- C
- C++
- Java
- Design
- CORE CS
  - Operating Systems
  - TCP/IP
  - DBMS
  - working
  - ile
  - Communication

Given a pointer to the node in a binary tree, write code to find its inorder successor? You may assume each node to also have a pointer to the parent node (along with pointer to right and left sub-trees). Hence, the structure of Node will be

```
struct Node
{
    int data;
    Node* lptr;    // pointer to the left subtree
    Node* rptr;    // pointer to the right subtree
    Node* parent;  // Pointer to the parent Node (null for root of
tree)
};
```

For example: if we have the below binary tree, then

TAG CLOUD

algorithm A
Array; Sort
Search Tr
Binary Tree

- Logic
  - Quantitative Aptitude
  - Puzzles
- Interview Tips
- Videos
- Articles

RECENT POSTS

Logger Rate Limiter Solution (LeetCode)

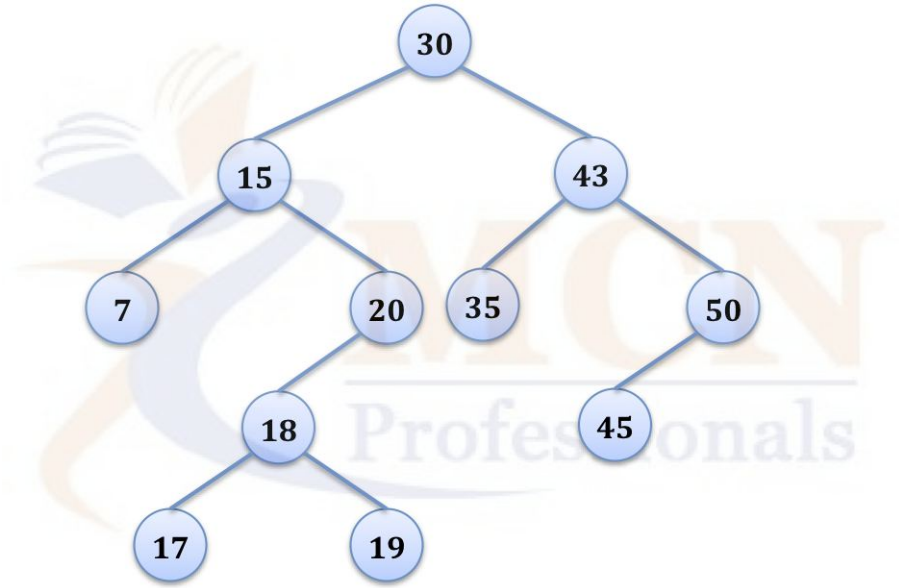
Fitting Shelves Problem

Assign Mice to Holes

Greedy Algorithm for Egyptian Fraction

Greedy Solution to Activity n.

/ith given




```
Node         Inorder Successor
-------        ---------------------
15               17
18               19
50              NULL
20              30
7                15
```

Solution:

If the Node has right child, then the in-order successor of the node ca be found by

- Move to the right child
- keep moving to the left child untill the left child becomes NULL. When we get a node for which the left child is null, return that node.

Bit twidling C++11 Design Pattern Divide-n-Conquer Dynamic Programming function Greedy Approach Greedy Programming Hash hashing Heap Helper Function Interview Puzzle IP Address Linked List Logic main Matrix Merge Number Series Operator Output Palindrome Permutation pointer Polish Notation Puzzle Quant Queue Recursion Reverse Searching sizeof Sliding Window Sorting Stack String TimeComplexity Traversal Trie

Max Distance between two occurrences of the same element

Swapping two variables without using third variable

Count max points on a line

Print all Subsequences of an Array

```
For example:
1. To find the in-order successor of 18
   - Move to the right child, i.e 19
   - Since the left child of 19 is NULL, return 19
2. To find the in-order successor of 15
   - Move to the Right child, i.e 20
   - Move to the left child, i.e 18 and keep moving to left child
till we get a NULL pointer as left child, i.e 17. Return 17.
```

If the Right child is NULL, then in-order successor can be found using the parent Node,

- Move to the parent Node, untill the Node becomes the left child of the parent.
- If Parent becomes NULL then return NULL, else return the parent

```
For Example:
1. In-order successor of 35 is 40 because 35 is itself the left
child
2. In-order successor of 20 will be 30.
   - Parent of 20 is 15, but 20 is the right child of 15 so move
to the parent (i.e 15).
   - Parent of 15 is 30 and 15 is the left child of 30 so return
30.
3. In-order successor of 50 will be NULL.
   - Parent of 50 is 43 and 50 is the right child of 43. So move
to the parent
   - Parent of 43 is 30 and 43 is also the right child of 30. So
move to the parent
   - Parent of 30 is NULL, so return NULL. (i.e 50, does not have
an in-order successor)
```

Code:

```
// Returns the in-order successor of node pointed to by d
Node* successor(Node * d)
{
   if(d == NULL)
       return NULL;
   // If d has a Right Child
   if(d->rptr != NULL)
   {
      // Move to Right Node
      d = d->rptr;
      // Move to the extreme left
      while(d->lptr != NULL)
         d = d->lptr;
      return d;
   }
   while(d)
   {
      Node* p = d->parent;
      if(p == NULL)
         break;
      if(p->lptr == d)
         return p;
      else
         d = p;
   }
   return NULL;
}
```

Let me know your Comments / Feedback / suggestions.

———————————————————————————————————-

Related Posts:

- How to compute the time and space complexity of...
- Tricks to compute the time and space complexities....

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment

Name * **Email *** **Website**

Save my name, email, and website in this browser for the next time I comment.

Post Comment

If you have any question, feel free to call us

+91 837 780 3450

krawat@ritambhara.in

USEFUL LINKS

About us
Articles
Contact Us
Online Courses

CONTACT ADDRESS

310, Neelkanth Plaza,
Alpha-1 (Commercial), Greater Noida
U.P (INDIA)