

1. 答：随着现代处理器性能的不断攀升，处理器和内存之间的速度差距不断扩大，形成“内存墙”的问题。为此，现代计算机使用多级分层存储结构，并引入缓存系统，利用数据的时间局部性和空间局部性来改善性能。

2. 答：(1) 过大的页会引起的问题：

- ① 造成内存浪费。过大的页中只有一小部分被使用，其余则被浪费。
- ② 页表过大。页的大小越大，虚拟地址与物理地址的映射关系就越多。

(2) 过小的页会引起的问题：

- ① 可能与进程所需空间不匹配。若进程所需空间大于页的大小，则无法满足。
- ② 页的数目过多。系统需要管理的页的数目过多，降低系统性能。

3. 解：(1) ① D 位：当虚拟地址被写时，对应的 PTE 的 D 位被置位；

② A 获取位：虚拟地址被读写或匹配时，对应的 PTE 的 A 位被置位；

③ G 位为全局映射，代表后续级别的所有映射表是全局性的；

④ U 位表明该页表是否可由 U 态使用。为 1 时可由 U 态使用，同时若 sstatus 寄存器中的 PUM 位清零，则 S 态软件也可以在 U 位为 1 时获取页；

⑤ R/W/X 为权限位，表明该页是否可读、写、执行；当三者均为 0 时，PTE 是一个指向子页表的指针；否则，是一个叶页表项；

⑥ V 位表明该 PTE 是否合法。若为 0，则 PTE 的 31~1 bit 位不关心并且可以由软件自由使用。

(2) 如果用户进程能够自由修改自己的页表，可能会改变页表的权限，改变后续映射表的作用范围，同时可能无法确定虚拟地址的状态。

(3) X/W/R 全为 0 时，该有效页表条目是指向子页表的指针。

4. 答：(1) PMP 控制寄存器中的 X/N 位控制的是相应物理内存区域的权限

(2) ① L 字段表示 PMP entry 处于锁定状态，此时对于配置寄存器和相应的物理寄存器的写入会被忽略。

当 L 字段为 1 时，M、S、U 模式都必须遵循配置寄存器的权限设置。

当 L 字段为 0 时，S、U 仍需遵循上述规定，而 M 模式不需要。

② A 字段与地址寄存器共同决定该 PMP entry 控制的物理地址的范围

5. 解：(1) 使用 64 位虚拟地址时，寻址空间为 2^{64} 字节

每个页表条目使用 8 字节空间，页大小为 4 KB

因此所需的页表条目数为 $\frac{2^{64}B}{2^{12}B} = 2^{52}$ 个

所需空间为 $2^3B \times 2^{52} = 2^{55}B = 32EB$

(2) 使用 48 位虚拟地址时，寻址空间为 2^{48} B

所需页表条目数为 $\frac{2^{48}B}{2^{12}B} = 2^{36}$ 个

所需空间为 $2^3B \times 2^{36} = 2^{39}B = 512GB$

(3) 多级页表能随着进程占用空间的增大，对应地增加该进程的页表数目，而当进程占用内存空间很小时，页表数目也很少，以此减少页表占用的内存空间