**OS简答题做法示例**

简答题为OS的一种常见题型，其解答不能太综合、简单，否则失分严重。一般的做法为：逐条列出答案要点；对列出的每条答案要点进行较为详细的解释、阐述。下面给出一个示例，供参考。

**例. 简述有哪些磁盘调度算法。**

**答：** 当有多个进程都请求访问磁盘时，应使各进程对磁盘的平均访问时间（主要是寻道） 最小。因此，磁盘调度的目标应是使磁盘的平均寻道时间最少。目前常用的磁盘调度算法有：

**（1）先来先服务**

根据进程请求访问磁盘的先后次序进行调度。优点：公平、简单，且每个进程的请求都能依次得到处理；存在的问题：未对寻道进行优化，致使平均寻道时间可能较长。

**（2）最短寻道时间优先（SSTF）**

算法选择要求访问的磁道与当前磁头所在的磁道距离最近的进程，以使每次的寻道时问最短。存在的问题：可能导致某些进程发生“饥饿”。这种调度算法不能保证平均寻道时间最短。

**（3）扫描（SCAN）算法（又称为电梯调度算法）**

SCAN算法不仅考虑欲访问的磁道与当前磁道的距离，更优先考虑磁头的当前移动方向。在磁头正在自里向外移动时，所选择的下一个访问对象应是其欲访问的磁道既在当前磁道之外，又是距离最近的。这样自里向外地访问，直至再无更外的磁道需要访问时，才将磁臂换向，自外向里移动。这时，每次选择要访问的磁道，在当前磁道之内且距离最近者这样的进程来调度。

算法既能获得较好的寻道性能，又能防止进程饥饿，被广泛用于大、中、小型机和网络中的磁盘调度。

存在的问题：当磁头刚从里向外移动过某一磁道时，恰有一进程请求访问此磁道，这时该进程必须等待，待磁头从里向外，然后再从外向里扫描完所有要访问的磁道后，才处理该进程的请求，致使该进程的请求被严重地推迟。

**（4）循环扫描CSCAN**

为了减少请求进程的延迟，CSCAN算法规定磁头单向移动。若规定只自里向外移动，当磁头移到最外的被访问磁道时，磁头立即返回到最里的欲访磁道，即将最小磁道号紧接着最大磁道号构成循环，进行扫描。

**（5）N-Step-SCAN算法**

SSTF、SCAN、CSCAN几种调度算法都可能出现磁臂停留在某处不动的情况，称为磁臂粘着。在高密度盘上更容易出现此情况。N-Step-SCAN算法将磁盘请求队列分成若干个长度为N的子队列。磁盘调度将按FCFS算法依次处理这些子队列，而每处理一个队列时，又是按SCAN算法。这样就可避免出现粘着现象。N值取得很大时，其性能接近SCAN算法；N=1时，则退化为FCFS算法。

（6）FSCAN算法：本算法是N-Step-SCAN算法的简化。它只将磁盘请求访问队列分成两个子队列。算法思想是，在扫描的过程中所有新产生的序列放在另外的一个队列中，当访问完当前队列之后，再访问新产生的一个队列。这种算法可以有效防止磁壁粘着现象。