

一、选择题/填空题

1. 程序状态字反映了与处理器有关的系统的状态。
2. 磁盘输入输出时间包括寻找时间、延迟时间（旋转时间）、传送/传输时间。
3. 通道中断 CPU 利用率最高。
4. 对文件进行读操作时，以物理块为单位进行读取。
5. 使用扫描法对打印机的分配算法。
6. 在虚拟内存管理中、地址变换机构将逻辑地址变换为物理地址，形成该物理地址的阶段是 **装载**，形成该逻辑地址的阶段是：编译？链接？/有争议。
7. 文件控制块实现按名存取
8. 系统抖动是指数据页面频繁的调入调出现象。
9. 系统调用发出外设请求---->OS 发出 I/O 指令。
10. 批处理操作系统的调度算法的性能评价中通常不包括响应时间，包括吞吐量、周转时间和 CPU 利用率。
11. 最短作业优先可以使平均等待时间最小。
12. 控制内存中进程数量的是长程调度。
13. 进程的特性：动态、并发、独立、异步。
14. 文件的逻辑结构：堆结构、顺序结构和散列结构。
15. 文件的物理结构：连续结构、链接结构和索引结构。
16. 文件的读写方式：顺序读写、直接读写/随机读写、索引读写。

二、简答题

1. 什么是微内核？微内核有哪些功能？微内核的优点有哪些？
优点：扩展 OS,平台间移植和可靠性、面向对象、分布式
2. 什么是机制和策略？举例说明。
答：机制是提供什么样的功能(做什么)，策略是指如何实现这些功能(如何做)。
举例：解决死锁问题，死锁避免就是机制，银行家算法就是策略。
3. 解释 CPU 调度和分派。
·CPU 调度将进程从就绪队列取出，是多道程序操作系统的基础。通过在进程之间切换 CPU 可以提高计算机的吞吐率。
·分派是将选择的进程加载到 CPU, CPU 的控制权交给短期调度程序选择的进程。
4. 用户模式和内核模式存在的原因。
 - **安全性**：有助于防止用户程序对系统造成不可修复的破坏。用户程序在用户态下运行，受到严格的限制，不能直接操作关键系统资源。
 - **稳定性**：如果一个用户程序崩溃或出现错误，它不会影响整个系统的稳定性。内核态下运行的操作系统可以继续正常工作，而不受用户程序的干扰。
 - **性能**：内核态具有更高的权限，可以执行特权指令，如直接访问硬件。将这些操作限制在内核态有助于提高系统的性能，因为不需要为每个用户程序都提供相同的权限。
5. 死锁恢复的方案。
6. 内中断和外中断的区别。

内中断：指令执行过程中出现的【主动如 trap 指令，被动如溢出、越界和缺页等】
外中断：外部事件引起的，如时钟中断。

7. 文件目录的作用和文件目录结构。

作用：文件名到文件物理位置的映射-按名存取、快速检索文件目录、文件共享和允许文件重名。

组织结构：单级目录(查找费时、重名问题、不利于文件共享)，两级目录结构和多级目录结构。

8. DMA 控制器的工作原理

接管数据地址总线控制权--->DMA 控制器中的字节计数寄存器计数器减一
--->恢复 CPU 对数据地址总线的控制权----->循环输入传输过程----->发出中断信号

30、基于硬件的逻辑地址到物理地址的映射是通过（ ）实现的。

A. 内存管理单元

B. 内存地址寄存器

C. 重定位寄存器

D. 动态装载寄存器

内存管理单元。

9. 进程有哪三种基本状态，不同状态之间转换的原因是什么？

进程一共有三种基本状态：就绪态，运行态，阻塞态。当处于运行态的进程时间片耗尽或者被更高优先级的进程抢占 CPU，进程就从运行态进入就绪态；处于就绪状态的进程，当进程调度程序为之分配了处理机后，该进程便由就绪状态转变成执行状态；正在执行的进程因等待某种事件发生(比如 I/O)而无法继续执行时，便从执行状态变成阻塞状态；处于阻塞状态的进程，若其等待的事件已经发生，于是进程由阻塞状态转变为就绪状态。

1. 缓存（cache）和缓冲（buffer）都是为了提高数据交换的速度而设计的，缓存是在读取硬盘中的数据时，把最常用的数据保存在内存的缓存区中，再次读取该数据时，就不去硬盘中读取了，主要针对的时，先把数据放入缓冲区，然后再一次性将数据写入到硬盘中。这种方式多使用顺序的方式，用来缓解不是读操作，减少了对硬盘的频繁访问，从而提高了数据处理速度。

缓冲则主要用于解决速度差异的问题，它常见于对数据的暂存，然后批量传输或者写入。例如在向硬盘写入数据同设备之间频繁地、缓慢地随机写，从而减少对系统的冲击。

但缓存主要针对读操作，目的是复用已读数据以减少对硬盘的访问；

而缓冲则主要针对写操作，目的是缓解由于速度差异引起的系统冲击。

10. 什么是缓冲？缓冲的作用/好处有哪些？

缓存是用于保存两个设备之间或者设备和应用程序之间所传输数据的内存区域。缓冲可以帮助缓解处理数据流的生产者和消费者之间的速度差异(比如 CPU 和 I/O 设备);协调传输数据大小不一致的设备;支持应用程序 I/O 的复制语义。参考《操作系统概念》13.4.2 缓冲。

11. 分层操作系统和微内核操作系统的优缺点。

·**分层操作系统:**

优点: (1) 把功能实现的无序性改成有序性, 可显著提高设计的准确性。

(2) 把模块间的复杂依赖关系改为单向依赖关系, 即高层软件依赖于低层软件。

缺点:

对各层之间的详细定义比较困难。

·**微内核操作系统:**

优点:

有效地分离了内核与服务、服务与服务, 使得它们之间的接口更加清晰, 维护的代价大大降低; 各部分可以独立地优化和演进, 从而保证了操作系统的可靠性;

缺点:

频繁地在核作系统的执行开销偏大。

12. 简述死锁和饥饿的区别。

13. 死锁的四个必要条件及预防死锁的方案、死锁恢复的方案

互斥条件、占有且等待、不可剥夺条件、环路条件

破坏互斥: 共享设备

占有且等待: 一次性全部申请, 动态申请

不可剥夺: 抢占, 资源状况容易保存和恢复的情况、影响系统的吞吐量, 延长进程周期。

循环等待: 按照某种顺序申请, 降低资源的执行速度, 还会拒绝资源的访问。

·**恢复方案:**

1. 撤销所有进程: 简单易实现, 代价大

2. 恢复到前面定义的检查点, 重新运行这些进程。代价大, 可能再次发生死锁。

3. 有选择性地撤销死锁进程, 直至不存在死锁。最小代价原则, 优先级、运行时间, 每次撤销都要检查一次

4. 剥夺资源, 直至不存在死锁。最小代价原则, 死锁检测算法。

14. 内存分配方法:

1. 内存空间连续分配: 单一连续分配, 固定分区分配、动态分区分配(首次适应、最佳适应和最坏适应、邻近适应等)

2. 分页存储管理: 页号+页内偏移量 PTR 页表寄存器: 页表始址和长度 页号和页帧号

3. 分段存储管理: 段号+段内偏移量 段表寄存器: 段表始址和段表长度 段号+段长+始址

15. 比较页式存储管理和段式存储管理的异同点

相同点: 都是两次访存, 支持 TLB, 存在于内存中

不同点: 【大小、可见、碎片、空间、共享、链接】

管理思想：页是信息的物理单位，页的大小固定，对用户不可见，一维地址空间，内部碎片不超过一页，可以实现页面共享但控制困难；不支持动态链接

段是信息的逻辑单位，段的大小不固定，对用户可见，二维地址空间，外部碎片，便于共享逻辑完整的信息，易于实现存取访问权限控制；支持动态链接

16. 页面缓冲算法有哪些及优点。

·**空闲页面链表**：用于分配给频繁发生缺页的进程，以降低进程的缺页率。

·**修改页面链表**：减少已修改页面换出的次数，降低将页面写回磁盘的频率，减少磁盘 I/O 次数。

17. 什么是局部置换算法和全局置换算法，驻留集和工作集

·局部置换算法可以为每个进程分配固定的页帧，没有考虑各个进程之间的访存差异。

·全局置换算法在可运行进程之间动态分配页帧，分配给每个进程的页帧数是动态变化的。

·驻留集是进程在运行时，当前时刻实际驻留在内存中的页面集合。

·工作集是进程在过去一段时间访问的页面的集合。

18. 什么是系统抖动，产生的原因和解决方法。

·进程数增加，CPU 利用率增加---->过多，页帧数减少，频繁发生缺页、CPU 利用率下降--->继续增加--->页帧更少，缺页更频繁，CPU 利用率更低

·原因：分配的页帧数量太少、置换算法选择不当

·解决：

----工作集策略：监视工作集，分配大于工作集的帧数-->多余，另起进程-->超过，挂起，换出

----缺页率策略：设置缺页率上下限：超过，分配更多页帧‘低于，撤走多余页帧。

19. 操作系统为用户提供了哪些接口，它们是如何工作的？

操作系统为用户提供两个接口：命令接口和系统调用。

·命令接口：系统为用户提供的。利用**操作命令来组织和控制作业的执行或管理计算机系统**。

·系统调用：编程人员使用系统调用，是**内核函数的集合**。请求操作系统提供服务，例如申请和释放外设等类资源、控制程序的执行速度等。

20. 系统调用和 API 函数有什么区别和联系。

两者都可以被开发人员使用来访问**内核的底层功能**，而程序员更喜欢根据 API 来编程。

(1) 程序的可移植性。API 设计使程序能在任何支持同一 API 的系统上编译并执行，但是系统调用无法直接移植。

(2) 对程序员而言，实际系统调用比 API 更为注重细节且更加难用。

21. 指令的执行分为用户模式与内核模式，这两种模式有什么特点？【硬件访问、特权指令和崩溃恢复、效率】

·内核模式：对底层硬件的访问不受限制，可以执行特权指令和非特权指令，一旦崩溃会停止整个 CPU，效率高。

·用户模式：无法直接访问底层硬件和内存，须借助 API 访问，只能执行非特权指令，用户模式下的崩溃始终是可恢复的，效率低。

22. 比较用户级线程和内核级线程的优缺点.

1. 内核级线程:

- 优点: 【并行执行, 不受阻塞影响, 变态开销】

1) 同一进程的多个线程可以调度到多个处理器上并行执行, 若进程中的一个线程被阻塞, 可以调度进程中的多个线程运行。

2) 内核程序支持多线程, 提高操作系统内核程序执行效率.

- 缺点: 线程的切换需要内核的参与, 用户态-内核态-用户态过程需要开销.

2. 用户级线程:

- 优点:

1) 线程切换不需内核参与, 减少模式切换开销;

2) 线程调度算法和过程用户自行选择, 与 OS 内核无关;

3) 可以在任何 OS 上运行, 不需要对底层内核进行修改.

- 缺点: 一阻塞全阻塞; 多个线程无法并行执行.

23. 什么是进程互斥和同步? 解决进程互斥的方法有哪些?

- 进程互斥是进程之间的间接制约关系, 进程互斥就是保证每次只有一个进程使用临界资源。

- 进程同步是进程之间共同完成一项任务时直接发生的相互作用关系, 代表了进程之间的执行次序。

方法: 中断禁用、专用机器指令, 信号量机制, 管程等.

24. 为什么要引入虚拟文件系统?

- 隐藏了各种硬件的具体细节

- 把文件系统操作和不同文件系统的具体实现细节分离开来, 为所有的设备提供了统一的接口。

Ext2, NTFS, FAT

25. 进程创建的原因和过程?

26. 进程终止的原因和过程?

2. 有三个进程 P1,P2,P3 共享一个缓冲区, 缓冲区只能容纳一条消息, 进程 P1 向缓冲区存入消息, 进程 P2, P3 从缓冲区中取消息, 要求 P1 必须等待 P2 和 P3 都取过消息后才能存入下一条消息。用 P、V(或者 wait)、signal)操作描述三个进程存取消息的情况。要求写出完整的过程, 说明信号量的含义并赋初值。

46. (7分) 某虚拟存储器的用户空间有 32 个页面，每页 4KB，内存大小为 64KB，假设某时刻系统为用户进程依次装入第 0、2、3、4 页，分配的物理块号是 8、10、2、5。该用户进程的长度是 8 页，请将以下十六进制的虚拟地址转换成物理地址，并判断是否发生缺页中断，如果发生缺页中断，请按照 LRU 置换算法列出被置换的页面。

- (1) 3A5C
- (2) 7C2A
- (3) AB2C
- (4) FFFF

(1) 虚拟地址为 14940，由于每页为 4KB，所以虚拟地址所在页号为 3，已经在物理内存中，对应物理块号为 2，没有发生缺页中断。物理地址为：

$$14940 - 3 \times 4096 + 2 \times 4096 = 10844 \quad 2A5C_{16}$$

(2) 虚拟地址为 31786，虚拟地址所在页号为 7，不在物理内存中，发生缺页中断。根据 LRU 算法，此时 8 号物理块 (0 号虚拟页) 被换出。虚拟地址 31786 对应物理地址为：

$$31786 - 7 \times 4096 + 8 \times 4096 = 31786$$

(3)(4) 发生越界中断。

2015

46. (7分) 假设给进程 P 分配了 4 个页框，P 访问虚页的顺序为：A, B, C, D, E, C, A, B, C, D, F。

(1) 分别给出使用页面置换 FIFO (先进先出) 算法、LRU (最近最久未使用) 算法和 OPT (理想型淘汰) 算法的缺页数。

(2) 对于 LRU 算法，为使缺页数不超过 6 次，分配给进程 P 的最小页框数是多少？请说明理由。
注：初始时 内存中没有进程 P 的任何页面。

46. (8 分) 设有如下表所示的进程参数

进程	达到时间	服务时间
A	0	3
B	2	6

第 10 页 共 11 页

获取 考研经验/复试资料/考研资讯 关注微信公众号 计算机与软件考研

各个学校计算机/软件专业考研真题 免费分享 <https://github.com/csseky/cskaoyan>

C	4	4
D	6	5
E	8	2

忽略进程调度和系统的其它开销时间，请回答：

(1) 针对不同的调度算法在答题纸上绘制下表，并进行填写 (4.5 分)

进 程	FCFS (先来先服务)		最短进程优先 (SPN)		轮转法 (RR, 时间片=4)	
	开始时间	结束时间	开始时间	结束时间	开始时间	结束时间
A						
B						
C						
D						
E						

2016

46. (8 分) 假设进程 P0、P1 竞争使用临界区，且它们都要执行以下代码：

```

C1      while (flag == true)
C2          do nothing;
C3      flag = true;
C4      临界区;
C5      flag = false;
    
```

其中：flag 的初值为 false。请回答：

- 1) 上述代码能实现临界区互斥访问吗？请解释你的结论。(3 分)
- 2) 上述代码能实现临界区空闲让进吗？请解释你的结论。(2 分)
- 3) 上述代码能实现临界区有限等待吗？请解释你的结论。(3 分)

注：

(2) 可以实现让进。当没有进程在临界区(所有进程都已经离开临界区，flag 值此时肯定为 false)，其他进程可以立刻进入临界区。

(3) 不能实现有限等待。要实现有限等待，必须控制在一个进程请求临界区之后，其他进程允许进入临界区的上限次数，但是题目中代码并没有此类的限制。

2018.

47. (8 分) 请回答以下问题。

(1) 什么是死锁？

(2) 设有进程 P1、P2、P3、P4、P5 和三类资源 A、B、C。当前各进程资源占用和请求状况如表 3，且还有待分配资源 A 类 1 个、B 类 0 个、C 类 1 个。

表 3

进程	已分配资源			请求资源		
	A	B	C	A	B	C
P1	0	1	0	0	0	0
P2	2	0	0	2	0	4
P3	1	1	0	0	1	0
P4	2	1	0	1	0	1
P5	2	0	2	3	0	2

问：系统是安全的吗？为什么？

(3) 将表 3 中进程 P3 的请求资源修改为 (1, 2, 2)，问系统出现死锁了吗？如果未出现死锁，请说明理由；如果系统出现死锁，(a) 写出死锁进程集合，(b) 如何解除该死锁？理由是什么？

49、（9分）假设有页面引用序列：1, 2, 4, 3, 1, 2, 5, 1, 3, 2, 4, 5。请回答：

（1）如果给进程分配的页框（帧）数为3，分别计算 OPT，LRU 页面置换算法的缺页数。

（2）如果给进程分配的页框（帧）数为4，分别计算 OPT，LRU 页面置换算法的缺页数。