

ASSIGNMENT-V

PAGE NO.:

DATE: / /

Section - A

Q-1.) The method of file organisation in which data records in a file are arranged in a specified order according to key field is

Aus- a.) Sequential access method.

Q-2.) File code which engineers add to the file name & limit access to a few users is called

Aus- b.) access code

Q-3.) A _____ file system is software that enables multiple computers to share file storage while maintaining consistent space allocation & file content

Aus- d.) Cluster

Q-4.) The file organisation that provides very fast access to any arbitrary record of a file is

Aus- c.) Hashed Tree

Q-5.) A huge collection of the information or data accumulated from several different sources is known as

Aus- c.) Data Warehouse

Q-6.) Which of the following can be used to extract or filter the data & information from the data warehouse?

Aus- c.) Data Mining

Q-7.) Which one of the following refers to the copies of the same data (or info) occupying the memory space at multiple places?

Aus- d.) Data Redundancy

Q-8) Which one of the following refers to "data about data"?

Ans - d) Metadata

Q-9) Which of the following is True?

Ans - c) B+ tree allows rapid random access as well as rapid sequential access.

Q-10) Before use of DBMS information was stored using

Ans - a) File Management system

Q-1) Define heap file organisation.

Ans - It is the simplest and most basic type of organisation. It works with datablocks. In heap file organisation, the records are inserted at the file's end. When the records are inserted, it doesn't require the sorting and ordering of records. → when the data block is full, the new record is stored in some other block. This new block need not to be the very next block, but it can ~~has to~~ select any data block in the memory to store new records. This heap file is also known as an unsorted file.

Q-2) What is sequential file organisation?

Ans - A sequential file contains records organised by the order in which they were entered. The order of the records is fixed. Records in sequential files can be read or written only sequentially. After you place a record into a sequential file, you cannot shorten, lengthen or delete the record.

Q-3) Define hash file organisation.

Ans - Hash file organisation uses the computation of hash function on some fields of the records. The hash function's output determines the location of disk blocks where records are to be placed. When a record has to be retrieved using hash key columns, then the address is generated and the whole record is retrieved using that address. In the same way, when a new record has to be inserted then the address is generated using the hash key & record is directly inserted. The same is done for delete & update.

Q-4.) What is a file system.

Ans - File system is the way in which files are named and logically placed for storage & retrieval. Without this, stored information wouldn't be isolated into individual files & would be difficult to identify and retrieve.

Section-C.

Q-1) Explain the B+ tree indexes of multiple keys with a suitable eg.

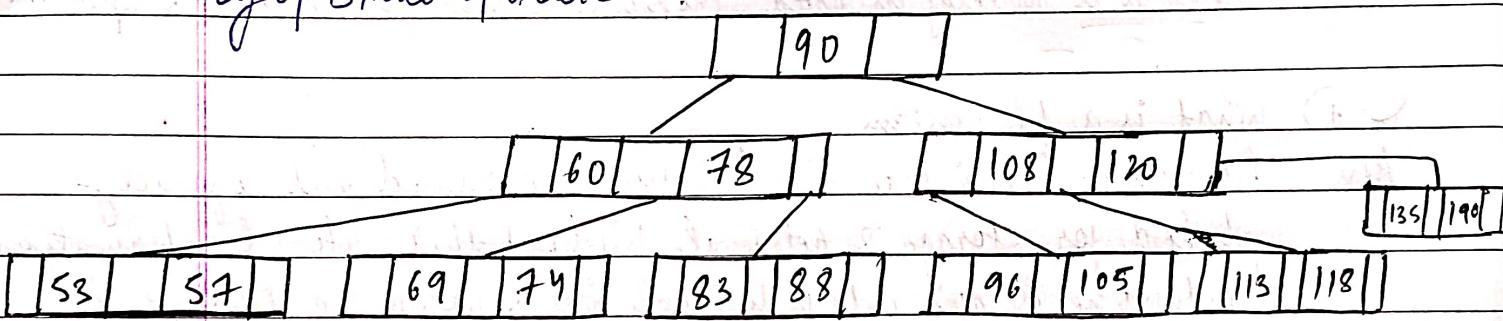
Ans - A B+ tree is an advanced self balancing tree. It uses a multilevel indexing system. Leaf node in the B+ tree represent actual data references. It keeps all of the leaf nodes at the same height. A link list is used to connect the leaf nodes in B+ tree so it can allow both the random & sequential access.

→ Both keys and records can be placed in the internal & leaf nodes of the B+ tree. In a B+ tree records or data can only be kept on the leaf nodes, whereas key values can only be placed on the internal nodes. To make search queries more efficient, the leaf nodes of the B+ tree in data structure are

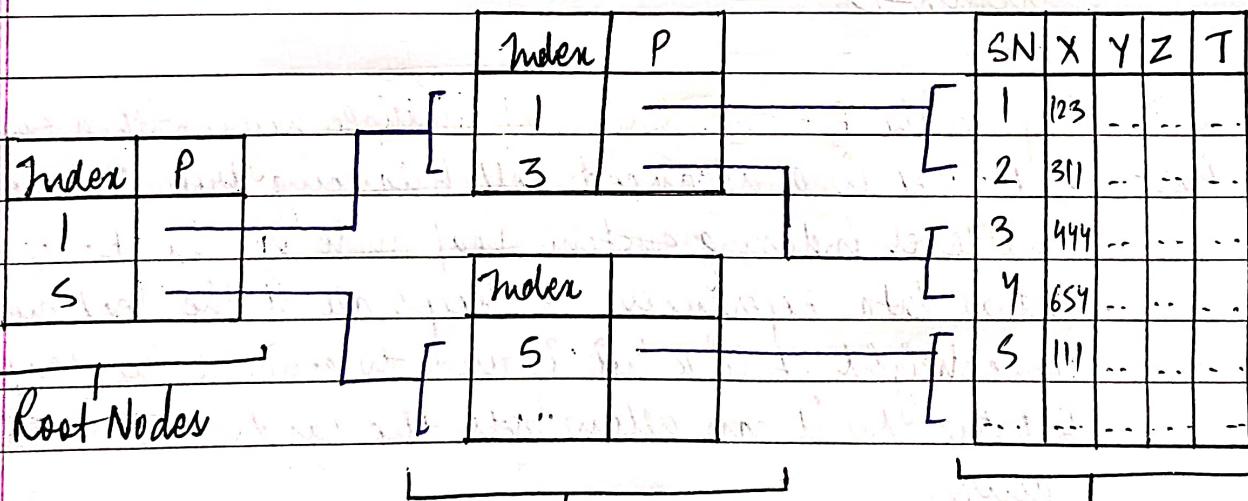
connected together in the form of singly linked list.

→ B+ trees are used to store vast amounts of data that are ~~too~~ large to fit in the main memory. The internal nodes of the B+ tree (the keys to access records) are securely stored in memory, whereas leaf nodes are placed in the secondary memory due to the restricted amount of main memory. B+ tree internal nodes are often referred to as index nodes.

Eg of B+tree of order 3 - :



→ In B+ tree the index of indices is formed in multi-level indexing as - :



Q.2) What are various issues one should consider while choosing file organisation and indexing techniques?

Ans - When choosing file organisation we should consider following factors.

- (i) Fast Retrieval
- (ii) High amount of work for processing data
- (iii) Efficient use of storage space
- (iv) Minimising need for reorganization
- (v) Protection from failures & data loss
- (vi) Accomodating growth
- (vii) Security from unauthorized use.

While choosing indexing technique it should be taken into consideration that -

- Frequency of insertion & deletion determines the hash index and hence it should be taken into consideration while choosing an indexing technique
- Data Distribution within the database, if the data is evenly distributed, a uniform indexing technique may be suitable. However if data exhibits skew or clustering, clustering indexing or multilevel indexing may be more appropriate

Section D :-

Q.1) How does Dynamic Hashing manage file expansion? What are the problems with static hashing?

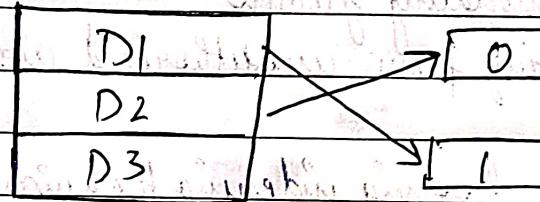
Ans - The drawback of static hashing is that it does not expand or shrink dynamically as the size of the database grows or shrinks.

In Dynamic hashing, data buckets grow or shrink (added or removed dynamically) as the records increases or

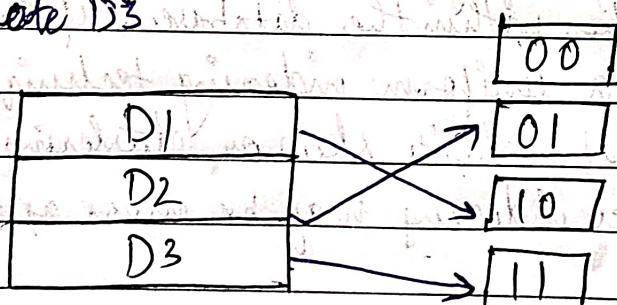
decrease and hence is also called extended hashing. In dynamic hashing, the hash function is made to produce large no. of values.

For eg - there are 3 data records D_1 , D_2 and D_3 . The hash function generates 3 addresses - 1001, 0101 and 1010 resp. This method of storing considers only part of this address - especially only first one bit to store the data. So it tries to load three of them at address 0 and 1.

$$\begin{aligned} h(D_1) &\rightarrow 1001 \\ h(D_2) &\rightarrow 0101 \\ h(D_3) &\rightarrow 1010 \end{aligned}$$



But the problem is that no bucket address remains for D_3 . The bucket has to grow dynamically to accommodate D_3 . So it changes the address to have 2 bits rather than 1 bit, & it updates the existing data to have 2 bit address. Then it tries to accommodate D_3 .



So,

- In ~~dynamic~~ dynamic hashing the performance doesn't decrease as the data grows in the system. It simply increase the size of memory to accomodate the data.
- As the memory is well utilized (it grows & shrink with data) there will not be any unused memory lying.

Q-2.) Describe the various components of data warehouse & expl. different data model used to store data with eg.

- Aus- Components of datawarehouse includes -
- 1.) Operational Source - it is a data source consisting of operational and external data.
 - 2.) Load Manager - it performs all operations associated with the extraction of loading data in the data warehouse.
 - 3.) Warehouse Manager - it is responsible for warehouse management processes like analysis, aggregation, backup & collection of data, de-normalization of data.
 - 4.) Query Manager - it performs all tasks associated with the management of user queries.
 - 5.) Detailed Data - it is used to store all the detailed data in database schema. Detailed data is loaded into data warehouse to complement the data collected.
 - 6.) Summarized Data - it is part of data warehouse that stores predefined aggregations generated by warehouse manager.
 - 7.) Archive & Backup data - the detailed data in the database schema. Data is relocated to storage archives such as magnetic tapes or optical disks.
 - 8.) Metadata - it is used for extraction & loading process, warehouse management process and query management process.
 - 9.) End-User Access Tool - it consist of analysis, reporting & mining. By using these tool, user can links with warehouse.

→ Diff. data models used to store data are -

- 1.) Relational data model - it designs the data in the form of row & columns within the table. Thus this model uses tables for representing data & in-between relationships.
- 2.) Entity-relationship model - It is the logical representation

of data as objects & relationships among them. Here objects are known as entities & relationship is association among these entities.

- 3.) Object Based Data Model - It is an extension of ER Model with notions of functions, encapsulations & object identity as well. This model supports a rich type system that includes structured & collection types.
- 4.) Semistructured data model - It allows the data specifications at places where the individual data items of the same type may have diff. attribute sets. The XML is widely used for representing the semistructured data.