B-Tree Search Complexity Proof

Shyamal Suhana Chandra

1 Theorem: B-Tree Search Complexity

Statement: Searching in a B-Tree with n keys takes $O(\log n)$ time.

2 Proof

1. Search traverses from root to leaf: $O(\log_t n)$ levels
2. At each level, perform binary search on at most $(2t - 1)$ keys: $O(t) = O(1)$ if t is constant
3. Number of levels: $O(\log_t n) = O(\log n)$
4. Total time: $O(\log n) \times O(1) = O(\log n)$

Conclusion: B-Tree search has $O(\log n)$ time complexity.