# Homework 5

[Storage + Security]

1. The two I/O methods are ( **synchronous** ) and ( **asynchronous** ). Among them, means that the process should be suspended until I/O completed, while means that the process runs while I/O executes.

: synchronous; asynchronous; interrupt; DMA

1. Virtual devices are implemented with ( **buffer** ) technology, which refers to the process of placing data in a temporary working area for another program to process, such as for printers.

channel; buffer; SPOOLing; controller

1. In ( **C-SCAN** ) disk scheduling algorithm, the head moves from one end of the disk to the other and when it reaches the other end it immediately returns to the beginning of the disk, without servicing any requests on the return trip.

A. SSTF B. SCAN C. C-SCAN D. FIFO

1. When the disk is in use, a drive motor spins it at high speed. Most drives rotate 60 to 200 times per second. Disk speed has two parts. The **transfert rate** is the rate at which data flow between the drive and the computer. The **disk seek time** sometimes called the random-access time, consists of the time necessary to move the disk arm to the desired cylinder, called the seek time, and the time necessary for the desired sector to rotate to the disk head, called the rotational latency. Typical disks can transfer several megabytes of data per second, and they have seek times and rotational latencies of several milliseconds.
2. A device communicates with a computer system by sending signals over a cable or even through the air. The device communicates with the machine via a connection point, or **port**. If devices use a common set of wires, the connection is called a **bus**, which is a set of wires and a rigidly defined protocol that specifies a set of messages that can be sent on the wires. In terms of the electronics, the messages are conveyed by patterns of electrical voltages applied to the wires with defined timings.
3. A hard disk has 40G, Its each block size is 1K，and each table entry of FAT needs 20 bits，then Its FAT (File Allocation Table) need （ **A** ）memory space

A）100M B）120M C）140M D）160M

1. A disk has 10 cylinders, each cylinder has 20 tracks, each track was divided into 16 sectors. Bit Map was used managing the disk ，if the word length is 16-bit ，then the Bitmap need （ **a** ）words

A）200 B）128 C）256 D）100

1. Hard disk is a popular permanent storage media used in modern computer systems. Its basic storage unit is ⑯ **sector,** whose default size if 512 bytes. So the hard disk could be seen as a collection of addressed ⑯, whose ID is indicated traditionally by 3 numbers of ⑰**cylinder**, head and ⑯. However, except for some reserved ⑯s like MBR (Master Boot Record, the first ⑯), the entire disk is usually reorganized into ⑱ **partition** first, and then the ⑯s in each ⑱ are further reorganized as ⑲**data block**. ⑲ is the basic unit for the file system when storing the data of a file into hard disk. To store a file, the information of available ⑲s should be known first. ⑳ **bit map** is the simple one of the data structures used to represent the usage status of ⑲s, in which 1 is defined for a free ⑲, and 0 is for an occupied ⑲.

⑯⑰⑱⑲⑳: data block, sector, track, data node, cell, frame, block, partition, segment, graph, cylinder, head, MBR, CHS, linked list, node, bit map, tree structure, matrix, page, frame

1. Some file systems allow disk storage to be allocated at different levels of granularity. For instance, a file system could allocate 4 KB of disk space as a single 4-KB block or as eight 512-byte blocks. How could we take advantage of this flexibility to improve performance? What modifications would have to be made to the free-space management scheme in order to support this feature?

We could take this flexibility to not allocate all the size needed at once making it easier to fill in the entire hard disk space. You just have to take the 4Kb block as 8 512 blocks and make them link to each others.

1. Suppose that a disk drive has 5000 cylinders, numbered 0 to 4999. The drive is currently serving a request at cylinder 143, and the previous request was at cylinder 125. The queue of pending requests, in FIFO order, is 86, 1470, 913, 1774, 948, 1509, 1022, 1750, 130，Starting from the current head position, what is the total distance (in cylinders) that the disk arm moves to satisfy all the pending requests, for each of the following disk-scheduling algorithms?

a) FCFS : **7081** b) SCAN **3363** c) C-LOOK **3319**

1. Here is a file system, which adopts multi-level index structure to support the search some records in a file. The block size is 512 bytes, and 3 bytes are used to represent the block number. If the cost to the logic block number in a physical block is not considered, please figure out the largest size of a file when using 2-level and 3-level index structures.

The maximum file size is 14796800 bytes for the 2-level index structure, because there is 170 pointer per block and there is two sets of index.

With the same thinking, we can find that the maximum file size for a 3-level index structure is 2525282000bytes.

1. Given a hard disk whose size is 500GB (We use 1GB=230B here), and the block size is 8KB. 32 bytes are used to locate one block. Please answer following questions: (10 points)

a. If bit map (bit vector) is used to manage the hard disk which should be also stored in hard disk, what’s the number of blocks used to store that bit map?

There is (500GiB/8KiB) = 65536000 block on the HDD and since there is a bit used for each block in the bit map, the bit map size is also 65536000 bits.

There is 2\*\*13 byte per block that mean there is (65536000 / 2\*\*13 \* 8) = 1000 block allocated for the bitmap.

b. Now FAT (File Allocation Table) is used to link the blocks of the hard disk, and the pointer size to locate a block is 32 bytes. If the blocks for the FAT itself are connected following linked list idea as Fig. 1 (each pointer uses 32 bytes), how many blocks should be used to store the FAT?

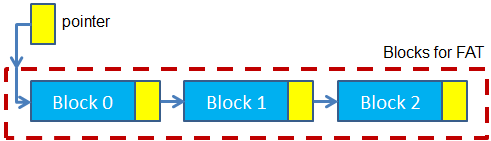


Fig. 1 Linked list idea to organize the data blocks for a file

Since there is (500GiB/8KiB) block, there must be 256 \* 1024 Blocks (or 2\*\*18).

c. Now we use 2-level index model to organize the data of a file as Fig. 2, and each index table (inner or outer) can only occupy one block at most. What’s the largest size of a file following this kind of organization?

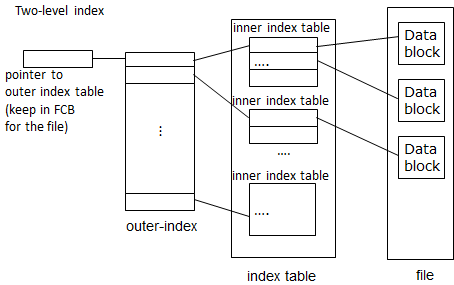


Fig. 2 2 level indexed structure to organized the data blocks for a file

There is (8KiB / 32)=256 pointers per blocks that means there is 536870912 bits in one file for the biggest file.

d. When using i-node (shown in Fig. 3) to organize the data or a file, we use all the bytes of one block to store an i-node. In that i-node, 5KB is used to represent the attributes of the file, and the rest bytes are used for indirect pointers. If the rest bytes are evenly assigned to of those 3 indirect pointers (namely, for instance, if we have 12\*32 bytes, single indirect type uses 4\*32 bytes, double indirect type uses 4\*32 bytes, and triple indirect types uses 4\*32 bytes), What’s the largest size of a file following this kind of organization?

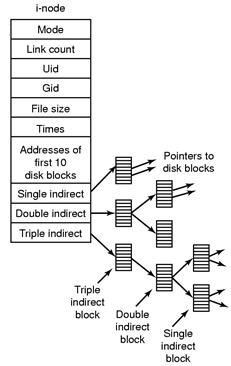


Fig. 3 Demonstration of an i-node

If we split the pointer evenly, there is 3KB left on the block for the three types of indirect block pointers, so each have 2000 pointers. Therefore we can access 2000 with the single indirect pointer, 2000\*6000 with the double and 2000\*6000\*6000 with the triple. When we sum this, we get a 72012002000blocks file, or a ~576Tb file.