unit-4 File Management

Q:> List and explain the several pieces of information that are associated with open file. [~~chap-12~~ unit-(11)]

Ans:- Several pieces of data are needed to manage open files:-

1.> Open file table: Tracks open files

2.> File pointer: pointer to last read/write location, per process that has the file open.

3.> File open count: counter of number of times a file is open - to allow removal of data from open-file table when last process closes it.

4.> Disk location of file: cache of data access information.

5.> Access rights: per-process access mode information.

OF2DA

Q:> Differentiate between linear list and hash table implementation of a file directory. [chap.12]

Ans:- Linear List of file names with pointer to the data blocks.

→ Simple to program

→ Time consuming to execute

* Linear search time
* Could keep ordered alphabetically via linked list or use B+ tree.

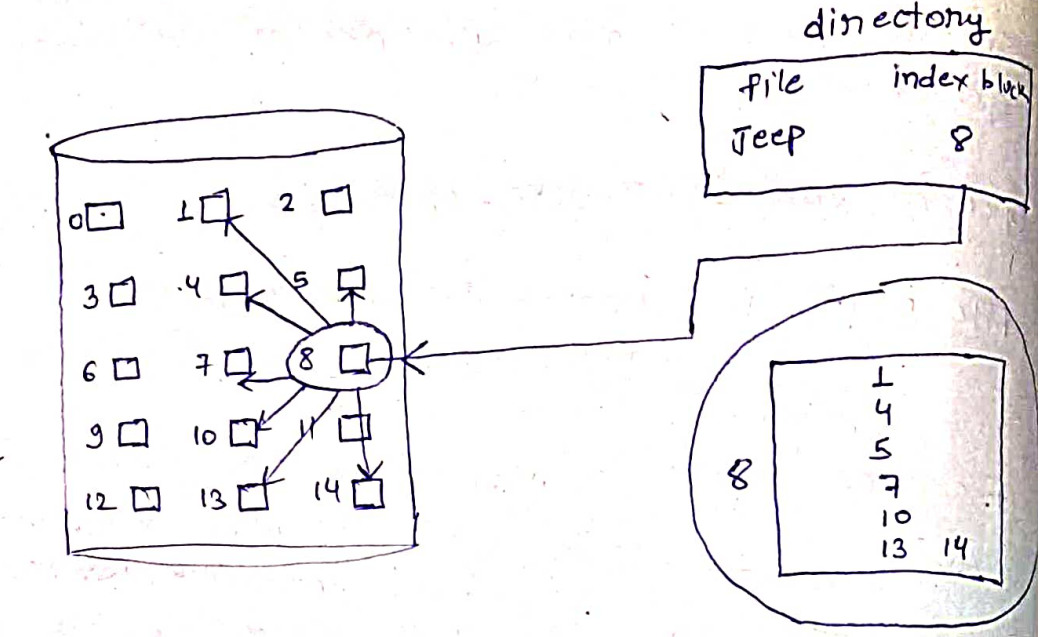
Hash table:- linear list with hash data structure

→ Decreases directory search time

→ Collisions: situations where two file names hash to the same location.

→ only good if entries are fixed size, or use chained-overflow method.

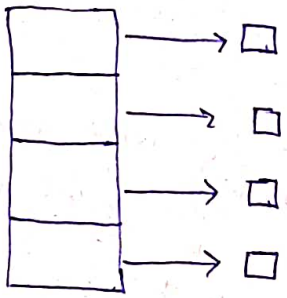
Q) Explain indexed file allocation with neat diagram.



Indexed allocation:

Each file has its own index block(s) of pointers to its data blocks.

Logical view:-



Need Index table, Random access

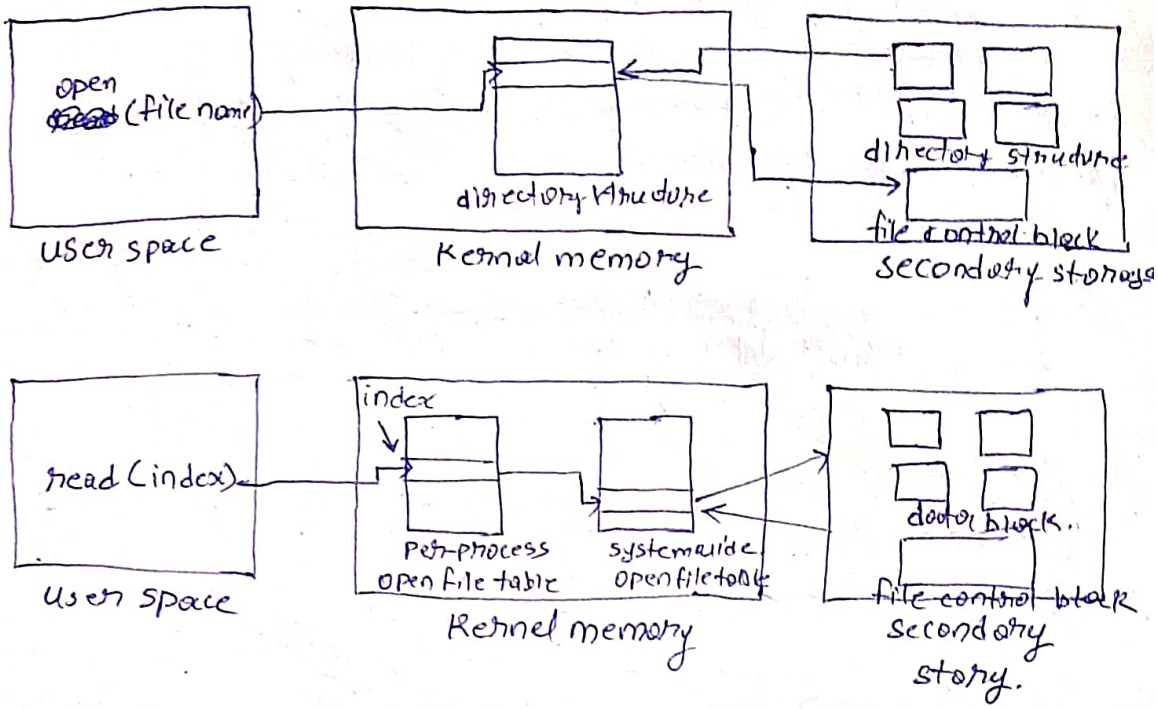
Advantages:-

1) It supports direct access
2) No external fragmentation

Disadvantages

1) overhead pointer
2) Multilevel Indexes.

Q) Illustrate the file open and file read operations of in-memory file system structure.



open return a file handle for subsequent use.

Data from read eventually copied to specified user process memory address.

Q.) Explain tree structured and acyclic graph directories with examples.

Sol:- A tree structure is the most common directory structure. The tree has a root directly and every file in the system has a unique path., help in making new subdirectories.

Advantages:

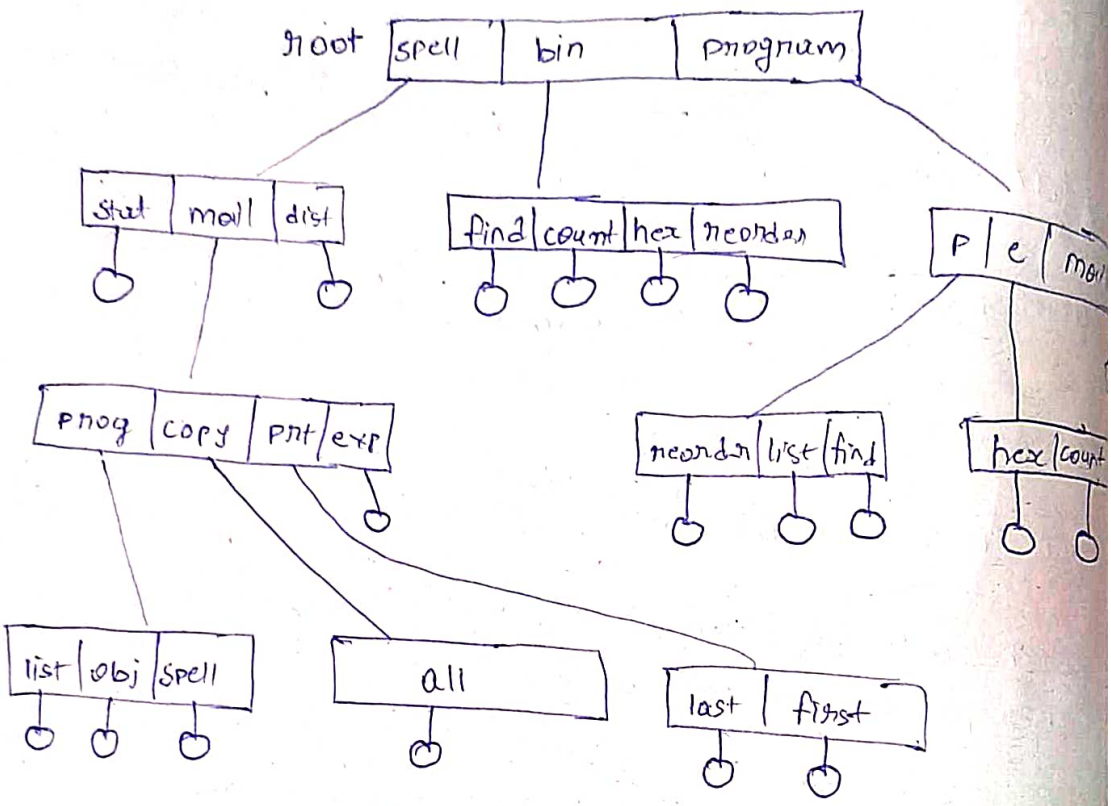
* very general, since full pathname can be given.
* very scalable, the probability of name collision is very low
* Searching become very easy, we can use both absolute path as well as relative.

Disadvantage:

1) Every file does not fit into the hierarchical model, files may be saved into multiple directory.

2) We cannot not share file

3) It is inefficient, because assessing a file may go under multiple directories.



⊜ Acyclic graph directory:

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

Advantage:

1) We can share files

2) Searching is easy due to different different path.

Disadvantages:

1> We share file via link, deleting it may create problems

2> If the link is soft link then after deleting the file we left with a dangling pointer.

3> In the case of hard link, to delete a file we have to delete all the references associated with it.