EC440, Fall 2025

Homework #5

Yigong Hu, yigongh@bu.edu

Problem 1: [Anderson]

A disk may have multiple surfaces, arms, and heads, but when you issue a read or write, only one head is active at a time. It seems like one could greatly increase disk bandwidth for large requests by reading or writing with all of the heads at the same time. Given the physical characteristics of disks, can you figure out why no one does this?

Problem 2: [Ryan]

Consider a disk with the following characteristics:

- Number of surfaces: 16
- Number of sectors / cylinder: 4096
- Number of tracks / surface: 2048
- Number of bytes / sector: 512

(a) How many platters does this disk have?

(b) How many sectors per track?

(c) What is the total size (i.e., capacity) of this disk? For full credit, please show your work.

Problem 3: [Ryan]

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 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 FCFS, SSTF, SCAN, C-SCAN.

Problem 4: [Ryan]

Consider a file in the Unix file system named "/a/b/c". Assume each directory file and i-node is one block long,

(a) How many disk read operations are required to find the i-node number of this file? You may assume that all i-nodes are cached in memory. In your answer, describe the purpose of each disk read.

(b) Repeat the above exercise for finding the first data block of the file.

(c) Repeat the above exercise assuming only the root i-node block is in memory.

Problem 5: [Ryan]

Consider a UNIX-style i-node with 10 direct pointers, one single-indirect pointer, one double-indirect pointer, and one triple-indirect pointer. Assume that the block size is 4K bytes, and that the size of a pointer is 4 bytes.

(a) How large a file can be indexed using such an i-node? Explain your answer in detail.

(b) Assume that the OS has already read the i-node for a file into main memory (i.e., the file buffer cache). How many disk reads are required to read data block number 800 into memory? Explain your answer in detail.

Problem 6: [Silberschatz]

How does a file cache help improve performance? Why do systems not use much larger caches if they are so useful?

Problem 7: [Ryan]

FAT file system

(a) Can we implement symbolic links in FAT file system? If so, show how, and if not, explain why.

(b) What about implementing hard links in FAT file system?

(c) An engineer has designed a FAT-like system and he has used 24 bits for each entry. For a 32-GB disk, what is the minimum size of a file allocation in this system? Justify your answer.

Problem 8: [Anderson]

What effect will doubling the block size in the UNIX Fast File System have on the maximum file size?

Problem 9: [Ryan]

Consider a system where free space is kept in a free-space list.

(a) Suppose that the pointer to the free-space list is lost. Can the system reconstruct the free-space list? Explain your answer.

(b) Suggest a scheme to ensure that the pointer is never lost as a result of memory failure.

Problem 10: [Silberschatz]

Consider a system that supports 5000 users. Suppose that you want to allow 4990 of these users to be able to access one file.

(a) How would you specify this protection scheme in Unix?

(b) Could you suggest another protection scheme that can be used more effectively for this purpose than the scheme provided by Unix?