THIRUTHANGAL NADAR COLLEGE

(Belongs to the Chennaivazh Thiruthangal Hindu Nadar Uravinmurai Dharma Fund)

Selavayal, Chennai-51.

A Self-Financing Co-educational College of Arts & Science

Affiliated to the University of Madras

Accredited with ‘B’ Grade by NAAC

An ISO 9001: 2015 Certified Institution

NAME OF THE DEPARMENT: BSC COMPUTER SCIENCE

SUBJECT : DATA BASE MGT. SYSTEMS

TOPIC : STORAGE METHODS

STAFF NAME : Mrs.P.T.KASTHURI BAI

Data storage methods :

Sequential storage method:

Sequential storage it is the simplest method of storing data. Each row is stored in a predefined order. Access is fast enough as long as the data is retrieved in the specified order.

Uses/advantages :

- Always retrieval is in retrieval is in fixed order
- Suitable for file containing common data.
- Taking backup of data is easier.

Drawbacks:

To read the n^{th} row of data we have to read n -1 rows. Similarly when we need to insert a row or delete a row or update a row we have to sequentially read n number of rows

To overcome the problem we can you pointers indexes and linked list.

Pointers:

When the data is stored it is stored in some memory location. This location is identified by some type of address. A variable that points to this address is called appointer. A disk drive is physically divided into several pieces. An individual drive is a volume. Each drive consists of several platters. Each side of the platter is accessed by a drive head. Each side is divided into tracks of data. The tracks are further split into sectors or cylinders. Data is stored a certain number of bytes from the start of a sector known as an offset.

Indexes.

An index is the most common method used to provide faster access to the data .

An Index sorts and stores the key values from the original table along with the pointer to the rest of the data in each row.

Index provide fast random and sequential access on table from any predetermined sort condition.

Indexes are small enough to fit into RAM

The database system can rapidly search the index and use the pointer to retrieve the desired data with Almost no access to the disk drive.

Linked list:

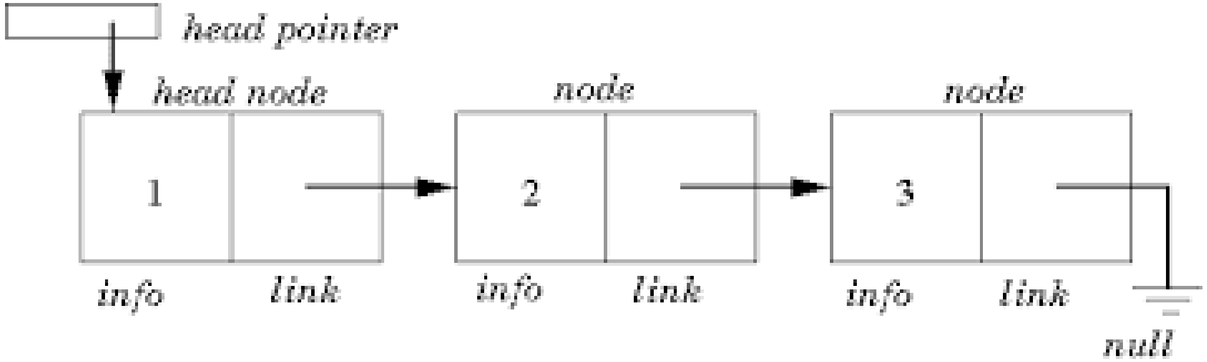
With the linked list any data element can be stored separately. The pointer is then used to link the next data item. An index element consists of three parts :

1.the key value to the pointer

2. The associated data element

3. A pointer to the next Index element.

The strength of the linked list lies in its ability to easily and rapidly insert and delete data. Inserting a row into a linked list requires three basic steps first store the data and the Index element keeping the address of each. Second find the point in the index to insert the new row using a binary search. Third change the link pointer .



A Linked List

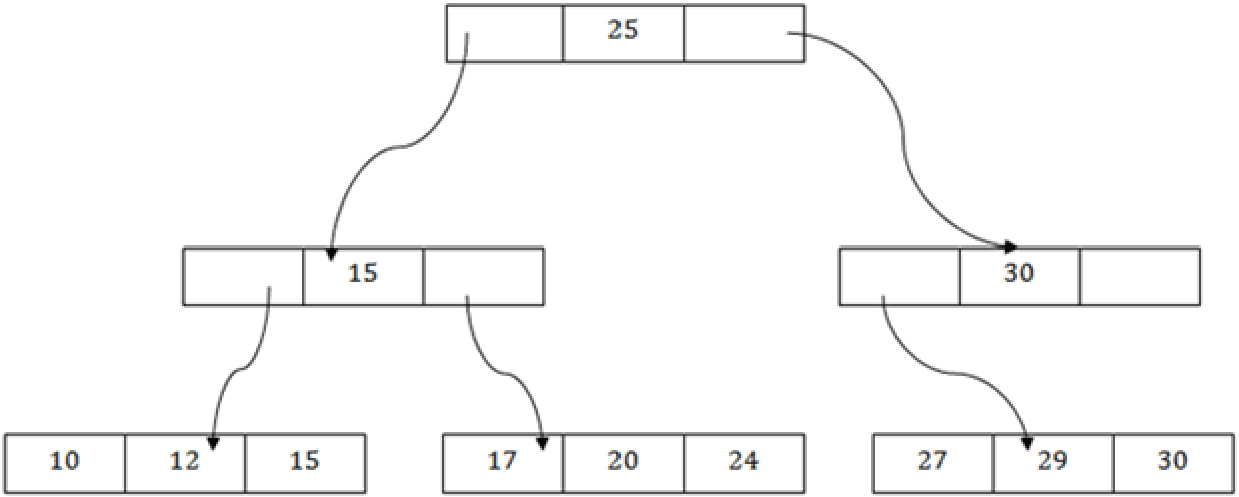
B+ tree

B plus tree consists of one hour more blocks of data called nodes link together bye pointer. The tree has single node at the top point called root node. The root node points to two are more blocks called child nodes. There are 6 basic rules .

B+ tree file organization is the advanced method of an indexed sequential access method. It uses a tree-like structure to store records in File.

It uses the same concept of key-index where the primary key is used to sort the records. For each primary key, the value of the index is generated and mapped with the record.

The B+ tree is similar to a binary search tree (BST), but it can have more than two children. In this method, all the records are stored only at the leaf node. Intermediate nodes act as a pointer to the leaf nodes. They do not contain any records.


25
15
30
10
12
15
17
20
24
27
29
30

The above B+ tree shows that:

There is one root node of the tree, i.e., 25.

There is an intermediary layer with nodes. They do not store the actual record. They have only pointers to the leaf node.

The nodes to the left of the root node contain the prior value of the root and nodes to the right contain next value of the root, i.e., 15 and 30 respectively.

There is only one leaf node which has only values, i.e., 10, 12, 17, 20, 24, 27 and 29.

Searching for any record is easier as all the leaf nodes are balanced.

In this method, searching any record can be traversed through the single path and accessed easily.

Direct or Hashed access

A Direct Access storage method is used for superfast Random Access to data . This method works by first setting aside enough space to store all the key values. The key value is converted to a storage location number.

Hashing techniques :

The principle idea behind the hashing technique is to provide a function h, called a hash function, which is applied to the hash field value of a record and computes the address of the disk block in which the record is stored. A search for the record within the block can be carried out in a buffer, as always. For most records, we need only one block transfer to retrieve that record. Suppose K is a hash key value, the hash function h will map this value to a block address in the following form: h(K) = address of the block containing the record with the key value K If a hash function operates on numeric values, then non-numeric values of the hash field will be transformed into numeric ones before the function is applied. The following is an example of possible hash functions: • Hash function h(K) = K mod M: This function returns the remainder of an integer hash field value K after division by integer M. The result of the calculation is then used as the address of the block holding the record.

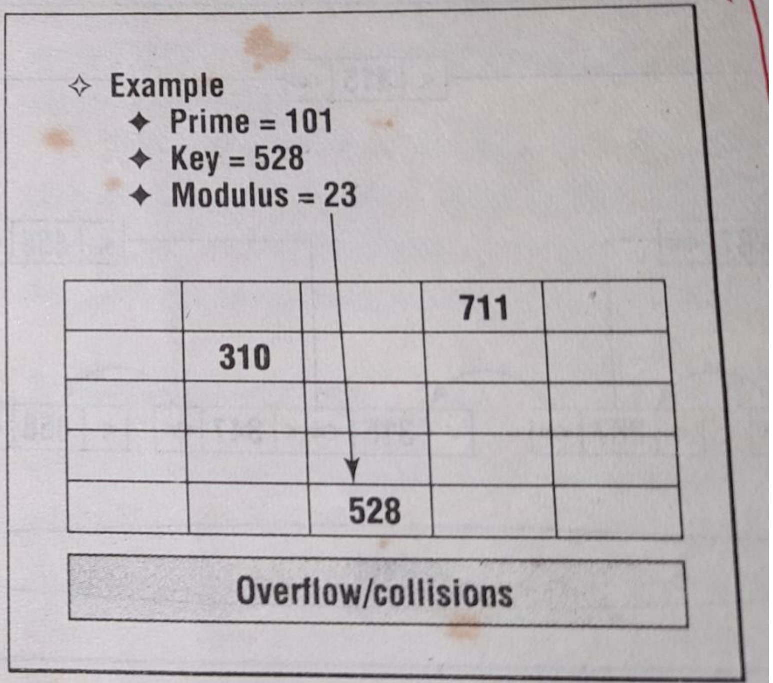


FIGURE 9.16 Hashed-key access. The key value (528) is converted directly into a storage location by dividing by a prime number (101). If two keys have the same remainder, one is stored in an overflow location.

Advantages :

- This method is extremely fast for finding and storing random data.
- The key value is converted into storage location and the data can be retrieved in one pass to the disc.
- It works best for transaction operations that requires instantaneous retrieval of small amount of data.
- It works best if the data does not change very often.