1999 Operating Systems

p1q12
SMH

d) What is a *hard link*? Does file system support for hard links place any restrictions on the location of file meta-data? [2 marks]

e) What is a *soft* (or *symbolic*) link? Does file system support for soft links place any restrictions on the location of file meta-data? [2 marks]

Describe with the aid of a diagram a UNIX *inode*. You should explain the motivation behind *indirect blocks*, and how they are used when accessing a file. [8 marks]

Answers to Question 3

Functions of Directory Service

One mark for each function

The directory service is responsible for *name resolution* (means by which pathnames/filenames are resolved to file meta-data, and for *access control.*

Directory Hierarchy

A directory hierarchy is a tree with variable branching factor (the maximum being implementation-specific). Each non-leaf node in the tree is a directory. Most leaf nodes are files, although some may be directories.

The illustration should clearly show these distinctions.

File meta-data

Two marks for getting below

The most important part of file meta-data is the information describing where on disk the contents of the file are to be found. This may be a simple table of blocks, or a hierarchy of tables, etc. etc.

One mark each for any further points. Example below.

Other file meta-data information includes the file size (i.e. how much of last block valid), the file type (dir, link, reg), the owner of the file, permissions on the file, the time of last access/creation/modification, the number of links (references) to the directory, etc.

Hard links

One mark for definition, one for meta-data location.

A hard link is a reference-counted "copy" of a file, a directory entry which refers to the same meta-data.

A system using hard links cannot keep file meta-data embedded within directories; an extra level of indirection is required. *A creative answer which describes how one could still keep meta-data in directories also deserves a mark providing sufficient detail is given.*

Soft links

One mark for definition.

A soft link is a "file" containing the pathname/filename of another file. In some implementations, the pathname will actually be contained within the directory entry; in others, a real file will be created.

One mark for "no".

Soft links do not place restrictions on the location of meta-data.

Unix inodes

Roughly two marks for a basic diagram/explanation, two marks for the motivation behind indirect blocks, and four marks for the full explanation of how indirect blocks work.

An UNIX inode holds file meta-data as described previously. It also holds the locations of the first n (e.g. 10) disk blocks are held in the inode, followed by the location of a single indirect block, followed by the location of a double indirect block, followed by the location of a triple indirect block.

Indirect blocks represent a trade-off between the speed of block location and the size of an inode. Since many files are small, having too many "direct" pointers in the inode would waste space. Using (the three kinds of) indirect blocks allows very large files to be supported, while not penalising small ones.

In a typical UNIX system, the first 10 blocks of a file are directly pointed to. Once this is determined, the file system can load the relevant block from disk.

Given a 512-byte block size, and a 4-byte block address, the next 128 blocks are obtained by a) first reading the single indirect block, and b) indexing into this using $b - 10$, where b is the block offset in the file, and c) loading the relevant block from disk.

The next 128^2 blocks are found by reading the double indirect block, indexing into this using $(b - 138)/128$ to obtain the location of a single indirect block, reading the single indirect block, indexing into this using $(b - 138)\%128$, and finally reading the relevant block from disk.

The final 128^3 are found via the triple indirect block. This contains the locations of up to 128 further double indirect blocks. Indexing follows the obvious scheme.

2 Paper 2

Short Question 1 (1 Mark)

What is the range of real numbers which can be represented by a floating point format with 1 sign bit, 8 bias-127 exponent bits, and 23 normalised mantissa bits?
You may round the mantissa value to the nearest integer. [1 mark]

Answer to Short Question 1

The exponent can range between -127 and $+128$, the mantissa between 1 and ~ 2. Hence the range is $(-2.2^{128}) - (+2.2^{128})$, or $-2^{129} - +2^{129}$.

Since in practise (e.g. IEEE single precision) certain exponent values are reserved (0 and 255), an answer which takes this into account and arrives at $-2^{128} - +2^{128}$ should get a mark *providing they state their assumption.*

Short Question 2 (1 Mark)

UNIX operating systems give each process its own virtual address space, initially containing three segments. What is contained in each of these segments?

Answer to Short Question 2

The three segments are the `text` (or `code`), `data` and `stack` segments; the first contains the program image (including perhaps read-only data), the second holds the initialised data followed by enough demand-zero space to hold the uninitialised data, followed by (eventually) the heap, while the last holds the stack of the process (a region of demand-zero pages).

This much detail is not required for the answer; a simple listing of the three segments will do. An answer which lists *four* (e.g. including the `bss` as a separate segment), or similar valid extensions, should also gain a mark. Omitting any of the above three should gain no marks.