

B+ trees:

MCQs:

In a B+ Tree, where are the actual records (data pointers) stored?

- a) In internal nodes only
- b) In leaf nodes only
- c) In both internal and leaf nodes
- d) In the root node only

What is the maximum number of keys a B+ Tree of order m can have in a single node?

- a) $m - 1$
- b) m
- c) $2m$
- d) $2m - 1$

The time complexity of search, insert, and delete operations in a B+ Tree with n keys is:

- a) $O(1)$
- b) $O(\log n)$
- c) $O(n)$
- d) $O(n \log n)$

When inserting into a B+ Tree, if a leaf node overflows, what typically happens?

- a) The tree is rebuilt from scratch
- b) The node is split and the middle key is promoted to the parent
- c) The entire level is restructured
- d) The overflow is ignored

In a B+ Tree of order m , what is the maximum number of children an internal node can have?

- a) m
- b) $m - 1$
- c) $2m$
- d) $2m - 1$

If a B+ Tree has height h and stores n keys, what is the worst-case time complexity of searching for a key?

- a) $O(1)$
- b) $O(\log n)$
- c) $O(h)$
- d) $O(n)$

Which of the following best describes the height h of a B+ Tree with n keys and order m ?

- a) $h = O(n)$
- b) $h = O(\log n)$ with base m
- c) $h = O(\log_m n)$
- d) $h = O(\log_2 n)$

During deletion from a B+ Tree, if a leaf node has fewer than the minimum number of keys:

- a) The tree is rebuilt from scratch
- b) Keys are borrowed from a sibling or nodes are merged
- c) The underflow is ignored
- d) The root is deleted automatically

Suppose a B+ Tree of order m has n keys. What is the maximum possible number of leaf nodes?

- a) n
- b) n/m
- c) $n/(m - 1)$
- d) $\log_m(n)$

Which of the following operations might cause merging of nodes in a B+ Tree?

- a) Search
- b) Insert
- c) Delete
- d) Range Query

In a B+ Tree of order m , the maximum number of keys in an internal node is:

- a) m
- b) $m - 1$
- c) $2m$
- d) $2m - 1$

In a B+ Tree of order m , the minimum number of keys in a non-root internal node is:

- a) $\lceil m/2 \rceil - 1$
- b) $\lfloor m/2 \rfloor$
- c) $m/2$
- d) $\lceil m/2 \rceil$

If the order of a B+ Tree is doubled, how does the height change approximately (for large n)?

- a) It doubles
- b) It halves
- c) It remains unchanged
- d) It increases slightly but not doubled

Short Question:

Given a max degree of 5, construct a B+ after the following operations in order:

Insert([7, 10, 1, 23, 5, 15, 17, 9, 11, 39, 35, 8, 40, 25])

Delete(9)

Delete(7)