

Q1. What is Uniprogramming:

What is the shortcoming of Uniprogramming:

Uniprogramming 指的是内存中同一时间只存在一个程序，也只运行这个程序，该程序在这一时刻占有所有的资源，包括内存与 CPU。

缺点是资源利用的效率非常低，比如一直执行这个程序，CPU 必须等 I/O 结束才能继续工作，极大的降低了 CPU 的使用效率。同时切换程序的代价十分大，也无法真正实现程序的同时运行。

Q2. What is Multiprogramming:

What is the shortcoming of Multiprogramming:

Multiprogramming 指的是内存中可以同时存在多个程序。一个 CPU 处理器同一时刻只能运行一个进程，但操作系统可以通过 context switching 来处理多个程序，从而在 wait I/O 的时候不占用 CPU 资源，此时可以根据一定的算法来选择下一个执行的进程。

缺点是内存分配变得比较复杂，会产生内碎片、外碎片等问题。

Q3. What is the segmentation mechanism and its advantages & disadvantages:

分段机制通过 segmentation table 将多个分离的不定长度的内存段（代码，数据，系统还有一些共享的内存段等）从虚拟地址空间映射到物理地址空间，且保证每个内存段对应的物理地址是连续的，有对应的 base 和 limit 的值（通过 $\text{base} + \text{offset}$ 定位到物理地址），理论上可以对应到物理内存空间的任何位置。

优点：

1. 数据的组织更加有逻辑，利于共享 share
2. 可以更好的保护数据，比如给某一特定的内存段设置权限，也能够将不同进程的地址空间隔离开来
3. 当进程的内存段足够小时，可以一定程度降低碎片

缺点：

1. 分段长度不确定，分配机制更加复杂
2. 一个进程需要多次分配、转换内存地址
3. 仍然存在碎片的问题

Q4. What is the paging mechanism and its advantages & disadvantages:

分页将物理内存分为大小一致的块，进程也被分为同样大小的页。将这个进程从虚拟地址空间映射到物理地址空间实际上就是将该进程的页从虚拟地址空间映射到物理地址空间，一个页的地址是连续分布的，不同页的物理地址不要求连续。映射规则由 page table 维护。

优点：

1. 内存分配更加容易实现，易于 share
2. 消除外碎片，可以更高效的利用内存

缺点：

1. Page table 需要占用额外的内存空间
2. 仍然存在内碎片的问题

Q5. How to reduce the occurrence of internal fragmentation when allocating memory? Briefly explain why this method works.

1. 动态分配内存或可变分区分配。通过动态的分配可变长度的内存，使得分配的内存正好适合于这一段，可以消除内碎片（但会产生外碎片）。

2. 使用 **Best Fit** 算法。通过 **Best Fit** 算法，每次分配的都是最合适的内存段，从而使得分配产生的内碎片尽可能的小，可以一定程度减少内碎片。

Q6. How to reduce the occurrence of external fragmentation when allocating memory? Briefly explain why this method works.

1. 通过分页或 固定分区分配。在分页机制下，每一页的大小都是固定的，物理地址空间的每一页都可以对应一个虚拟地址空间的页，因此可以消除外碎片。其他固定分区分配的思路与此类似。

2. 使用 **Worst Fit** 算法。通过 **Worst Fit** 算法，每次分配的都是最大的内存段，从而使得分配产生的“碎片”也尽可能的大。而大“碎片”可以分配给比它小的段，因此不成为真正意义上的碎片，可以一定程度减少外碎片。

3. 采取合适的内存回收机制。如将地址连续的内存块合并成为一个更大内存块。