Next Page

Question 1 (5 points)

What are the 5 index evaluation metrics.

-access type supported efficiently
-access time
-insertion time
-deletion time
-space overhead

Next Page

Page 1 of 19

Next Page

Question 2 (1 point) ✓ *Saved*

What are the two basic types of indices.

ordered indices

hash indices

Next Page

Page 2 of 19

Next Page

Question 3 (1 point)

A search-key can only contain one attribute.

- ○ True
- ◉ False

Next Page

Page 3 of 19

Next Page

Question 4 (1 point) *Saving...*

An index will typically improve which of the following?

- ○ Access time
- ○ Insertion time
- ○ Deletion time
- ◉ All of these

Next Page

Page 4 of 19

Next Page

Page 5 of 19

Question 5 (1 point) ✓ *Saved*

write a SQL statement to create an index named foo on attribute baz in relation bar

`E INDEX foo ON bar (baz)`

CREATE INDEX foo ON bar(baz)

Next Page

Page 5 of 19

Next Page

Page 6 of 19

Question 6 (1 point) ✓ *Saved*

This is the process off adding a large number of entries to a B+Tree at one time.

- (•) bulk loading
- ( ) batch loading
- ( ) group loading
- ( ) mass loading

Next Page

Page 6 of 19

Question 7 (1 point) ✓ *Saved*

What type of index is used to prevent (some) queries having to fetch the actual records

covering indices

Next Page

Page 7 of 19

Question 8 (1 point) *Saving...*

A secondary index must be dense

- (•) True
- ( ) False

Next Page

Question 9 (1 point)

If an index does not fit in memory, we often create what, which contains an outer index in memory and a inner index on disk?

multi-level index

Next Page

Question 11 (1 point)

This type of index requires every search-key to appear.

- ( ) complete
- ( ) total
- (•) dense
- ( ) spare

Next Page

Question 12 (1 point) ✓ *Saved*

This type of index can be used to handle larger indices that do not fit into memory.

- ○ dense
- ○ hashed
- ◉ multilevel
- ○ value

Next Page

Question 13 (1 point) ✓ *Saved*

A secondary index is also known as a clustering index.

- ○ True
- ◉ False

Next Page

Question 14 (1 point) ✓ *Saved*

An index file contains index entries that are search-key, pointer pairs.

- ◉ True
- ○ False

Next Page

Question 15 (3 points) ✓ *Saved*

Explain the process of finding a value in a B+Tree.

It would be similar to BFS. We check if the value smaller or greater than the root node first. If it smaller, we continue checking the left and disregard the right. If it is greater, we continue checking the right and disregard the left. We keep on checking our child nodes and repeat the same step until we found our value.

Next Page

Question 16 (2 points) ✓ *Saved*

Explain how a B-Tree is different from a B+-Tree.

- In B tree, all of the keys and record are stroed in internal and leaf node. While in B+ tree, keys are indexes stored in inernal nodes and records stored in the leaf node.
- B tree cannot have duplicated keys while B+ tree can.
- B-tree have greater depth than B+ tree.

Next Page

Question 17 (1 point) ✓ *Saved*

In a B+Tree a leaf node that is not the root must have:

- ○ ceiling((N-1)/2) and N values
- ○ ceiling(N/2) and N values
- ◉ ceiling((N-1)/2) and N-1 values
- ○ ceiling((N)/2) and N-1 values

Next Page

Question 18 (1 point)

What type of index is used when we want to efficiently query multiple keys with an array of bits

bitmap indices

Next Page

Question 19 (1 point) ✓ *Saved*

In a B+Tree a internal node that is not a leaf or the root must have:

- ○ between ceiling((N)/2) and N-1 children
- ○ between ceiling((N-1)/2) and N children
- ◉ between ceiling(N/2) and N children
- ○ between ceiling((N-1)/2) and N-1 children

Next Page