

# 操作系统作业一

## 1. 从系统的角度理解，操作系统主要负责哪两大功能？

- a) 操作系统是一个控制程序
  - i. 控制其它程序执行，防止错误与意外发生
- b) 操作系统是一个资源分配器
  - i. 管理系统资源
  - ii. 解决互斥的资源请求，提高效率，合理地利用系统资源

## 2. 什么是系统调用？阐述系统调用与 API 的区别和逻辑关系。

- a) 系统调用是由操作系统提供的进程与内核间的接口，用户程序通过系统调用来使操作系统为其完成一定的任务；是操作系统与计算机硬件的沟通方式
- b) API 是应用程序接口的缩写。
  - i. 两者的区别：API 是一系列函数的定义，与计算机内核无直接关系，可以由程序员在用户态下直接使用；而系统调用是通过中断向内核发请求，实现内核提供的某些服务。
  - ii. 两者的联系：API 中的某些函数会进行系统调用，以此访问计算机资源等。API 实现了对许多系统调用的封装，这样使程序员不必具体了解操作系统的实现过程，不必直接进行系统调用，有利于保证系统的稳定与安全

## 3. 阐述 Dual Mode 的工作机制，以及采用 Dual Mode 的原因。

- a) 工作机制：
  - i. 由硬件提供 Mode bit，用于系统区分用户态与内核态，以进行特权判断等
  - ii. 在开机时，硬件以内核模式启动
  - iii. 操作系统被加载至内存，并以用户模式开启用户应用程序

- iv. 当中断发生时，硬件从用户态切换至内核模式
- v. 当操作系统获得控制权时，切换至内核模式
- vi. 操作系统服务(系统调用)完成后，返回至用户模式

b) 采用 Dual Mode 的原因：

便于操作系统自我保护，以及保证系统其它组件的安全。

例如：操作系统中可能存在清理内存、中断时钟等“危险指令”，设置内核态及特权模式，将其设置为特权指令，可以防止用户随意使用造成不良后果。

#### 4. 分析 Monolithic 结构，层次化结构，模块化结构和微内核结构的优劣。

a) Monolithic 结构：将所有功能服务合并至一级

- i. 优点：系统调用接口和内核通信的开销较小
- ii. 缺点：内核层包含功能过多，设计上较难实现，且难以维护

b) 层次化结构：每层只能调用更低层次的功能操作和服务

- i. 优点：简化了构造与调试（从低层到高层逐层调试）
- ii. 缺点：合理定义各层划分存在一定困难；各层间交互时效率可能较差，每层都为系统调用增加额外开销

c) 模块化结构：内核提供核心服务，其它服务在内核运行时动态实现

- i. 优点：任何模块都可以调用任何其他模型；较为灵活；无需消息传递，更高效
- ii. 缺点：设计时对模块间的接口规定难以满足多变的实际需求

d) 微内核结构：从内核中删去不必要的部件，将这些部件作为系统级或用户级程序实现

- i. 优点：便于扩展操作系统，新增服务可以在用户空间增加，而不需要对内核进行修改；大多数服务作为用户进程来运行，安全性、可靠性更高
- ii. 缺点：用户空间与内核空间之间通信较为频繁，造成性能开销较大

## 5. 举例说明采用机制与策略分离的设计原因。

- a) 例一：操作系统提供了赋予程序更高优先级的机制。如过该机制与使用时策略不能分开，则该只能赋予某些特定程序以优先权，灵活性差。而采用了机制与策略分离的设计后，灵活性更高：比如为了提高人机交互性，可以给 I/O 密集型程序优先权；若有其它特殊需求，则也可以赋予 CPU 密集型程序更高的优先级
- b) 例二：UNIX 系统提供了分时、实时、批处理、公平分享等多种调度机制，用户可以通过加载表来控制，实现不同的调度方案  
分析：由用户选择调度方案（即策略），更好地应对不同应用场景