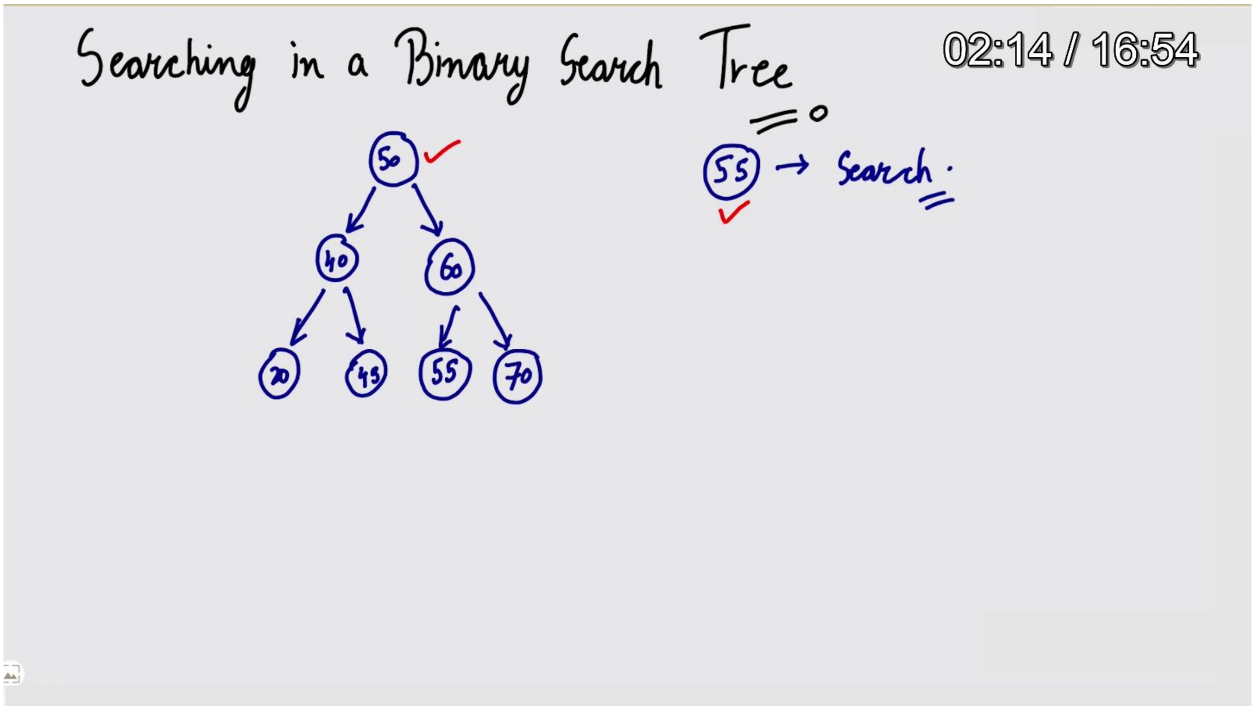
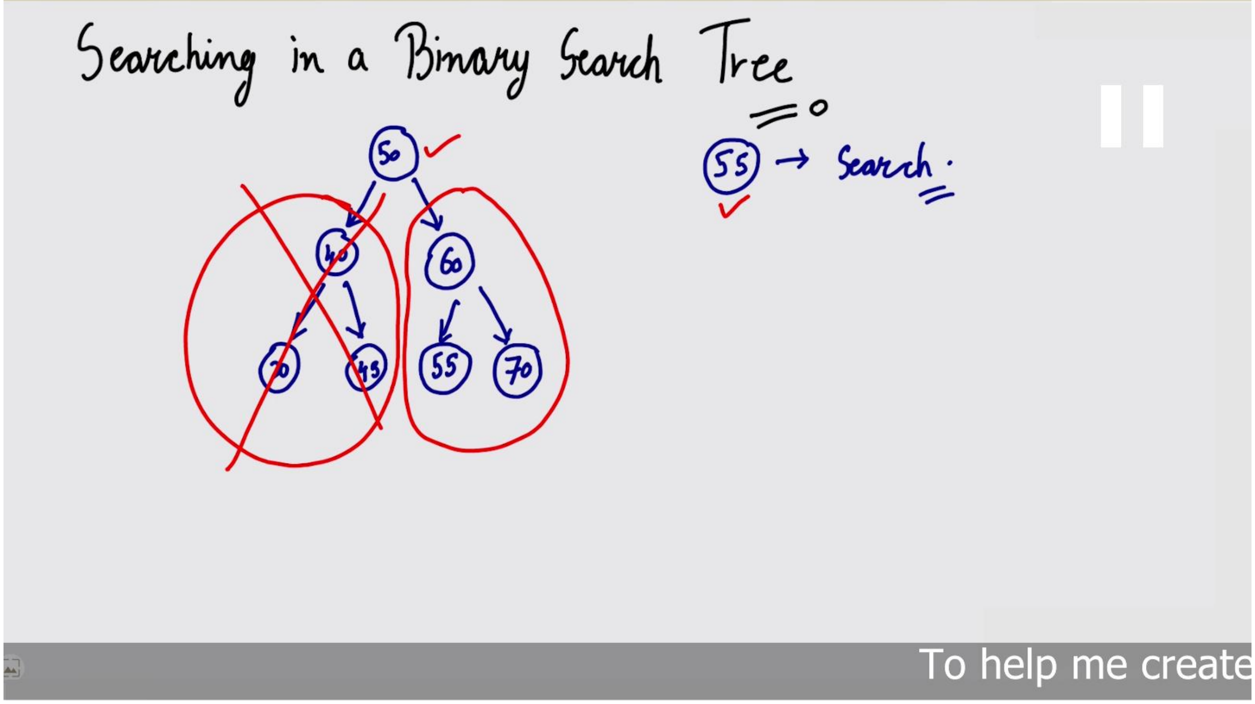


Searching in a Binary Search Tree

02:14 / 16:54

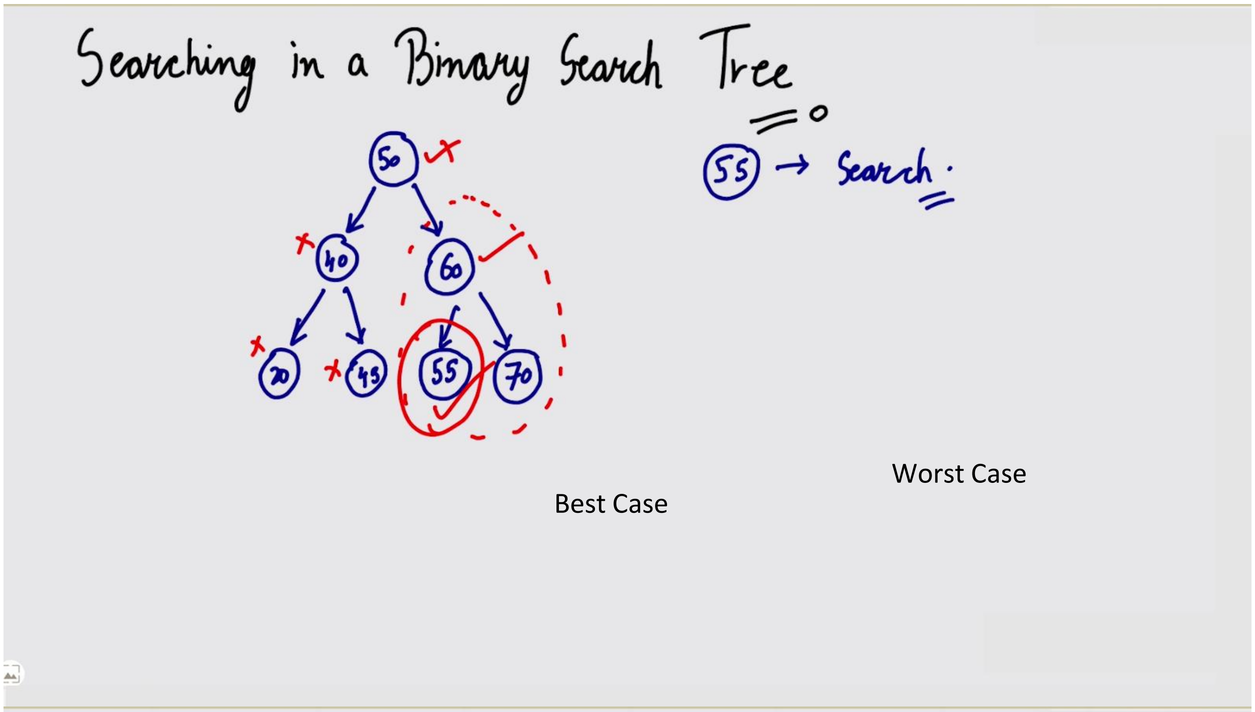
55 → Search:



Searching in a Binary Search Tree

55 → Search:

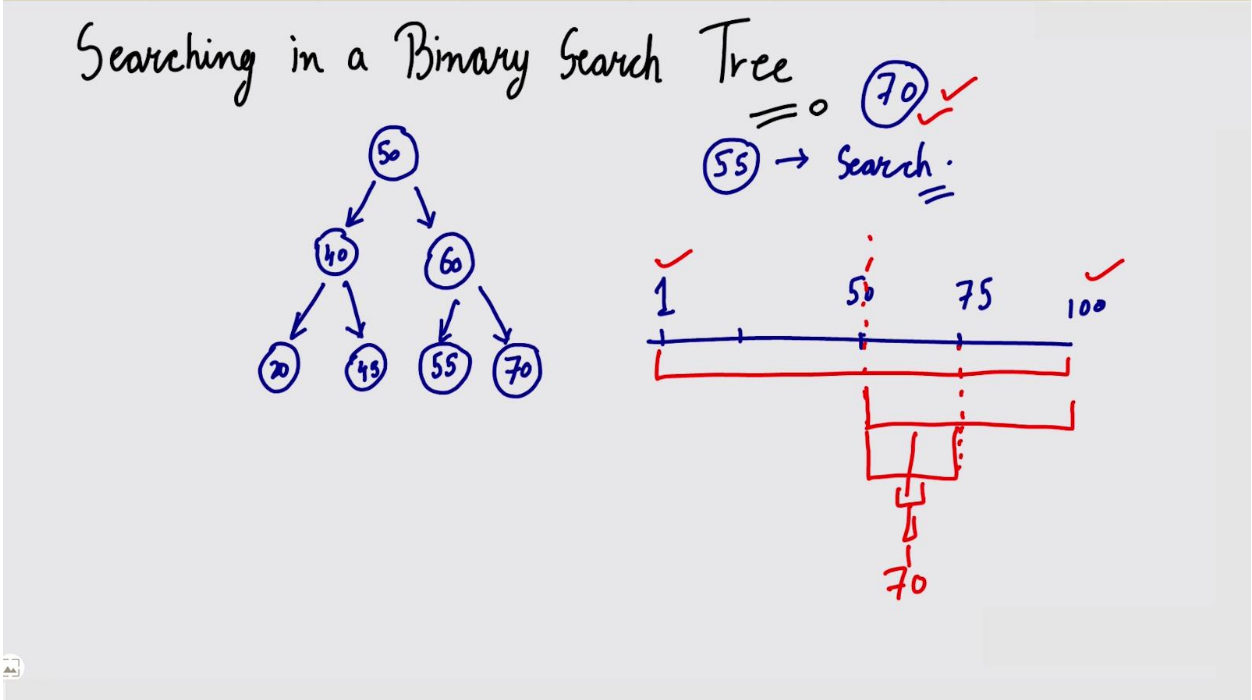
To help me create



Searching in a Binary Search Tree

Worst Case

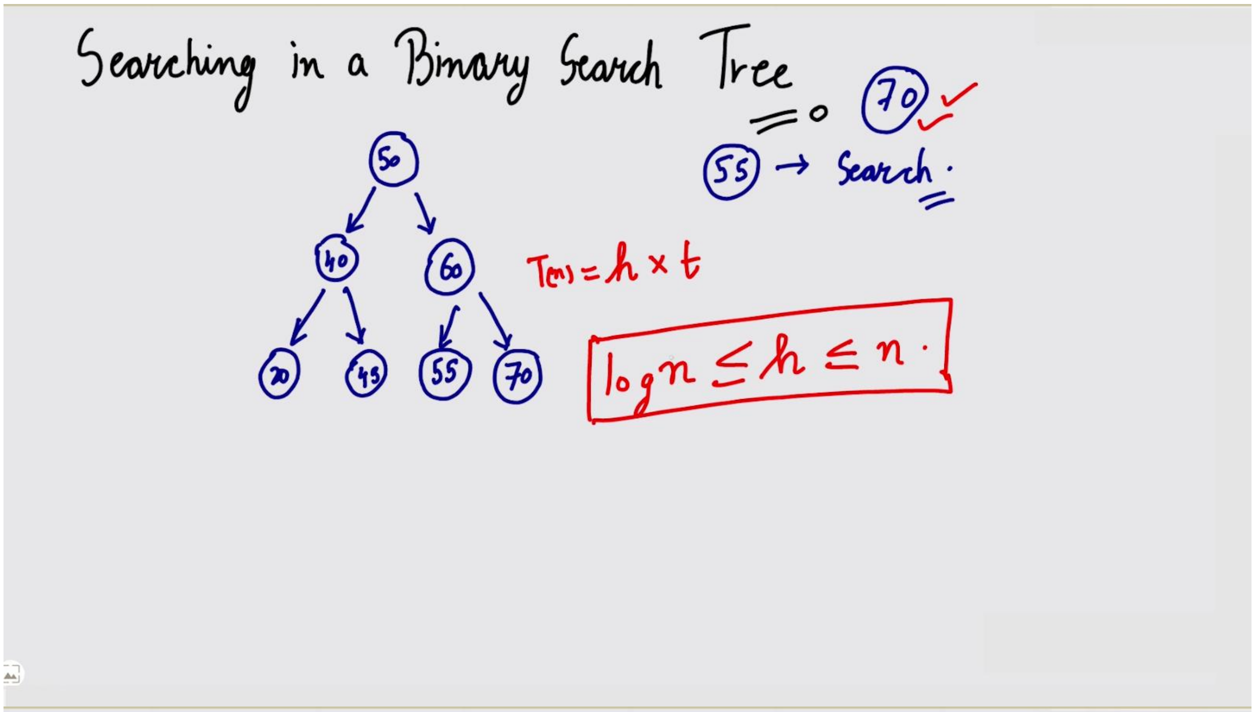
Best Case



Searching in a Binary Search Tree

Time Complexity = $\mathrm{O}(\log \mathrm{n})$ → Best case

Time Complexity = $O(n) \rightarrow Worst\ Case$



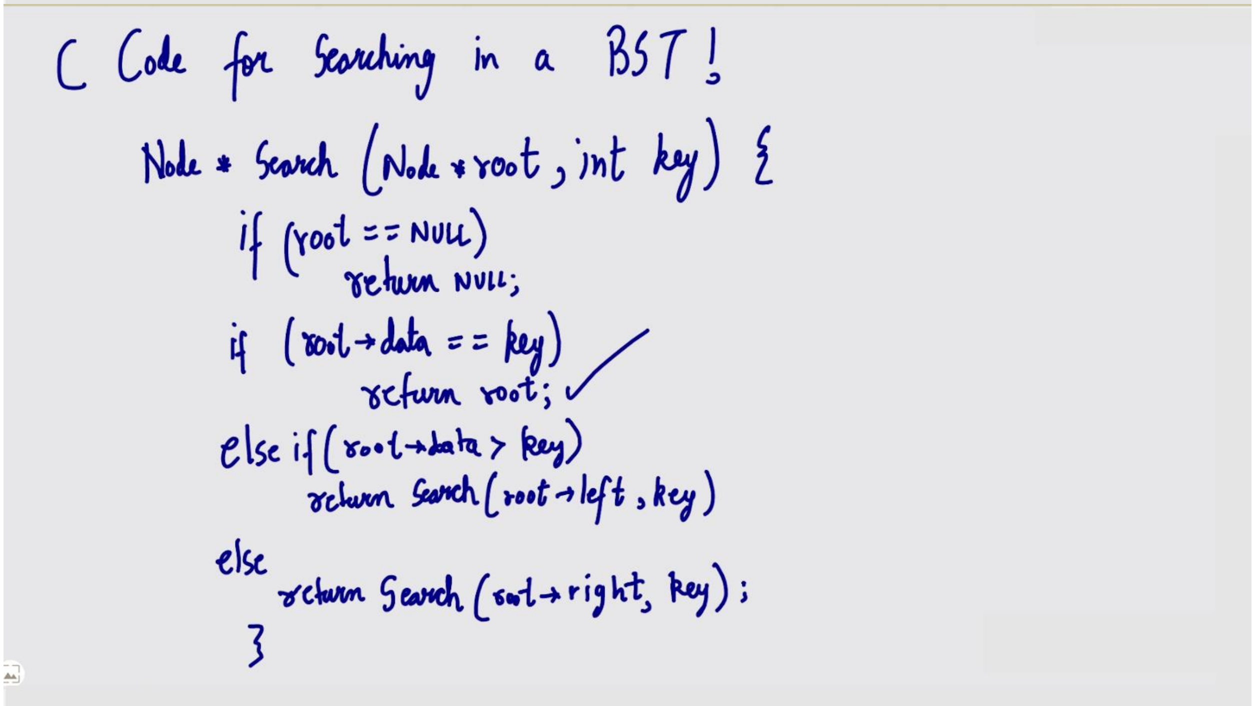
Searching in a Binary Search Tree

= 0 70 ✓ ✓

55 → Search.

$T(n) = h \times t$

$\boxed{\log n \le h \le n.}$



C Code for Searching in a BST!

```
Node * Search (Node *root, int key) {
      if (root == NULL)
            return NULL;
      if (root->data == key)
              return root; ✓
      else if (root->data > key)
            return Search(root->left, key)
      else
         return Search (root->right, key);
      }
```