

1. 现代计算机需要层级的原因：

- ① 模块化和分工：层级结构使得计算机系统可以被模块化地设计和实现。
- ② 抽象和封装：每个层次都提供了对底层实现的抽象，隐藏了底层的细节，使得上层可以更好地使用和操作系统。
- ③ 简化复杂性：将系统划分为不同层次，可以将复杂性分解为更小的、易于管理的模块，降低了系统设计和开发的复杂性。
- ④ 提高可移植性：层次结构使得不同的层次可以独立设计和实现，因此可轻易在不同硬件和操作系统上移植和运行。

2. 过大：

- ① 内部碎片：当页的大小超过需要的内存大小时，每个进程可能会浪费一些空间，导致内部碎片。这意味着在每个页中可能有一些未被使用的空间。
- ② 页面调度开销：如果页的大小过大，系统需要更多的时间和资源来将页调入内存或从内存中调出，这增加了页面的开销，可能会降低系统的性能。

过小：

- ① 外部碎片：当页的大小太小，无法容纳一个完整的数据块或程序时，会导致外部碎片，无法得到有效利用。
- ② 页面表开销：需要更多的页面表项来管理相同数量的内存，增加访问开销。

3. 1) 位 0：存住位 (V) 表示对应的虚拟页是否在物理内存中存在。

位 1：可读位 (R) 表示是否可读。

位 2：可写位 (W) 表示是否可写。

位 3：执行位 (X) 表示是否有执行权限。

位 4：用户位 (U) 表示是否可由用户模式访问。

位 5：全局位 (G) 表示是否是全局页。

位 6：访问位 (A) 表示是否被访问过。

位 7：脏位 (D) 表示是否已被修改。

- 2) ① 安全性问题. ② 内存分配、释放和管理混乱.
③ 冲突和混乱. ④ 破坏系统稳定性, 性能下降.
3) 挂上执行、写入和读取, 通常用于实现一些特殊内存保护策略.

4) ① PMP 控制寄存器用于物理内存保护, 提供对物理内存区域的访问权限.

② PMP 中 R/W/X 位用于指定字段的物理内存区域的访问权限.

③ 作用是控制硬件线程对物理内存的直接访问, 而不涉及虚拟地址和页表.

2). L 位: 用于指示对应的 PMP 寄存器是否被锁定,

A 位: 用于指示对应的 PMP 寄存器是否用于物理内存地址的匹配.

5. 1). $2^4 \times 2^3 / 2^{12} = 2^{5.5} B = 2^{15} TB = 32 PB.$

2). $2^{48} \times 2^3 / 2^{12} = 2^{39} B = 512 GB$

3). ① 分层结构:

② 节省内存空间.

③ 弹性分配: