FILES

The basic input/output (I/O) operations where input to the system may be provided through the keyboard by using scanf() function in C and cin>> in C++. The output may be taken onto the computer's screen by printf() of C or cout<< of C++.

There are some limitations with these functions such as:

(1) It is very difficult to handle large volume of data.
(2) It is difficult to store the input data for future use.
(3) It is difficult to store the output data for future use.
(4) Some of the input data are in image form or in text form.
(5) The data are stored into the main memory which has limited size.

To deal with such problems, there is a special kind of data structures called files. A file is also called an external data structure.

Basic Terminology

a) **field**: It is the basic unit of the record which represents meaningful information about the record. eg name, age, sex, roll-no.

b) **Record**: It is a collection of logically related fields about any real world entity.
ex. Student is a collection of logically related information such as name, age, sex, roll no etc.

c) **file**: A file is a collection of records.
ex A file of students of CSE branch.

d) **file organization**: It is concerned with the arrangement of records on the disks.

e) **key field**: It is the field that contains a unique value for each record, so the record in the file is uniquely identified by specifying the value of a particular field of the record.

f) **Index**:- It is a pointer used to determine the location of the record in a file that satisfies some condition.

Basic operations to the file

a) **Creation**: First the file structures is created before feeding the data into the records. Creation of the file determines the name of the

file; the position of the file where the I/O operation is performed; the file structure defined, whether it is a text file or binary file and the access method defined.

b) **Open**: This operation prepares the file for read and write operations.

c) **Retrieve or Search**: This operation is performed to search for a record or a set of records having a particular value in a particular field or where the field value satisfies certain conditions.

d) **Insert**: The operation deals with the insertion of a new record or set of records at a specific location.

e) **Update**: The existing records of the file may be updated by changing the values of the fields or the new record may be inserted.

f) **Delete**: Any record or a set of records may be deleted from the file.

Classification of files [on the basis of access method or way of data stored]

(1) **master file**:

A master file represents the static view of some aspects of an organisation at a particular point of time. It contain records which are permanent in nature and not frequently

changed.

eg name, address, DOB etc.

(2) Transaction File :-

A transaction file contains the collection of data that are applied to a master file to reflect any changes in the master file. The transaction file is also called the "log" file.

(3) Program File : It contains instructions in high level or assembly or m/c languages to process the data. This may be stored in the main memory of in some other files.

(4) Text files : The character or numeric data are stored in the form of respective ASCII code. These kind of files contain alphanumeric and graphic input data.

(5) Binary Files : These are similar to the text files except these allow writing the numeric data in less number of bytes as compared to text files.

but we cannot read the data directly from the files as they are written. we can read the files only through the progeams containing some special commands to read them.

FILE ORGANIZATION

A file is a collection of logically organized records which are mapped onto disk blocks. A key element in file management is the way in which the records themselves are organized inside the file. File organization refers to the way the files are logically arranged on a file system and the physical arrangement of data in a file into records and pages on the disk.

Choosing a file organization is a design decision, so we must consider some factors that effect the organization of a file.

(a) Storage efficiency
(b) The nature of the operations that are to be performed on the file.
(c) Easy record addition/updation/removal, without disruption.
(d) The frequency and fast access of file.
(e) The external storage medium on which the file is stored.
(f) Ease of information retrieval.
(g) Redundancy, reliability, security and integrity.
(h) Response time in completion of a file operation

The most common file organization techniques based on their physical storage and keys used to access records are as follows:

(1) Sequential organization.
(2) Direct Organization.
(3) Indexed Sequential Organization.

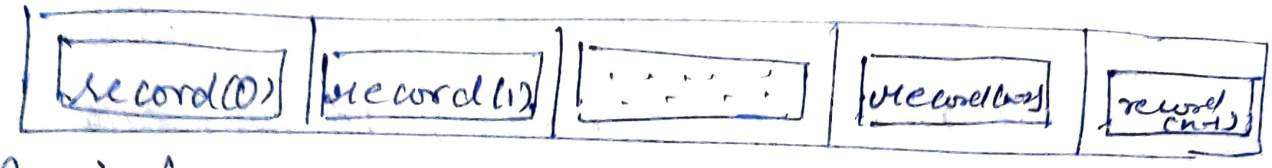
(1) Sequential file Organization:-

This is the most common technique of file organisation. In this method, the records are arranged one after another in a sequence when the files are created. So in the sequential file organisation, the records are placed sequentially on the storage medium.

The records are kept in sequence to their key value. The key value is the value of some field which uniquely identifies the records. The records can be accessed in the order of their key values.

The nth record of the file cannot be accessed directly until we traverse the preceding (n-1) records.

The order of records is fixed and they can only be read or written sequentially.



Beginning of file

End of file

Fig: Sequential organisation of file.

If a new element or field is added to the record then the entire file is reorganized. Deletion is also done in the similar fashion. But it is not done immediately. The records are always marked for deletion and it is written to the "log file" also called Transaction file. n then the record is dropped from the primary data file.

Advantages:

(1) Easy to process and implement.
(2) Minimizes the number of block accesses.
(3) A sequential file is accessed one record at a time, from first to last, in order. Each record can be of varying length.
(4) There is no overhead overhead to calculate any address calculation function to locate a particular record, only the key value is sufficient to locate it.
(5) Sequential files can be implemented on magnetic tapes as well as on disks.

(6) well suited for batch processing application.
(7) Record in a file can be of varying size.

Disadvantages:

(1) It is difficult to maintain physical sequential order as records are inserted and deleted.

(2) Insertion of new field requires reorganisation of files which is expensive; therefore updates are not easily accomodated.

(3) The records cannot be directly accessed so random access is not possible.

(4) Data redundancy is very high because the same data are kept in various files which are sorted on different fields.

(5) A record of a sequential file can only be accessed by reading all the previous records.

(6) Especially not good for programs that make frequent searches in the files.

Direct file organisation

The disadvantage of the sequential organisation is that the records can only be accessed in the order of the key value. The direct file organisation technique allows us to access the records directly. Records are accessed by addresses that specify their disk location. This technique is also called as relative file organisation or random file organisation.

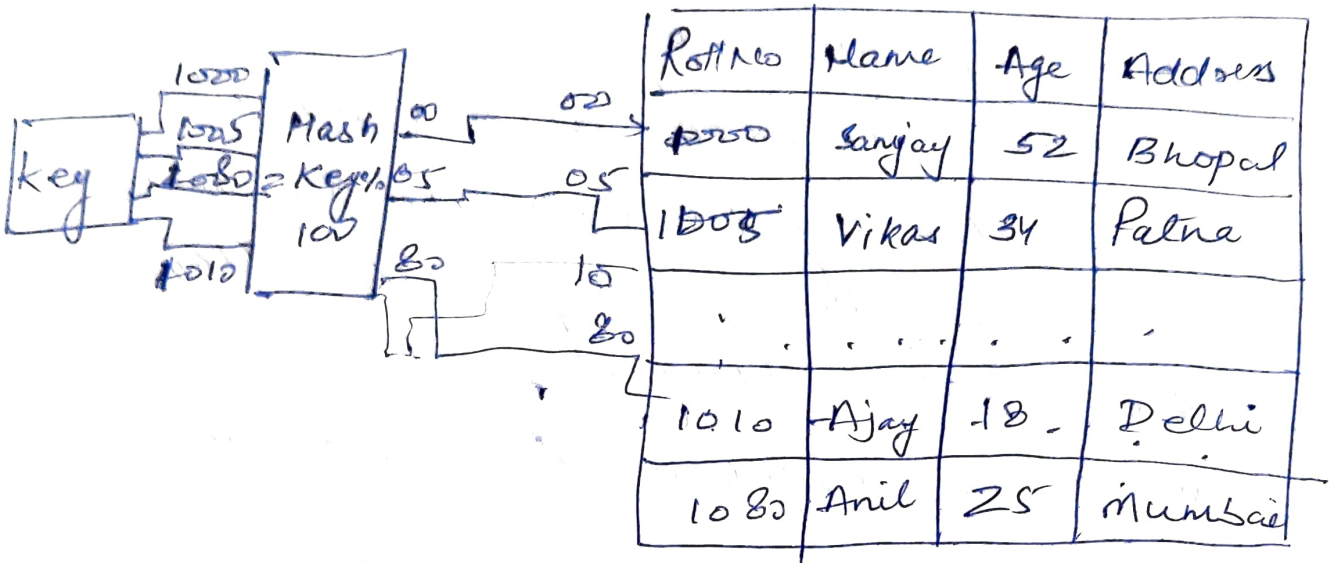
The relation b/w the key value and the physical address is established by using a mapping function commonly known as hash function as given below:

$$H(key) = physical\ address$$

The technique to convert the record's key into physical storage address is called hashing and the mapping function is called hash function.

1. division method
2. mid-square method
3. folding method

The address generating function often map a large number of records to the same address. This situation is known as collision.



Advantage:

(1) A direct organisation is an effective way to organize a file when there is need to access individual records directly.

(2) Records can be directly or randomly accessed.

(3) A relative file can also be accessed sequential-ially files cannot be accessed directly.

(4) Well suited for interactive application.

(5) Updating the records is possible and can be easily implemented.

(6) It has an excellent search retrieval performance.

Disadvantage

(1) It requires the calculation of mapping function

(2) Records are scattered.

(3) To get the advantages of this file organization the files must be stored on some direct storage device like disk.

(4) A key field is required for this organisation as well as fixed record length.

(5) Additional space is required.

Indexed Sequential file

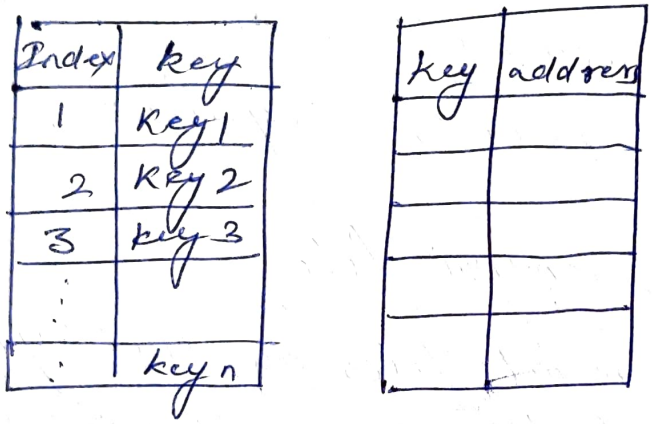
Indexed Sequential files are the hybrid of sequential and direct access files which are used to effectively organize the data. Such data can be accessed directly through some key and it can also be accessed sequentially through the same key.

Since indexed Sequential file organisation support both direct and Sequential file organisation method, so it is better for the applications that require both batch and interactive processing.

An indexed Sequential file has two parts: Index part and Sequential part.

Index would be built as a tree of key values where each node contains a pointer to the Sequential data file.

The indexes speed up the search process.



Index	key
1	key1
2	key2
3	key3
:	
:	keyn

key	address

Advantage:

(1) This allows the Sequential as well as random access of records.

(2) Due to its capability of supporting both Sequential and direct file organisation. It is better for interactive as well as batch oriented applications.

Disadvantages:

(1) It allows storage only on the disks.

(2) It supports direct access so involves overhead of calculating the mapping function.

(3) The records can only be fixed sizes.