设备管理例题 2

同济大学计算机系操作系统课程作业　　邓蓉

一、一个磁盘组有 100 个柱面，每个柱面有 8 个磁道（磁道号=磁头号），每根磁道 8 个扇区，每个扇区 512 字节。现有含 6400 个记录的文件，每条记录 512 字节。文件从 0 柱面、0 磁道、0 扇区顺序存放。

1、 若同根磁道，相邻磁盘数据块存放在相邻物理扇区（现代硬盘，磁盘数据缓存的尺寸：整根磁道）

（1）3680#记录存放的位置。 柱面号=？， 磁道号=？， 扇区号=？

（2）78#柱面，6#磁道，6#扇区存放该文件的第几个记录？

【参考答案】

柱面，从最外圈向内递增编号。从 0 开始，先编号 n 号柱面的所有磁道，后编号 n+1 号柱面的所有磁道。属于同一柱面的多个磁道，按磁头号递增对磁道依次编号。一根磁道，所有扇区递增编号，之后编号下一个扇区。

（1） 每个柱面 8*8 = 64 个扇区。

3680/64 = 57　　柱面号是 57。

3680 % 64 = 32　　这个记录存放在 57#柱面的 32#扇区。

每根磁道 8 个扇区。

32 / 8 = 4　　32#扇区在 4#磁道（4#读写头管理的那个磁道）

32 % 8 = 0　　是 4#磁道的 0#扇区。

3680#记录存放的位置是：　57#柱面，4#磁道，0#扇区。

（2）78*64+6*8+6 = 5046。

2、 若同根磁道，相邻磁盘数据块错开一个物理扇区（老式硬盘，磁盘数据缓存的尺寸：一个扇区）

（1）3680#记录存放的位置。 柱面号=？， 磁道号=？， 扇区号=？

（2）78#柱面，6#磁道，6#扇区存放该文件的第几个记录？

【参考答案】

（1）3680#记录存放的位置是：　57#柱面，4#磁道，0#扇区。

（2）78*64+6*8+3 = 5043

二、假定磁盘的移动臂现在正处在第 8 柱面，有如下 6 个请求者等待访问磁盘。假设寻道时间>>旋转延迟。请列出最省时间的响应次序：

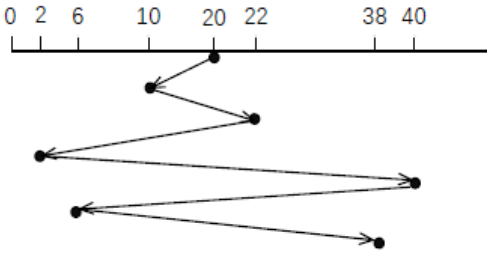
序号	柱面号	磁头号	扇区号
（1）	9	6	3
（2）	7	5	6
（3）	15	20	6
（4）	9	4	4
（5）	20	9	5
（6）	7	15	2

【参考答案】优先考虑缩短寻道时间。优化后的柱面访问次序：7，9，15，20
所以最省时间的相应次序应该是：（2，6），（1，4），3，5。括号内请求，次序可以互换。

三、当前磁盘读写位于柱面号 20，此时有多个磁盘请求以下列柱面号顺序送至磁盘驱动器：10，22，2，40，6，38。寻道时，移动一个柱面需要 6ms。

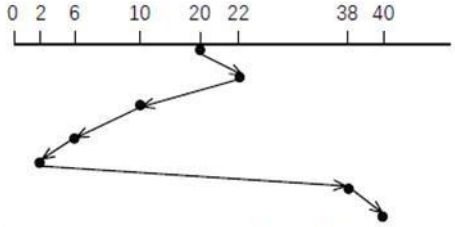
1、按下列 3 种算法计算所需寻道时间，画磁头移动轨迹。

（1） 先来先服务



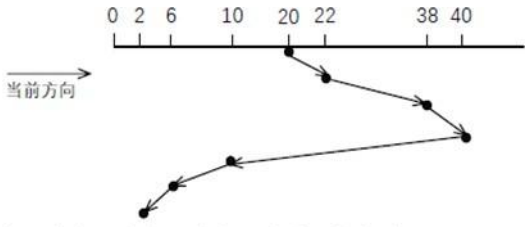
移动磁道数 =(20-10)+(22-10)+(22-2)+(40-2)+(40-6)+(38-6)=146
寻道时间 =146×6 = 876ms

（2） SSTF（下一个最邻近柱面）



移动磁道数 =(22-20)+(22-10)+(10-6)+(6-2)+(38-2)+(40-38)=60
寻道时间 =60×6 = 360ms

（3） 电梯调度算法（当前状态为向上）



移动磁道数 =(22-20)+(38-22)+(40-38)+(40-10)+(10-6)+(6-2)=58
寻道时间 =58×6 = 348ms

2、为什么电梯调度算法性能优于另两种算法?

答：观察上述磁臂移动轨迹，可以发现，电梯调度算法磁臂折返距离最短。

四、外设的独占和共享

1、 从硬件驱动的角度，所有外设必须独占使用。为什么?

【参考答案】外设芯片中的数据缓存，命令、状态、数据寄存器只能存放一个请求的 IO 参数和 IO 数据。另外，对外设而言（以读操作为例），一次典型的 IO 操作包括：CPU 发 IO命令，外设执行 IO 命令，数据寄存器中的新数据读入内存。全部完成后，CPU 才可以向 外设发下一个 IO 命令。综上，从硬件驱动的角度，所有外设必须独占使用，没有例外。

2、多道系统，硬盘是共享设备。并发执行的多个任务可以同时访问硬盘。
内核引入了（ 磁盘高速缓存，IO 请求队列 ）填内核数据结构 将必须独占使用的物理硬盘改造成逻辑上的共享设备。

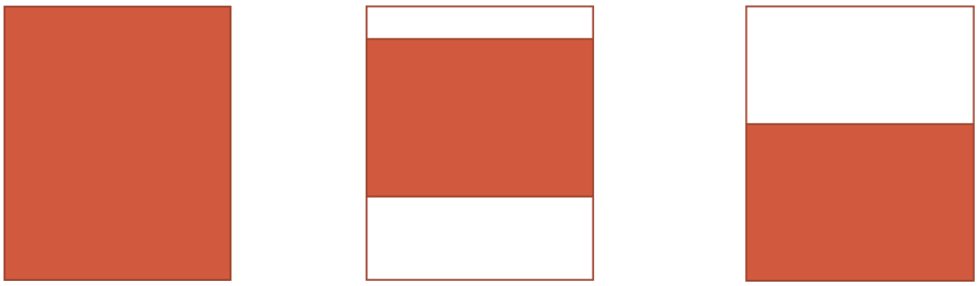
3、打印机是必需独占使用的外设。并发 2 个打印任务，两个输出内容交织在一起，打印结果是不可用的。Spooling 技术借助硬盘这个共享设备，将原先必须独占使用的打印机改造成能够同时为多个用户提供打印服务的共享设备。思路是：

- 系统维护一个 FIFS 的打印队列。
- 进程需要打印时，
  - 新建一个临时磁盘文件，命名之。把要打印的内容写入这个文件。
  - 生成一个打印作业控制块，包含进程 ID，用户 ID，临时文件名……，送打印队列尾。
  - 进程返回。无需等待打印 IO 完成。
- 打印机完成当前打印任务后，取打印队列队首打印作业控制块，构造针对打印机的新的 IO 命令。

PS1：相对打印机，磁盘是很快的设备。所以，Spooling 技术同时也改善了打印机这个 IO 子系统的响应速度。是个很不错的 IO 优化技术呐。对照期末自己在学子打印资料看到的现象，体会下Spooling 技术。
PS2：教科书上 Spooling 技术相关的，有输入井和输出井这两个概念。打印机的输出井就是一个文件夹，用来存放临时文件（上面高亮的那个）。

五、不借助缓存写磁盘，需要先读后写吗？
答：需要。这是因为扇区是磁盘读写的原子单位。写命令会更新整个扇区（512 字节）。所以，当写入的字符量不足 512 字节时，必须先读以保护不会写操作改写的区域。

六、针对下图中的所有情况，写数据块进程是否会入睡，需要执行几次 IO 操作？假设系统缓存队列非空，只有一个进程访问磁盘上的文件系统。



答：

（1） 缓存块不命中。需要为数据块分配缓存块，自由缓存队列非空，进程不会睡。
- 子图 (a)，无需先读，不延迟写。一次异步写IO，进程不睡。
- 子图 (b)，需要先读，延迟写。一次同步读IO。进程会睡，等待读操作结束。
- 子图 (c)，需要先读，不延迟写。两次 IO 操作，一次同步读，一次异步写。进程等待读操作结束时会睡。

（2） 缓存块命中。不需要为数据块分配缓存块。
- 子图 (a)，无需先读，不延迟写。一次异步写IO。进程不睡。
- 子图 (b)，需要先读，延迟写。没有 IO 操作。进程不睡。

- 子图 (c)，需要先读，不延迟写。一次异步写 io。进程不睡。

  注：这里的先读指的是执行 Bread( )函数，缓存命中时不需要执行磁盘读操作。

  延迟写，当前不执行写操作。 不延迟写，立即执行写操作。

七、仅考虑 556#数据块。请问，完成序列中全部的写操作，需要执行几次磁盘 IO 操作？总耗时多少？平均耗时？ 读写序列：

556（写 40#字节），556（写 1#字节），556（写 0~127#字节），600，782，891，900，556（写 128~255#字节），556（写 256~383#字节），556（写 384~511#字节）。

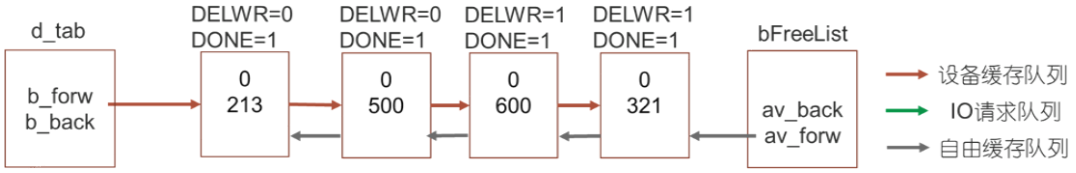
不使用磁盘高速缓存。已知发出 IO 请求至 IO 操作完成，进程平均需要等待 T。

【参考答案】每次写566#扇区均先读后写。6*2=12 次IO。总耗时12*T+6t；平均耗时2*T+t。

（1） 使用高速缓存。已知自由缓存队列中有足够多的干净缓存。将IO 请求放入 IO 请求队列后至 IO 操作完成，进程平均需要等待 T。从缓存复制数据到用户空间（含上锁、解锁操作），平均耗时 t。

【参考答案】使用缓存，2 次 IO。第一次写 556#块的时候，先读会 IO，最后一次写 556#块的时候，写满了，一次异步写 IO。总耗时 2*T+6t；平均耗时 T/3+t。

八、看图，回答问题



（1） 图中为什么没有 IO 请求队列?

答：所有缓存块B_DONE 是 1，IO 已完成。所以，IO 请求队列是空的。

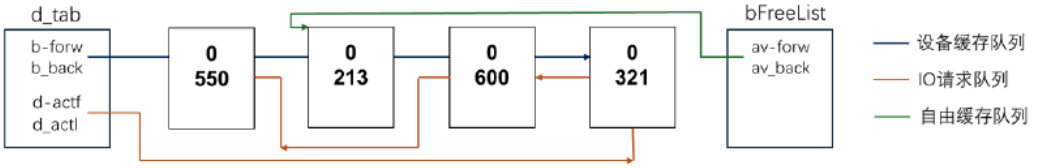
（2） 读 213，500，600，321 块。不管是否脏缓存（是否有延迟写标记），进程不会睡，没 IO。

（3） 写 213，500，600，321 块。不管是否脏缓存（是否有延迟写标记），进程不会睡。缓存写满了，异步写回磁盘，一次 IO。没写满，没有 IO。

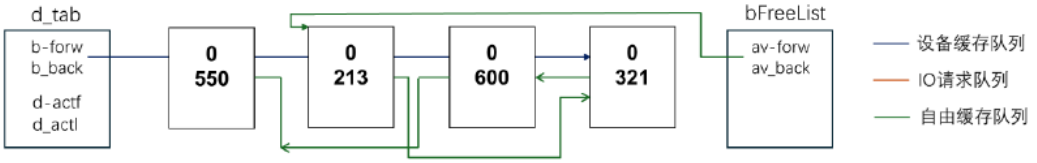
2、3，注意自由缓存队列的变化。GetBlk，锁住某个缓存块之后，会把它从自由缓存队列里抽出来。Brelse 释放缓存块的时候，送自由缓存队列末尾。

（4）写550#块，偏移量为8的字节。

答：缓存不命中。分配自由缓存队列第一个不脏的Buffer（不带延迟写标记B_DELWR），将沿途所有脏缓存写回磁盘。321，600号数据块异步写回磁盘。为550#数据块分配500#数据块原先占据的缓存块。先读后写，一次同步读 IO。



3个IO请求块送入IO队列。设备缓存队列队首是最后发出的IO请求。脏缓存块在队列中位置不变（考试不要求）。



550#块写操作完毕解锁