

1. 什么是多道程序设计？多道程序设计与分时系统的区别是什么？

多道程序设计技术，就是指允许多个程序同时进入内存并运行。即同时把多个程序放入内存中（前提是内存放的下），并允许它们交替在 CPU 中运行，它们共享系统中的各种硬、软件资源。当一道程序因 I/O 请求而暂停运行时，CPU 便立即转去运行另一道程序

分时是指多个用户分享使用同一台计算机。多个程序分时共享硬件和软件资源。这个技术的原理是把处理机的运行时间分成很短的时间片，并按时间片轮流把处理机分配给各联机作业使用。

他们的区别是，多道程序的目的是最大化处理器的使用率，而分时系统则是最小化响应时间。

2. 什么原因推动了操作系统从批处理发展到多道程序，进而发展到分时系统？

从批处理发展到多道程序，是因为 CPU 的运行速度远大于 IO 设备的运行速度，如果不发展为多道程序，则会极大增加 CPU 的空置率，增加运行总时间。

发展到分时系统，是为进一步提高 CPU 利用率，支持多用户、多进程。

3. 什么是陷阱？与中断的区别是什么？什么是系统调用？

陷阱：系统调用/软件中断

与中断的区别是陷阱是同步异常，是程序内部有意设置的，可以复现。

“系统调用”是操作系统提供给应用程序（程序员/编程人员）使用的接口，可以理解为一中可供应用程序调用的特殊函数，应用程序可以通过系统调用来请求获得操作系统内核的服务。

4. 判断：可移植的操作系统可以从一个系统架构移植到另外一个系统架构而无需修改。

a) 请解释为什么构建完全可移植的 OS 是不可能的？

不同的硬件有自己运行的不同的指令集，并且有可能架构不一样，比如冯诺依曼架构和哈佛架构，这些硬件上的差异导致没有办法建立一个完全可移植的操作系统

b) 如果需要你设计一个高度可移植的 OS，那么请描述你需要设计的两个层次？

第一个层次是直接和硬件交互的层次，这个层次会针对所有可能移植到的硬件作出支持，并且向上提供统一接口，让上层不比关注硬件的具体实现。

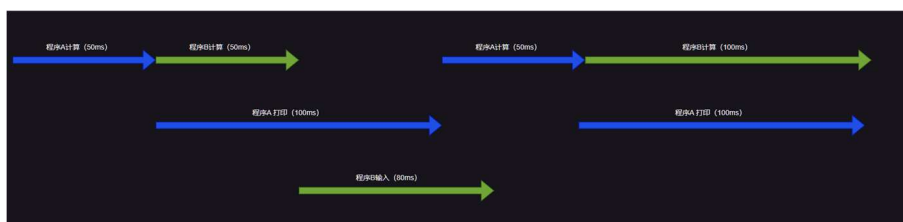
第二层则是和正常的操作系统一样的层次，负责提供操作系统接口给用户。

5. 在设计操作系统时，一些设计指标是相互矛盾的，例如资源利用率、吞吐量、处理时间、健壮性等。请给出一对相互矛盾的设计实例。

在内存管理策略上，为了高效利用内存资源，使用了分页系统，将物理内存划分成固定大小的页面，以便更有效地利用内存空间。通过分页，系统可以灵活地分配和回收内存，从而提高内存资源的利用率。然而，分页系统可能会增加响应时间，因为在访问页面时可能需要额外的时间来处理页表和地址转换。特别是在内存不足时，系统可能需要频繁地进行页面置换，导致额外的延迟。

6. 一个计算机系统有输入机一台、打印机两台，现有二道程序同时投入运行，且程序 A 先开始运行，程序 B 后运行。程序 A 的运行轨迹为：计算 50ms，打印信息 100ms，再计算 50ms，打印信息 100ms，结束。程序 B 运行的轨迹为：计算 50ms，输入数据 80ms，再计算 100ms，结束。要求：

a) 用图画出这二道程序并发执行时的工作情况。



b) 说明在二道程序运行时, CPU 有无空闲等待? 若有, 在哪段时间内等待? 为什么会空闲等待?

有, 在 100ms-150ms 之间, 因为 A 程序在输出, B 程序在输入, 均无法开始计算。

c) 程序 A、B 运行时有无等待现象? 在什么时候会发生等待现象

A 无等待。

B 在 180ms-200ms 之间等待, 因为 A 在计算, CPU 没有在空闲。