

7. 为什么要引入：程序在运行过程中经常要在内存中移动位置，为了保证这些被移动了的程序还能正常执行，必须对数据的地址加以修改，即重定位。如何实现：增设一个重定位寄存器，用它来存放程序在内存中的起始地址。程序执行时，真正访问的内存地址是相对地址与重定位寄存器中的地址相加而形成的。

12. 固定分区存储管理：1>优点：实现简单，无需复杂的分配和回收算法。2>缺点：由于分区大小是事先固定的，因而可容纳作业的大小受到限制。当用户作业的地址空间大于分区的存储空间时，会造成存储空间浪费。

动态分区存储管理：1>优点：内存利用率高，不会产生固定分区管理因分区过大而造成的内存浪费。2>缺点：系统开销较大，因为每次分配和回收内存都需要进行分区的划分和合并操作，且实现相对复杂。

18. 页面：分页存储管理将进程的逻辑地址空间分成若干大小相等的片。物理块：把内存空间分成与页面相同大小的若干个存储块。页面大小一般由操作系统决定，其确定方法基于系统设计、硬件架构、内存利用效率等因素。

19. 页表：是分页式存储管理使用的数据结构，一个进程分为多少页，它的页表就有多少行，每一行记录进程的一页和

它有效的物理块的页号、块号对应关系。
页表的作用是进行地址交换。

22. 首先, 以前转换过程中查找过的页表项会被记录在快表之中。在每次需要访问逻辑地址中的数据的时候, 都会根据页号去到快表中查看有没有对应的匹配项, 如果有就直接得到相应的物理块号而不需要进过利用始址、页号和页表项长度计算出主内存中查表, 得到物理块号后依然是送到物理寄存器中与页内地址结合形成最后的物理地址。

23. 共同点: 都采用离散分配的方式, 且都要通过地址映射机构来实现地址交换。不同点: ①页是信息的物理单位而段是信息的逻辑单位 ②页的大小固定且由系统决定, 段的大小不固定 ③分页用户程序地址空间是一维的, 分段用户程序地址空间是二维的。