什么？【分析：前三个特征无法改变，由资源分配机制所决定的。只有第4个特征是偶然发生的，可能改变】

- **互斥**：至少有一个资源必须处于互斥模式，即一次只有一个进程使用。如果另一资源申请该资源，那么申请进程必须延迟到该资源释放为止。
- **持有并等待**：一个进程必须占有至少一个资源，并等待另一个资源，而该资源为其他进程所占有。
- **非抢占**：资源不能被抢占，即只有进程完成其任务之后，才会释放其资源。
- **循环等待**：进程{P0, P1, ..., Pn}, P0等待的资源为P1所占有，P1等待的资源为P2所占有，Pn-1等待的资源为Pn所占有，Pn等待的资源为P0所占有。

#### 8. 资源分配图的方法判定死锁？

- 如果图不包含环，则不存在死锁。
- 如果图包含环，则
  - 如果每种资源类型只有一个实例，则死锁。
  - 如果每种资源类型存在若干个实例，则只是有可能会发生死锁。

#### 9. 动态分区分配的分配算法：

- 最先适配算法
  - 算法思想：按**分区先后次序**从头查找，找到符合要求的第一个分区。
  - 算法实质：尽可能**利用存储区低地址空闲区**，尽量在高地址部分保存较大空闲区，以便一旦有分配大空闲区要求时，容易得到满足。
  - 算法优点：分配简单，合并相邻空闲区比较容易
  - 算法缺点：查找总是从表首开始，前面空闲区往往被分割的很小，满足分配要求的可能性低，**查找次数较多**。
- 循环最先适配算法
  - 算法思想：按**分区先后次序**，从**上次分配的分区起**查找（到最后分区时再回到开头），找到符合要求的第一个分区。
  - 算法特点：算法的**分配和释放的时间性能较好**，使**空闲分区分布得更均匀**，但**较大的空闲分区不易保留**。
- 最佳适配算法
  - 算法思想：在所有**大于或者等于要求分配长度**的空闲区中挑选一个**最小的分区**，即对该分区所要求分配的大小来说是最合适的。**分配后，所剩余的块会最小**。
  - 算法实现：空闲存储管理表采用**从小到大的顺序结构**。
  - 优点：较大的空闲分区可以被保留。
  - 缺点：空闲分区是**按大小**而不是按地址顺序**排列**的，因此释放时，要在整个链表上搜索地址**相邻的空闲区**，**合并**后，又要**插入**到合适的位置。
- 最坏适配算法
  - 算法思想：分区时**取**所有空闲区中**最大**的一块，把**剩余的块再变成**一个新的小一点的**空闲区**。
  - 算法实现：空闲区按由大到小排序
  - 优点：分配时，只需查找一次就可成功，分配算法很快。
  - 缺点：**最后剩余分区会越来越小，无法运行大程序**。
- 以上的算法存在碎片问题，有提出“紧凑技术”来解决，但仍不够好，从根源上解决引出了离散分配方式。

#### 10. 页式存储管理的硬件支持

- 页表始址寄存器
- 页表长度寄存器
- 联想寄存器——快表

- 为缩短查找时间，可将页表从内存装入到关联存储器(TLB)，按内容查找，即逻辑页号——>物理页号
- 快表存储了一部分从逻辑地址到物理地址的映射。若做地址映射，则先查快表，若其中有映射，则直接调用；若没有则再在页表查询映射关系。

#### 11. 页面置换算法：

- 先进先出算法 (FIFO)
  - 选择建立最早的页面被置换
  - 评价：性能较差，抖动现象，Belady现象
- 最佳算法 (OPT)
  - 选择“未来不再使用的”或“在离当前最远位置上出现的”页面被置换。
  - 评价：理想算法
- 最近最久未使用算法 (LRU)
  - 使用离过去最近作为不远将来的近似，置换最长时间没有使用过的页。
  - 评价：性能接近最佳算法
- 最不常用算法 (LFU)
  - 选择到当前时间为止被访问次数最少的页面被置换
  - 每页设置访问计数器，每当页面被访问时，该页面的访问计数器加1
  - 发生缺页中断时，淘汰计数值最小的页面，并把所有计数清零
- 轮转算法 (clock)
  - 也称最近未使用算法 (NRU)，是LRU和FIFO的折衷。
  - 实现方法
    - 每页标志位use，若该页被访问则置use=1
    - 置换时采用一个指针，从当前指针位置开始按地址先后检查各页，寻找use=0的页面作为被置换页，并将指针经过的页修改为use=0，最后指针停留在被置换页的下一个页。

#### 12. 磁盘调度算法列举

- 先来先服务
  - 优点：简单，公平
  - 缺点：效率不高，相邻两次请求可能会造成最内到最外的柱面寻道，使磁头反复移动，增加了服务时间，对机械也不利。
- 最短寻道时间优先
  - 优先选择距当前磁头最近的访问请求进行服务，主要考虑寻道优先
  - 优点：改善了磁盘平均服务时间
  - 缺点：造成某些访问请求长期等待得不到服务（不公平、饥饿）
- 扫描算法 (电梯算法)
  - 由最短寻道改进而来，一个方向上经过的所有最近点都能被扫描到【即单向的最短寻道】
  - 但会到达全局最远处和最近处（虽然没有要求到这里）
- 单向扫描算法
  - 相比于扫描算法，只会到达访问队列中要求的最高点和最低点

#### 13. 廉价磁盘冗余阵列RAID——SFT-III技术

- 技术特点：并行交叉存取
- 优点：可靠性高、磁盘I/O速度高、性能/价格比高

## 2. 问题集锦

1. 父进程不能与子进程共享虚拟地址空间，在创建子进程时会为子进程分配资源，如虚拟地址空间等。
2. 临界资源一次只能为一个进程所用。【临界资源：在一段时间内只允许一个进程访问的资源。又称独占资源。】
3. 页表存放在内存中，属于进程的现场信息。（页表始址会填入PCB）
4. 何为调度公平性？【由此延伸需要记一下公平性】
  - 一个I/O请求在有限时间内满足
5. 操作系统通常应包括下列**功能模块**：
  - (1) **进程管理**。当多个程序同时运行时，解决处理器(CPU)时间的分配问题。
  - (2) **存储器管理**。为各个程序及其使用的数据**分配存储空间**，并保证它们互不干扰。
  - (3) **设备管理**。根据用户提出使用设备的**请求进行设备分配**，同时还能**随时接收设备的请求**（称为中断），如要求输入信息。
  - (4) **文件管理**。主要负责文件的**存储、检索、共享和保护**，为用户提供**文件操作的方便**。
6. 操作系统**定义**：
  - 是指**控制和管理整个计算机系统**的硬件和软件资源，并合理的**组织和调度**计算机的工作和资源的**分配**，以提供给**用户和其它软件**方便的**接口和环境**，它是计算机系统中最**基本的系统软件**
7. 操作系统的**特征**：
  - 并发性
  - 共享性
  - 异步性
  - 虚拟性
8. 设计目标
  1. 答案1：
    - **计算机系统资源的管理者——四大功能**
    - 作为**用户与计算机硬件系统**之间的**接口**，方便用户快捷、可靠的操纵计算机硬件并运行自己的程序
    - 用作**扩充机器**
  2. 答案2：
    - 用户目标：OS应该便于使用、易于学习、可靠、安全与快速。
    - 系统目标：易设计、实现和维护；应该灵活、可靠、没有错误且高效。
9. 操作系统的软件体系结构发展阶段是 单一结构，核心层次结构，微内核结构
10. 微内核结构是以**微内核为OS核心**，以**客户/服务器为基础**，采用**面向对象程序设计的特征**，是当今最有发展前途的**OS结构**。
11. PCB：进程控制块，描述进程的基本情况和运行状态，是进程存在的唯一标志。
12. 进程由PCB、程序段和数据段组成。
13. **页表指出逻辑地址中的页号与所占主存块号的对应关系**。作用：页式存储管理在用动态重定位方式装入作业时，要利用页表做地址转换工作。
14. 快表就是存放在**联想寄存器的部分页表**。它起页表相同的作用。由于采用页表做地址转换，读写内存数据时CPU要访问两次主存。

15. 虚拟存储器 是指具有**请求调入功能**和**置换功能**，能从逻辑上对**内存 容量加以扩充**的一种**存储 器系统**

16. 特点：

- 不连续性：物理内存分配和虚拟地址空间使用的不连续性
- 部分交换：每次作业交换一小部分
- 大空间：提供大范围的虚拟地址空间，其总容量不超过物理内存和外存交换区容量

1. **虚拟扩充**：不是物理上而是逻辑上扩充了内存容量。

2. **部分装入**：每个作业不是全部一次性地装入内存，而是只装入一部分。

3. **离散分配**：不必占用连续的内存空间，而是“见缝插针”。

4. **多次对换**：所需的全部程序和数据要分成多次调入内存。

17.

## 20年OS题总结

---

1. 段式也没有解决碎片问题——>复习策略：需要再收束一下各章知识，当然，先以题代练

## 考前早上需要看的

---

1. 翻最后一节PPT磁盘
2. 复习作业题，速览
3. 看期末卷勾画处，速览
4. 看本md所有内容，速览