Tutorial 10
File System Interface, Implementation and Internals
(Chapters 13, 14, and 15)
Operating Systems CS 3SH3 Term 2, Winter 2022
Prof. Neerja Mhaskar

Tutorials are not mandatory. They are simply a tool for you to understand the course concepts better.

Tutorial Format: The questions will be posted a day before or on the day of the tutorial on the course website. You can choose to solve these problems beforehand and come in with your solutions. I or one of the TAs helping me will check your solutions. If you have all of the questions correct you can choose to leave. If you have any of them incorrect, it is recommended that you stay and understand the solutions.

Solutions to the tutorial will not be posted online.

Question 1: Describe the steps taken by a file system to create a file. Your solution should state all the data structures updated in the process.

Question 2: Describe the steps taken by a file system to open a file. Your solution should state all the data structures updated in the process.

Question 3: What problems could occur if a system allowed a file system to be mounted simultaneously at more than one location?

Question 4: Fragmentation on a storage device can be eliminated by recompaction of the information. However, recompacting and relocation of files are often avoided. Why?

Questions 5 (Disk Allocation): Consider a file currently consisting of 100 (0-99) 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.

1. The block is added at the beginning.
2. The block is added in the middle.
3. The block is added at the end.

4. The block is removed from the beginning.
5. The block is removed from the middle (the middle block considered here is the 50th block).
6. The block is removed from the end.

Question 6 (Disk Allocation): Consider a file system on a disk that has both logical and physical block sizes of 512 bytes. Assume that the information about each file is already in memory. For each of the three allocation strategies (contiguous, linked, and indexed), answer the following questions: If we are currently at logical block 10 (the last block accessed was block 10) and want to access logical block 4, how many physical blocks must be read from the disk?

Question 7 (Disk allocation):

Consider a file system that uses inodes to represent files. Disk blocks are 8-KB in size and a pointer to a disk block requires 4 bytes. This file system has 12 direct disk blocks, plus single, double, and triple indirect disk blocks. What is the maximum size of a file that can be stored in this file system?

Questions 8 (Free space management): Consider a system where free space is kept in a free-space list.

1. Suppose that the pointer to the free-space list is lost. Can the system reconstruct the free-space list? Explain your answer.
2. Suggest a scheme to ensure that the pointer is never lost as a result of memory failure.

Question 9 (Free space management):

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?

Question 10 (Protection in Unix/Linux): What access rights does the command:

`chmod 531 test.txt`, specify on the file test.txt?