OPERATING SYSTEM
WEEK 14 ASSIGNMENT

1. Explain different directory implementation methods.

The directory implementation algorithms are classified according to the data structure they are using. There are mainly two algorithms which are used in these days.

1. Linear List

In this algorithm, all the files in a directory are maintained as singly lined list. Each file contains the pointers to the data blocks which are assigned to it and the next file in the directory.

2. Hash Table

To overcome the drawbacks of singly linked list implementation of directories, there is an alternative approach that is hash table. This approach suggests to use hash table along with the linked lists.

A key-value pair for each file in the directory gets generated and stored in the hash table. The key can be determined by applying the hash function on the file name while the key points to the corresponding file stored in the directory

2. Describe in detail about free space management with neat.

- Since disk space is limited, it needs to reuse the space from deleted files for new files
- Write-once optical disks allow only one write to any given sector, and thus reuse is not physically possible
- To keep track of free disk space, the system maintains a free-space list.
- Free-space list records all free disk blocks
- Some sort of data structure is required to keep track of which file blocks are allocated, and which are free.
- To create a file, search the free-space list for the required amount of space and allocate that space to the new file. This space is then removed from the free-space list.

When a file is deleted, its disk space is added to the free-space list.

The free-space list, despite its name, may not be implemented as a list

Bit Vector

- A Bitmap or Bit Vector is a series or collection of bits where each bit corresponds to a disk block
- The free-space list is implemented as a bit map or bit vector
- A bit map or bit vector is a sequence of bits. the ith bit represents the ith physical block. If the ith physical block is free, the ith bit in the vector is 1, else it is 0.

- It is simple to implement bit-vectors and devise algorithms for locating free blocks and runs of contiguous free blocks. Instructions that might be used: "ISZERO" and bit-shift.
- However, bit vectors are not efficient to use unless they are cached entirely in primary memory. It is fairly common nowadays (the year 2019) for a laptop computer to have a terabyte disk and 16GB of primary memory. If the disk has 4KB blocks or clusters, the bit vector would need about 32 MB of physical memory, which is about 0.2% of the 16GB.

bit[*i*] = 0 Þ block[*i*] free

1 Þ block[*i*] occupied

3.Explain in detail about tree structured and acyclic graph directories.

Once we have seen a two-level directory as a tree of height 2, the natural generalization is to extend the directory structure to a tree of arbitrary height. This generalization allows the user to create their subdirectories and to organize their files accordingly. A tree structure is the most common directory structure. The tree has a root directory, and every file in the system has a unique path.

Advantages:

1. Very generalize, since full pathname can be given.
2. Very scalable, the probability of name collision is less.
3. Searching becomes very easy, we can use both absolute paths as well as relative

Disadvantages:

1. Every file does not fit into the hierarchical model, files may be saved into multiple directories.
2. We can not share files.
3. It is inefficient, because accessing a file may go under multiple directories.

Acyclic graph directory

- An acyclic graph is a graph with no cycle and allows to share subdirectories and files. The same file or subdirectories may be in two different directories. It is a natural generalization of the tree-structured directory.

- It is used in the situation like when two programmers are working on a joint project and they need to access files. The associated files are stored in a subdirectory, separating them from other projects and files of other programmers, since they are working on a joint project so they want the subdirectories to be into their own directories. The common subdirectories should be shared. So here we use Acyclic directories.

- It is a point to note that the shared file is not the same as the copy file . If any programmer makes some changes in the subdirectory it will reflect in both subdirectories.

Advantages:

- We can share files.
- Searching is easy due to different-different paths.

Disadvantages:

- We share the files via linking, in case deleting it may create a problem,
- If the link is a soft link then after deleting the file we left with a dangling pointer.
- In case of a hard link, to delete a file we have to delete all the references associated with it.