Total Questions : - 37 + 20 + 26 + 21 + 12 + 18 + 10 = 144

PARALLEL AND DISTRIBUTED DATABASES

1. In a **shared-memory** system, multiple CPUs are attached to an interconnection network and can access a common region of main memory.

2. In a **shared-disk** system, each CPU has a private memory and direct access to all disks through an interconnection network.

3. In a **shared-nothing** system, each CPU has local main memory and disk space.

4. **Interference** is a basic problem with shared-memory and shared-disk architectures.

5. The **speed-up** curves show how, for a fixed database size, more transactions can be executed per second by adding CPUs.

6. The **scale-up** curves show how adding more resources (in the form of CPUs) enables us to process larger problems.

7. The **shared-nothing** architecture has been shown to provide linear speed-up and linear scale up

8. If an operator consumes the output of a second operator, we have a **pipelined parallelism**

9. An operator is said to **block** if it produces no output until it has consumed all its inputs.

10. If data is distributed and all servers run the same DBMS software, then it is called as **homogeneous** distributed database system.

11. If different sites run under the control of different DBMSs, then it is called as **heterogeneous** distributed database system.

12. Heterogeneous distributed database system also referred as a **multidatabase** system.

13. A **Client-Server** system has one or more client processes and one or more server processes, and a client process can send a query to any one-server process.

14. In a **Collaborating Server** system we can have a collection of database servers, each capable of running transactions against local data.

15. **Fragmentation** consists of breaking a relation into smaller relations.

16. In **horizontal** fragmentation, each fragment consists of a subset of rows of the original relation.

17. In **vertical** fragmentation, each fragment consists of a subset of columns of the original relation.

18. **Replication** means that we store several copies of a relation or relation fragment.

19. A **local** name field, which is the name assigned locally at the site where the relation is created.

20. A **birth site** field, which identifies the site where the relation was created.

21. Joins in a Distributed DBMS are performed by **Fetch As Needed, Ship to One Site, Semijoins** and **Bloomjoins**

22. Copies of a modified relation are updated only periodically in **asynchronous** replication approach,

23. In **synchronous** replication; all copies of a modified relation must be updated before the modifying transaction commits.

24. In **voting**, a transaction must write a majority of copies in order to modify an object and read at least enough copies to make sure that one of the copies is current.

25. In **read-any write-all**, to read an object, a transaction can read any one copy, but to write an object, it must write all copies.

26. In Primary Site Asynchronous Replication Changes are usually propagated in two steps called **Capture and Apply**

27. Create a copy of all the data sources at one location and use the copy rather than going to the individual sources such a copied collection of data is called a **data warehouse**

28. A single site is in charge of handling lock and unlock requests for all objects in **Centralized** Lock management

29. In **Fully distributed** Lock management; requests to lock or unlock a copy of an object stored at a site are handled by the lock manager at the site where the copy is stored.

30. Dead lock detection algorithm identifies `deadlocks’ due to delays in propagating local information that do not really exist then such deadlocks are called **phantom**

31. In 2PC the transaction manager at the site where the transaction originated is called the **coordinator** for the transaction.

32. In 2PC the transaction managers at sites where its sub transactions execute are called **subordinates**

33. In a **Collaborating Server** system, there is no distinction between client and server processes.

34. In a **Middleware** system, a special server allows coordination of queries across multiple databases.

35. If a relation is fragmented and replicated, each partition needs a globally unique Name called the **relation** name.

36. **Distributed catalog** management is needed to keep track of what is stored where.

37. **Semijoins** and **Bloomjoins** reduce the number of tuples sent across the network by first sending information that allows us to filter irrelevant tuples.

Internet Databases

1. XML is an emerging document description standard that allows us to describe the **content** and **structure** of a document in addition to giving display directives.

2. XML documents have less rigid structure than a relational database and are said to be **semi structured .**

3. Two broad classes of search are boolean queries and **ranked queries**.

4. **Ranked** queries ask for documents that are most relevant to a given list of keywords;

5. **precision** is the percentage of retrieved documents that are relevant to the query

6. **recall** the percentage of relevant documents in the database that are retrieved

7. **Inverted files** and **signature** files are two indexing techniques that support Boolean queries.

8. **Signature** files address the space problem associated with inverted files but must be sequentially scanned.

9. The **HITS** algorithm uses a combination of Boolean queries and analysis of links to a page from other Web sites to evaluate ranked queries.

10. An application server has **pre-forked** threads or processes and thus avoids the startup cost of creating a new process for each request.

11. A **DTD** is a set of rules that allows us to specify our own set of elements, attributes, and entities.

12. In **object exchange model (OEM)** each object is described by a triple consisting of a label, a type, and the value of the object.

13. An **inverted file** file is an index structure that enables fast retrieval of all documents that contain a query term.

14. A **signature** file contains an index record for each document in the database this index record is called the signature of the document.

15. Each signature has a fixed size of b bits; b is called the signature **width**

16. A document whose signature matches the query signature but that does not contain all terms in the query is called a **false positive**

17. To reduce the amount of data that has to be retrieved for each query, we can vertically partition a signature file into a set of **bit slices**, and such an index is called bit-sliced signature file

18. An **authority** is a page that is very relevant to a certain topic and that is recognized by other pages as authoritative on the subject.

19. **Authority** pages, usually have a significant number of hyperlinks to authority pages

DECISION SUPPORT

1. Arrange in correct orders of Creating and Maintaining a Warehouse
   EXTRACT
   CLEAN
   TRANSFORM
   LOAD

REFRESH
PURGE

2. The system catalogs associated with a warehouse are very large and are often stored and managed in a separate database called a **metadata repository**

3. OLAP systems that use arrays to store multidimensional datasets are called **multidimensional OLAP (MOLAP)** systems.

4. The relation, who relates the dimensions to the measure of interest, is called the **fact table**.

5. The relations who are much smaller than the fact table in a typical OLAP application; are called the **dimension** tables.

6. OLAP systems that store all information, including fact tables, as relations are called **relational OLAP (ROLAP)** systems.

7. If we are given total sales per city, we can aggregate on the Location dimension to obtain sales per state. This operation is called **roll-up** in the OLAP

8. If total sales by state are given, we can ask for a more detailed presentation by exploring it in detail is called as **drilling down** on Location.

9. The result of pivoting is called a **cross-tabulation**

10. **slicing** a dataset amounts to an equality selection on one or more dimensions.

11. **Dicing** a dataset amounts to a range selection.

12. A combination of a fact table and dimension tables is called a **star schema**

13. Columns with few possible values are called **sparse**

14. The collection of bit vectors for a column is called a **bitmap index** for that column.

15. A data warehouse is just a collection of **asynchronously** replicated tables and periodically maintained views.

16. When a query is posed on the view, the (unmodified) query is executed directly on the precomputed result. This approach is called **view materialization**

17. A materialized view is said to be **refreshed** when we make it consistent with changes to its underlying tables.

18. A view can be refreshed within the same transaction that updates the underlying tables. This is called **immediate view maintenance**

19. Updates are captured in a log and applied subsequently to the materialized views. This is called **deferred view maintenance**

20. Materialized views that are refreshed periodically are also called **snapshots**

21. As the computation progresses, the answer quality is continually refined. This approach is called **online aggregation**

22. An algorithm is said to **block** if it does not produce output tuples until it has consumed all of its input tuples.

23. Index structures that are especially suitable for OLAP systems include **bitmap indexes** and join indexes.

24. In **immediate view** maintenance the view is updated within the same transaction that modifies the underlying tables;

25. In **forced maintenance** we refresh the view after a certain number of changes have been made to the base tables.

26. In **top N** queries we only want to retrieve the first N rows of the query result.

DATA MINING

1. **Data mining** consists of finding interesting trends or patterns in large datasets, in order to guide decisions about future activities.

2. An algorithm is **scalable** if the running time grows (linearly) in proportion to the dataset size.

3. The **knowledge discovery process** can roughly be separated into four steps.
   Data selection
   Data cleaning
   Data mining
   Evaluation

4. The **support** of an itemset is the fraction of transactions in the database that contain all the items in the itemset.

5. All itemsets whose support is higher than a userspecified minimum support called **minsup**

6. **The a priori property**, Every subset of a frequent itemset must also be a frequent itemset.

7. The number of data groups is very large, but the answer to the query is usually very small, we call such a query an **iceberg query**.

8. The **support** for a set of items is the percentage of transactions that contain all of these items.

9. The **Confidence** for a rule *LHS* ⬄ *RHS* is the percentage of such transactions that also contain all items in *RHS*

10. In market basket analysis.If we use the *date fi*eld as grouping attribute then it is called as **calendric** market basket analysis.

11. A **subsequence** of a sequence of itemsets is obtained by deleting one or more itemsets, and is also a sequence of itemsets.

12. If the dependent attribute is categorical, we call such rules **classification rules**.

13. If the dependent attribute is numerical, we call such rules **regression rules**.

14. Trees that represent classification rules are called **classification trees** or **decision trees**

15. Trees that represent regression rules are called **regression** **trees**.

16. Each internal node of the decision tree is labeled with a predictor attribute. This attribute is often called a **splitting** attribute.

17. In the process of decision tree making, in the **pruning phase**, the final size of the tree is determined.

18. Similarity between cluster records is measured computationally by a **distance** function.

19. The output of a clustering algorithm consists of a **summarized** representation of each cluster.

20. A **partitional** clustering algorithm partitions the data into *k* groups such that some criterion that evaluates the clustering quality is optimized.

21. In **hierarchical** clustering algorithm we merges two partitions in each step until only one single partition remains in the end.

SPATIAL DATA MANAGEMENT

1. A spatial data object occupies a certain region of space, called its spatial extent, which is characterized by its **location** and **boundary**.

2. A **Point** data has a spatial extent characterized completely by its location;

3. A **region data** has a spatial extent with a location and a boundary.

4. In **Z-value 0111** space filling curve the value of point is obtained by interleaving the bits of the *X* and *Y* values; we take the first *X* bit (0), then the first *Y* bit (1), then the second *X* bit (1), and finally the second *Y* bit (1).

5. **Spatial range** *queries* specify a query region and aim to retrieve all objects that fall within or overlap the query region.

6. **Nearest neighbor** *queries* specify a query point and aim to retrieve the object closest to the query point.

7. **Spatial join** *queries* compute all pairs of objects that satisfy user-specified proximity constraints.

8. A **multidimensional** or *spatial* index utilizes spatial relationships between data objects to organize the index.

9. We can use the recursive nature of space-filling curves to recursively partition the space; this is done in the **Region Quad** *tree* index structure.

10. A **Grid file** is a spatial index structure for point data. Each dimension is partitioned into intervals that are maintained in an array called a *linear scale*.

11. **R trees** are height-balanced tree index structures whose search keys are *bounding boxes*.

12. The **generalized search tree (GiST)** is a generic index template for tree-structured indexes.

DEDUCTIVE DATABASES

1. A Datalog program consists of a collection of rules. A rule consists of a **head** and a **body**

2. DBMSs that support Datalog are called **deductive database** systems since the rules are applied iteratively to deduce new tuples.

3. The meaning of a Datalog program can be defined either through least model semantics or through least **fixpoint** semantics.

4. A **model** of a program is a collection of relations that is consistent with the input relations and the Datalog program.

5. A model that is contained in every other model is called a **least** **model.**

6. The least fixpoint is a fixpoint that is smaller than every other **fixpoint**

7. If we consider Datalog programs without **negation** every program has a least fixpoint and the least fixpoint is equal to the least model.

8. We say that a table T depends on a table S if some rule with **T in the head contains S**, or (recursively) contains a predicate that depends on S, in the body.

9. If a Datalog program contains not, it can have more than one least **fixpoint**

10. In a stratified program, the relations can be classified into numbered layers called **strata**

11. Straightforward evaluation of recursive queries by repeatedly applying the rules leads to **repeated inferences**

12. Simple repeated application of the rules to all tuples in each iteration is also called **Naive** fixpoint evaluation.

13. We can avoid repeated inferences using **Seminaive** fixpoint evaluation.

14. **Seminaive** fixpoint evaluation only applies the rules to tuples that were newly generated in the previous iteration.

15. To avoid unnecessary inferences, we can add filter relations and modify the Datalog program according to the **Magic Sets** program-rewriting algorithm.

```
16.SameLevel(S1, S2) :- Magic SameLevel(S1), Assembly(P1, S1, Q1),
                        SameLevel(P1, P2), Assembly(P2, S2, Q2).

17.SameLevel(S1, S2) :- Assembly(P1, S1, Q1),
                        SameLevel(P1, P2), Assembly(P2, S2, Q2).

18.Components(Part, Subpart) :- Assembly(Part, Part2, Qty),
                                Components(Part2, Subpart).
```