Jay Patel

CS 352

HW7

Professor Johnny Wang

11.10

The operating system can perform the following operations that would be infeasible otherwise by keeping a central open-file table. Let’s say that a file that is currently being accessed by one or more than one processes. If the file is deleted, and until all processes accessing the file have closed it, the file should not be removed from the disk. This check can be only performed if there is centralized accounting of number of processes accessing the file. But one the other side, if two processes are accessing the file, then two separate states need to be maintained to keep track of the current location of which parts of the file are being accessed by the two processes. This requires the operating system to maintain separate entries for the two processes.

11.14

The file system could prefetch/elect the subsequent blocks of future requests to these blocks, when a block is accessed. This optimization would reduce the waiting time experienced by the process for future requests.

11.17

Several concurrent updates to some results for file in user obtaining incorrect information and the file being left in an incorrect state is with a single copy. Whereas, there is storage waste and the various copies may not be consistent with respect to each other with multiple copies, there.

12.10

Below are the three techniques for allocating disk blocks for both sequential and random file access:

Contiguous sequential: As the file is stored contiguously it works very well.

Sequential access: Involves traversing the contiguous disk blocks simply.

Contiguous Random: The adjacent disk block containing the position you wish to seek works very well.

Linked sequential: As you are following the links from one block to the next hence it is satisfactory.

Linked Random: Until you arrive at the intended seek point of the file it may require following the links to several disk blocks hence it is poor.

Indexed Sequential: Sequential access involves sequentially accessing each index hence it works very well.

Indexed Random: It is easy to determine the index that is allocated with the disk block containing the position you wish to seek to hence it works well.

12.15

Let K be the starting file address

a. **Contiguous**: Divide the logical address by 512 with X as quotient and Y as the remainder.
    i. Add X to K to obtain the physical block number. Y is the displacement into that block.
    ii. 1
b. **Linked**: Divide the logical physical address by 511 with X as quotient and Y as the remainder.
    i. Run down the linked list for X + 1 blocks. Y + 1 is the displacement into the last physical block.
    ii. 4
c. **Indexed**: Divide the logical address by 512 X as quotient and Y as the remainder.
    i. Get the index block into memory. At location X, Physical block address is contained in the index block. Y is the displacement into the physical block.
    ii. 2

12.17

Data blocks would have to be read into the main memory and have to write back out to their new locations with relocation of the files on secondary storage. Moreover, since many disk files are not sequential the relocation registers apply only to sequential files. For this criterion, many new files will not require contiguous disk space; even sequential files can be allocated noncontiguous blocks if links between logically sequential blocks are within the disk system.