Matt Ritchie

CS420

HW8

1) Consider a file currently consisting of 100 blocks. Assume that the file-control block (and the index block, in the case of indexed allocation) is already in memory. Calculate how many disk I/O operations are required for contiguous, linked, and indexed (single-level) allocation strategies, if, for one block, the following conditions hold. In the contiguous-allocation case, assume that there is no room to grow at the beginning but there is room to grow at the end. Also assume that the block information to be added is stored in memory.

a) The block is added at the beginning.

b) The block is added in the middle.

c) The block is added at the end.

d) The block is removed from the beginning.

e) The block is removed from the middle.

f) The block is removed from the end.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Linked | Indexed | Contiguous |
| A | 1 | 1 | 201 |
| B | 52 | 1 | 101 |
| C | 3 | 1 | 1 |
| D | 1 | 0 | 198 |
| E | 52 | 0 | 98 |
| F | 100 | 0 | 0 |

2) Why must the bit map for file allocation be kept on mass storage, rather than in main memory?

If it was in the main memory, the free-space list would be lost in the event of a memory failure/system crash, but on the mass storage it would survive.

3) What are the advantages of the variation of linked allocation that uses a FAT to chain together the blocks of a file?

When accessing a block in the middle of a file, the location can be tracked down using the pointers in the FAT without needing to trace through all the pointers in the file to get to it. Most of the FAT can usually be cached so the pointers can be directly determined with memory access.

4) None of the disk-scheduling disciplines, except FCFS, is truly fair (starvation may occur).

1. Explain why this assertion is true.

New requests for the track over which the head currently resides can theoretically arrive as quickly as these requests are being serviced.

1. Describe a way to modify algorithms such as SCAN to ensure fairness.

Requests waiting for over a certain time are given maximum priority, and an additional bit ‘flag’ could be set which prevents any new requests from taking a higher priority and bumping this request down the queue. The rest of the queue will then have to be re-sorted.

5) 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: 143->86->1470->913->1774->948->1509->1022->1750->130=**7081**

b) SSTF: 143->130->86->913->948->1022->1470->1509->1750->1774=**1745**

c) SCAN: 143->913->948->1022->1470->1509->1750->1774->4999->130->86=**9769**

d) LOOK: 143->913->948->1022->1470->1509->1750->1774->130->86=**3319**

e) C-SCAN:143->913->948->1022->1470->1509->1750->1774->4999->86->130=**9813**

f) C-LOOK: 143->913->948->1022->1470->1509->1750->1774->86->130=**3363**