1. I/O控制有哪几种？简述他们区别。

共有四种，分别是程序控制I/O、中断驱动、DMA和通道。

程序控制I/O，也称轮询或查询方式I/O，它由CPU代表进程向I/O模块发出指令，然后进入忙等状态，直到操作完成之后进程才能够继续执行。这种方式由于存在忙等，浪费CPU资源，通常效率较低。

中断驱动：当I/O操作结束后，由设备控制器主动通知设备驱动程序，而不是由设备驱动程序不断地去轮询查看设备的状态。这种方式通常而言效率较高，但在网络传输等这种IO迅速的场合，也可能因为不断中断CPU的工作而变得效率低下。

DMA是指直接存储器访问方式，是由一个专门的控制器来完成数据从内存到设备或者是从设备到内存的传输工作。这种方式分隔了IO设备和CPU的工作，使得CPU可以全速前进，效率较高。

通道与DMA的原理几乎是一样的，通道是一个特殊功能的处理器，它有自己的指令和程序专门负责数据输入输出的传输控制。CPU将“传输控制”的功能下放给通道后，只负责“数据处理”功能。这样，通道与CPU分时使用内存，实现了CPU内部运算与I/O设备的并行工作。一般而言，通道更适用于大型设备，在PC机上则较少见到。

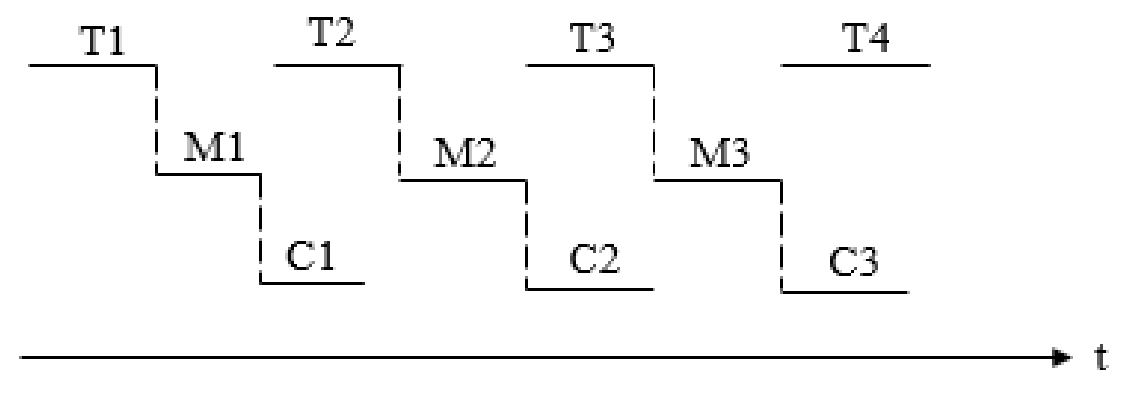
1. 何为设备独立性？如何实现设备独立性？

设备独立性是指操作系统能将所有IO设备都看作是一个文件，用文件的方式来操控它。具体到IO设备内容的操作，由设备驱动程序进一步完成，操作系统只需要向设备驱动程序发出统一的控制信号即可。为了实现设备独立性而引入了逻辑设备和物理设备这两个概念。在应用程序中，使用逻辑设备名称来请求使用某类设备，而系统在实际执行时，还必须使用物理设备名称来操控设备的具体内容。

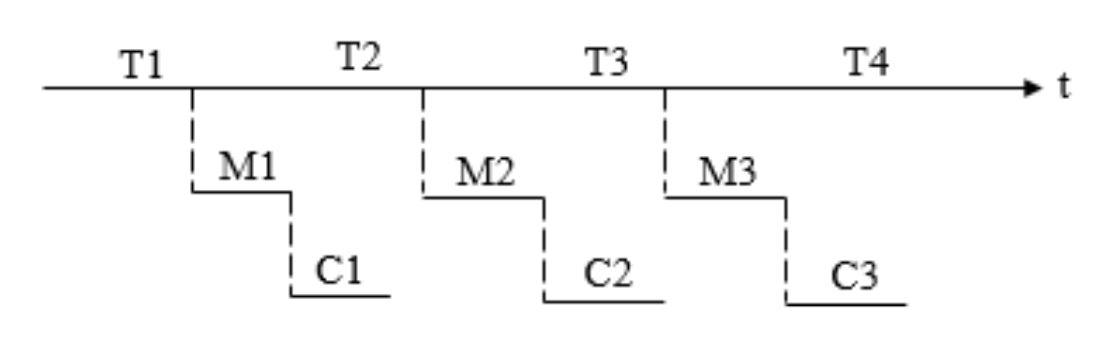
1. 在I/O系统中引入缓冲的主要原因是什么？某文件占10个磁盘块，现要把该文件的磁盘块逐个读入主存缓冲区，并送用户区进行分析。一个缓冲区与磁盘块大小相等。把一个磁盘块读入缓冲区的时间为100μs，缓冲区数据传送到用户区的时间是50μs，CPU对一块数据进行分析的时间为50μs。分别计算在单缓冲区和双缓冲区结构下，分析完该文件的时间是多少？

引入缓冲的原因：匹配CPU和外设之间的速度差，提高外设的利用率。

缓冲区的读写操作，可以与CPU的处理工作并行。记设备读到缓冲区的时间为T，缓冲区拷贝到用户区的时间为M，CPU处理用户区数据的时间为C。由于T>C，所以系统处理每一块数据的时间为M+T=150μs。一共有十个磁盘块的数据，再加上最后一个周期的CPU处理时间50μs，总计为1550μs。



双缓冲区可以在一个缓冲区向带来的优势是，读磁盘的操作甚至可以和缓冲区向用户区拷贝数据的时间并行。因此，此时需要比较的是T和M+C的大小了。在本样例中，M+C和T都是100μs，因此，总时间为10个100μs，再加上头一个溢出的100μs，总计1100μs。



1. 假设磁盘请求以柱面 10、35、20、70、2、3 和 38 的次序到达。寻道时磁头每移动一个柱面需要 6ms，磁头起停时间忽略不计，计算以下各算法所需的寻道时间是多少:

a) 先来先服务

b) 最短寻道时间优先

c) 扫描算法，假设磁头初始位置为 15，方向向上(向大柱面号)

1. FIFO方法按顺序处理，0到10要60ms，10到35要150ms，35到20要90ms，20到70要300ms，70到2要408ms，2到3要6ms，3到38要210ms，总计为1224ms。
2. 最短寻道时间方法，0到2要12ms，2到3要6ms，3到10要42ms，10到20要60ms，20到35要90ms，35到38要18ms，38到70要192ms，总计420ms。
3. 扫描算法，从15开始找，15到20要30ms，20到35要90ms，35到38要18ms，38到70要192ms，70到10要420ms，10到3要42ms，3到2要6ms，总计738ms。
4. 请在课件的基础上进行调研，总结在实现文件系统时可以从哪些方面提高文件系统的性能，简要给出相关方法的具体解决思路。

由于磁盘与CPU性能差距很大，因此提高文件系统的性能首先可以减少对磁盘的访问次数。比如，可以为文件系统设置块高速缓存，将机械硬盘换成SSD的盘。块高速缓存为文件系统设置缓冲区，提前存储一些数据块的内容，读取时先检查数据是否在缓冲区中，如果在，则可以直接进行读操作。这种方法充分利用了程序访问数据的局部性，提高了磁盘读写效率。

同样是减少磁盘的访问次数，也可以对目录项进行分解，从当前目录进行访问，不要老是从根目录访问，减少目录的调用次数；还可以采用提前读取的策略，准备好磁盘数据，供快速读写；

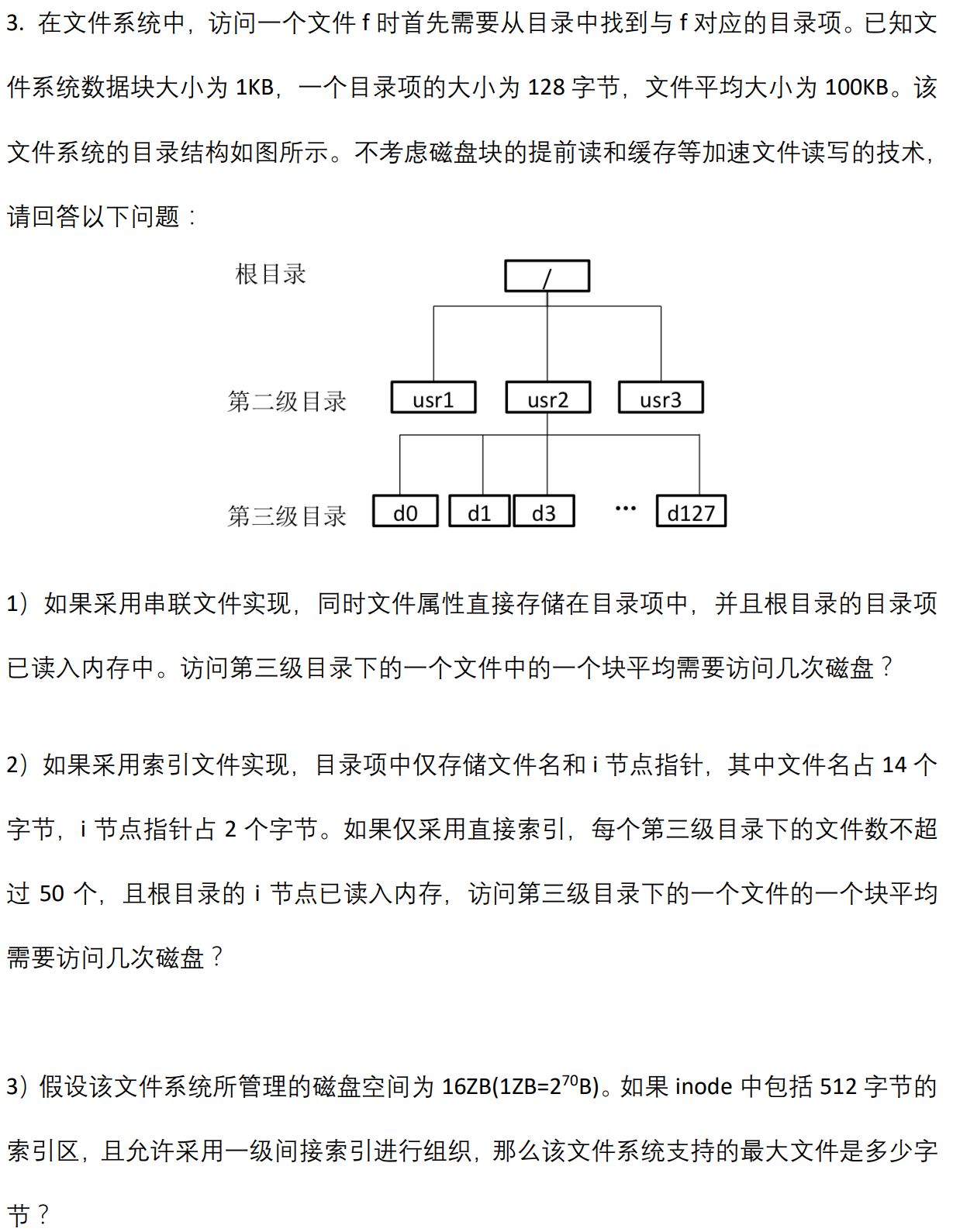
对于磁盘的性能而言，也有一些可以优化的地方。比如，整理磁盘的碎片，使得磁盘空间能够更有效利用，也能使磁头移动距离缩短，增加连续访问的可能。

1. 简述文件控制块（FCB）中一般都存储哪些信息。

首先，应包含文件的一些基本信息。文件要有一个由字符串构成的文件名，可以为不同长度。对于不同的文件存储方式（连续结构、索引结构、串联结构），需要存储一个能表明文件物理地址的数据。如对于连续结构，一个指针足矣；而对于索引结构，则需要一个索引表的首地址。另外，还有文件的逻辑结构和物理结构信息。逻辑结构指文件是流式文件（源程序，目标代码文件等），还是记录文件（串结构，顺序结构等）；物理结构指的是文件在物理存储中的存储方式，如索引、连续存储等。

其次，应包含一些访问控制信息。例如，要包含文件所有者是谁，各个用户拥有的访问权限有哪些，以保证文件的安全性和保密性。

最后，还要包括一些文件的使用信息，如创建时间、上一次修改时间、当前使用信息等。



1. 首先，根目录的目录项内容包含二级目录的FCB（以及二级目录的文件名），因此读取二级目录项不需要访问磁盘，在内存中取出即可。

然后，考虑三级目录项的读取。一个数据块1KB，一个目录项128B，因此一个数据块有8个目录项。一个二级目录包含128个三级目录项，总计16个数据块。数据块之间以串联存储，平均访问次数应为（1+16）/2=8.5次。

由三级目录项，需要经过一次访问，得到三级目录的内容——文件的FCB，需要一次访问。

最后，考虑文件的读取。文件大小为100KB，分布在100个数据块上，平均访问次数应为（1+100）/2=50.5次。

总计60次磁盘访问。

1. 首先，目前有根目录的inode（索引表），用这个inode可以取出根目录的全部内容（3个目录项，大小在一个数据块之内），用掉一次访问；

从根目录的内容中，可以取出名称为usr2的目录项内容，其中包含usr2的索引表地址。用这个索引表地址，可以读入usr2的索引表，用掉一次访问；

Usr2下面有128个目录项，而一个数据块为1KB，仅能容纳1024/16 = 64个目录项，因此这需要两个数据块来完全存储这些目录项。而根据usr2的索引表，一次从磁盘只能读取一个数据块，有可能读对，也可能不对，因此平均需要1.5次访问；

读到了3级目录的目录项之后，从里面可找到3级目录的inode指针（索引表地址），用这个指针读出3级目录的inode，用了1次；

有了3级目录的inode，由于3级目录小于50个目录项，所以一个数据块足矣，最后用一次访问，读取到这个数据块，即有了文件的FCB。

文件FCB里，找到文件自己的inode指针，读取inode，用一次；

最后，从inode内容里，读一个数据块，用一次。

总计7.5次。

1. 总计2的74次方个B，一个数据块是1KB，所以有2的64次方个数据块。一个inode索引有512B，一级间接索引。索引表的表项必须为64位，8B，有64个表项。每个表项有一个数据块1KB大小，又可以延伸出1KB/8B=128个表项，总计8K个表项。8K个表项，一个表项对应一个数据块，所以最大的文件为8K\*1KB=8MB.