

- 1) In a UNIX File System, the size of a block is 4KB, and each block can hold 1024 (210) block addresses. Assume that there are 12 direct block pointers and a singly, doubly and triply indirect pointer in each inode.
 - a) Using the inode scheme, what is the maximum size of a file?
 - b) Assuming no information other than the file inode is already in main memory, how many disk accesses are required to access the byte in position 13,423,956? (Hint: If you don't have a calculator with you, this information might save you time: $2^{23} < 13,423,956 < 2^{24}$)
- 2) For this question about basic file systems, assume a simple disk model where each disk read of a block takes D time units. Also assume the basic layout is like the very simple file system or the Fast File System.
 - a) Assume that all data and metadata are initially only on disk. Assume further that all inodes are in separate blocks, and that each directory is only one block in size. How long does it take to open the file /a/b/c/d.txt?
 - b) Assume after opening the file, we read the file in its entirety. It is a big file, containing 1036 blocks. The inode itself has room for 12 direct pointers and 1 indirect pointer. Disk addresses are 4 bytes long, and disk blocks are 4KB in size. After opening it, how long does it take to read the entire file?
 - c) Now assume a new inode structure is introduced, in which there is only one pointer: a double indirect pointer, which points to the double indirect block, which can point to 1024 indirect blocks, each of which can point to 1024 blocks. After opening the file, how long does it take to perform 50 random reads within a very large file?
 - d) How long does it take to close a file, approximately (assuming disk accesses are the dominant cost)?
- 3) Crash and File System Consistency:
 - a) Fskck is the Unix file system crash recovery program. For each of the following fsck error messages, what do you believe fsck has seen in the crashed file system to generate the error message:
 - i) File's inode link count is 1, should be 2.
 - ii) Free bitmap entry for block 3323 is 0, should be 1 (allocated).
 - b) Which of the following conditions would likely represent a serious problem with a file system (Justify answer for each.)
 - i) A write to a data block that contains no on-disk inodes pointing at it.
 - ii) A write to a data block that contains multiple on-disk inodes pointing at it.
 - iii) A write to a data block that is marked as free in the on-disk bitmap.