Database Management Systems – Tutorials

Chapter 7 Physical Data Structures

Questions

1. What operations do we need to carry out on data that is stored in a database?
2. List some different data structures that are possible for storing data on disk?
3. What are the advantages of using an index and what are its disadvantages?
4. Given the student-subject database, give a set of queries and then discuss what data structure might be the best if the database is quite large.
5. What is an indexed-sequential file?
6. What is a B-tree? How does the B^+-tree defer from a B-tree?
7. Which of the two structures is preferred as an access structure in database management and why?
8. How do the dynamic hashing schemes differ from the classical hashing schemes?
9. Why is the classical technique not suitable for storing information on disk?
10. Discuss extendible and linear hashing. How can they be used for constructing an index?

Answers

1. What operations do we need to carry out on data that is stored in a database?

Some of the common operations on data are:

- insertion
- retrieval
- deletion
- update
- report generation

2. List some different data structures that are possible for storing data on disk?

Data may be stored as either an unstructured file or an unordered file. More efficient storage techniques include ordered files, indexed sequential files, B-tree files, and dynamic hashing files.

3. What are the advantages of using an index and what are its disadvantages?

Index assists in rapid retrieval of records. The disadvantage of an index is that it requires additional storage. The additional storage grows as the file grows. The additional storage becomes very large if there are a number of indexes on the file.

4. Given the student-subject database, give a set of queries and then discuss what data structure might be the best if the database is quite large.

Some common queries on a student database are likely to be:

- Give an ordered list of students – a sequential or indexed sequential file on the student name would be the best
- Produce a class list for each subject – two indexes might be preferred but an indexed sequential file will perform satisfactorily
- Give a list of students in each subject that have got the grade A or failed – may need to use a sequential file unless an index has been built on the grade
- Give a list of students that are doing more than 4 subjects – sequential file may be the best
- Produce a list of students who have a GPA of 1.5 or below – sequential file may be the best

5. What is an indexed-sequential file?

An indexed-sequential file is an ordered file on which an index has been built.

6. What is a B-tree? How does the B^+-tree defer from a B-tree?

A B-tree of order m is a multiway search tree such that:
(a) Every node except the root has at most (m–1) keys and at least [m/2] – 1 keys
(b) The root may have as few as one key
(c) Every node with k keys has k+1 descendants except the leaves
(d) All leaf nodes are on the same level

B^+-tree defers from a B-tree in that every key in the non-leaf nodes is replicated in the leaf nodes and the leaf nodes are linked together to form a sequential file.

7. Which of the two structures is preferred as an access structure in database management and why?

Two methods are the B-tree and dynamic hashing. B-tree has the advantage that it may be used as an indexed sequential file which is very fast for range queries. Hashing leads to very quick retrieval when the hashing key value is given but range queries are very slow.

8. How do the dynamic hashing schemes differ from the classical hashing schemes?

The classical hashing assumes that the size of the file is fixed and known in advance. If the file is close to being full the performance declines quickly. The dynamic hashing files assume that files are dynamic and therefore grow and shrink. Dynamic hashing schemes allow such growth or shrinkage gracefully allowing consistently good performance whatever the size of the file.

9. **Why is the classical technique not suitable for storing information on disk?**

   As noted above, the classical hashing assumes that the size of the file is fixed and known in advance. Since files in a database grow and shrink with time, the performance of a file using classical hashing may decline.

10. **Discuss extendible and linear hashing. How can they be used for constructing an index?**

    It is possible to categorize dynamic hashing methods into two classes. The first class consists of those methods which use a directory structure; the second class of methods do not use a directory. The best known methods based on directory structures are variations of a scheme called *extendible hashing*. The most well known methods that do not use a directory are based on a dynamic hashing scheme called *linear hashing*. Both methods can be used as an index for retrieval if the value of the key on which the file is hashed is known.

Additional Questions (without any answers)

1. Explain some of the basic processes needed in memory buffer management
2. List the various factors that determine the efficiency of a storage structure
3. Describe how data is stored on a magnetic disk
4. Describe the different types of index types
5. Explain the basic features of an indexed-sequential file
6. Analyse the performance of B-tree
7. Explain the concept and uses of inverted files