

### 1. 请分别从系统和用户的角度，阐述操作系统的主要功能？

答：从用户角度，大部分PC机操作系统的主要功能是优化用户进行的工作，主要的设计目标是使得用户使用方便；对于同大型机或小型机相连的终端，操作系统的功能是优化资源利用率，确保用户没有使用超额以外的资源；对于工作站系统，还需要兼顾方便性和资源利用率。

从系统角度，操作系统是资源分配器，负责管理如CPU时间、内存空间、文件存储空间、IO设备等资源，操作系统需要考虑为各个程序和用户合理地分配这些资源。另外一个视角是，操作系统是一个控制程序，负责控制各种IO设备和用户程序等需求，防止资源等错误或者不当使用。

### 2. 解释什么是系统调用？阐述系统调用与 API 的逻辑关系。

答：系统调用提供操作系统服务接口。根据书上所表述的，我个人理解是系统调用相对于API是更底层也是更直接的服务程序，它由内核函数实现（有些可能直接对硬件进行操作），系统调用函数是进入内核态的接口。API是在用户态提供的接口函数（有点类似于在用户态对系统调用加了一层封装），但是，调用API完成功能时，一个API可能会进行系统调用，也可能不会，可能使用多个/多次系统调用（这取决于API所要实现的功能），程序员在使用系统提供的API时，无需关注底层的细节（系统调用的参数往往更加复杂），而是根据操作系统规定好的API实现相应功能来进行调用。

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

答：计算机硬件可以通过一个模式位来区分内核模式和用户模式，用于区分操作系统执行的任务和用户执行的任务。当计算机执行用户应用时，系统处于用户模式，而当用户应用通过系统调用，请求操作系统服务时，系统将切换到内核模式执行系统调用。采用双重模式的主要目的是为计算机系统提供保护，防止用户程序的错误影响到操作系统或底层硬件。

### 4. 概述操作系统需要提供的服务有哪些？

答：操作系统提供的服务根据计算机组成/体系结构或者是设计需求的不同而可能有所不同。根据书中第二章所述，操作系统提供的常见服务有：

- 用户界面：包括命令行界面，批处理界面，图形用户界面。
- 程序执行：加载程序到内存。
- I/O操作

- 文件系统操作
- 进程间通信
- 错误检测
- 资源分配：如CPU调度、内存管理、文件、I/O资源管理
- 记账：记录用户使用的资源类型和数量。
- 保护与安全

5. 分别阐述 Monolithic 结构，层次化结构，模块化结构和微内核结构的特点和优劣。

答：

- Monolithic 结构：结构简单、占用空间小，系统调用和内核通信开销较小；但功能有限，没有很好地区分功能的接口和层次，导致操作系统易出错，易受攻击。
- 层次化结构：层次化的操作系统，易于开发，简化了构造和调试。缺点是难以合理定义各层，系统调用开销较大。
- 微内核：将内核中不必要的部件移除，作为系统级和用户级的程序实现，内核较小，仅保留最基础的功能，用于为用户空间的各种程序和服务提供通信。这种微内核结构优点是便于扩展操作系统，修改和移植的代价小。但带来的缺点是这种结构的性能会受损。
- 模块化结构：目前来看最佳的操作系统设计结构，其思想是内核提供核心服务，其他服务在内核运行时动态实现，无需重新编译内核即可增加功能。这种结构吸收了层次化结构和微内核结构的优点。缺点：需要保证设计使操作系统各模块间耦合度低（就是说不同的模块间应该有足够的独立性），且无法找到一个可靠的模块开发顺序（模块开发齐头并进，没有先后顺序，很难做到开发的每一步决定都是建立在可靠的基础上）。增加新的内容时可能产生系统碎片化（模块间功能的冗余和冲突、新旧版本模块的更迭和兼容）和安全问题（新模块的漏洞不一定可以及时修复）。

6. 举例说明什么是机制与策略分离的设计原则。

答：机制决定如何做，而策略决定做什么。策略与机制分离对于灵活性来说很重要。策略可能会随着位置或时间的改变而改变。在最坏的情况下，策略的每一次改变都需要底层机制的改变。比较好的情况是有一个足够灵活的通用机制，它能够在一系列策略中发挥作用，策略的改变仅需要重新定义系统的某些参数。无论何时，只要决定是否分配资源，就必须作出策略决定。无论何时，只要问题是“如何做”而不是“是什么”，这就必须要由机制决定。

举例说明：比如机制是调度算法，策略是拿实际调度中各种调度算法的混合使用和优先级分配，再比如机制是数学公式，策略是拿数学公式解题。虽然答案很合理但并没有举例。

7. 概述 **multi-programming** 和 **multi-tasking** 的概念及其设计目的。

答：多道程序设计是通过安排作业（编码和数据）使得CPU总有一个执行作业，从而提高CPU利用率。其方法是将多个作业保存在作业池上，作业选择和作业调度同时进行，当某个作业需要等待时，CPU就切换到另一个作业。

多任务是多道程序设计的延伸，对于多任务系统，CPU还是通过切换作业的方式执行多个作业，但切换频率很高，因此用户可以在程序运行时与其交互。分时系统允许多个用户同时共享一台计算机。