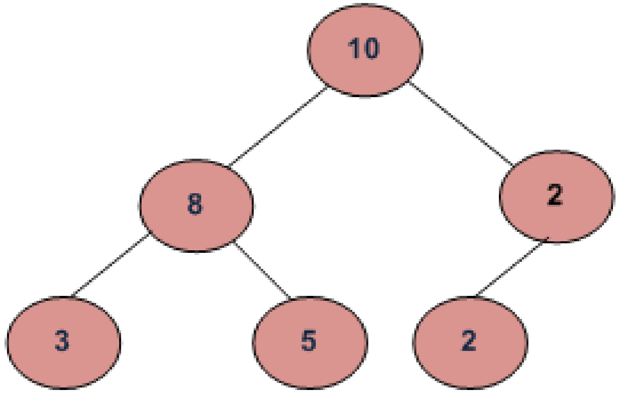
Root to leaf path sum equal to a given number

Given a binary tree and a number, return true if the tree has a root-to-leaf path such that adding up all the values along the path equals the given number. Return false if no such path can be found.



For example, in the above tree root to leaf paths exist with following sums.

21 –> 10 – 8 – 3
23 –> 10 – 8 – 5
14 –> 10 – 2 – 2

So the returned value should be true only for numbers 21, 23 and 14. For any other number, returned value should be false.

Algorithm:
Recursively check if left or right child has path sum equal to (number – value at current node)

Implementation:

Interview Experiences
Advanced Data Structures
Dynamic Programming
Greedy Algorithms
Backtracking
Pattern Searching
Divide & Conquer
Mathematical Algorithms
Recursion
Geometric Algorithms

```
#include<stdio.h>
#include<stdlib.h>
#define bool int

/* A binary tree node has data, pointer to left child
   and a pointer to right child */
struct node
{
   int data;
   struct node* left;
   struct node* right;
};

/*
 Given a tree and a sum, return true if there is a path from the root
 down to a leaf, such that adding up all the values along the path
 equals the given sum.

 Strategy: subtract the node value from the sum when recurring down,
 and check to see if the sum is 0 when you run out of tree.
*/
bool hasPathSum(struct node* node, int sum)
{
  /* return true if we run out of tree and sum==0 */
  if (node == NULL)
  {
     return (sum == 0);
  }

  else
  {
    bool ans = 0;

    /* otherwise check both subtrees */
    int subSum = sum - node->data;

    /* If we reach a leaf node and sum becomes 0 then return true*/
    if ( subSum == 0 && node->left == NULL && node->right == NULL )
      return 1;

    if(node->left)
      ans = ans || hasPathSum(node->left, subSum);
    if(node->right)
      ans = ans || hasPathSum(node->right, subSum);

    return ans;
```

Popular Posts

All permutations of a given string

Memory Layout of C Programs

Understanding “extern” keyword in C

Median of two sorted arrays

Tree traversal without recursion and without stack!

Structure Member Alignment, Padding and Data Packing

Intersection point of two Linked Lists

Lowest Common Ancestor in a BST.

Check if a binary tree is BST or not

Sorted Linked List to Balanced BST

```
  }
}

/* UTILITY FUNCTIONS */
/* Helper function that allocates a new node with the
   given data and NULL left and right pointers. */
struct node* newnode(int data)
{
  struct node* node = (struct node*)
                       malloc(sizeof(struct node));
  node->data = data;
  node->left = NULL;
  node->right = NULL;

  return(node);
}

/* Driver program to test above functions*/
int main()
{

  int sum = 21;

  /* Constructed binary tree is
            10
          /    \
        8       2
      /   \    /
    3      5  2
  */
  struct node *root = newnode(10);
  root->left        = newnode(8);
  root->right       = newnode(2);
  root->left->left  = newnode(3);
  root->left->right = newnode(5);
  root->right->left = newnode(2);

  if(hasPathSum(root, sum))
    printf("There is a root-to-leaf path with sum %d", sum);
  else
    printf("There is no root-to-leaf path with sum %d", sum);

  getchar();
  return 0;
}
```

Time Complexity: O(n)

References:
http://cslibrary.stanford.edu/110/BinaryTrees.html

Author: Tushar Roy

Please write comments if you find any bug in above code/algorithm, or find other ways to solve the same problem

Related Tpoics:

- Print a Binary Tree in Vertical Order | Set 2 (Hashmap based Method)
- Print Right View of a Binary Tree
- Red-Black Tree | Set 3 (Delete)
- Construct a tree from Inorder and Level order traversals
- Print all nodes at distance k from a given node
- Print a Binary Tree in Vertical Order | Set 1
- Interval Tree

695 Subscribe

Recent Comments

affiszerv Your example has two 4s on row 3, that's why it...

Backtracking | Set 7 (Sudoku) · 43 minutes ago

RVM Can someone please elaborate this Qs from above...

Flipkart Interview | Set 6 · 1 hour ago

Vishal Gupta I talked about as an Interviewer in general,...

Software Engineering Lab, Samsung Interview | Set 2 · 1 hour ago

@meya Working solution for question 2 of 4f2f round....

Amazon Interview | Set 53 (For SDE-1) · 1 hour ago

sandeep void rearrange(struct node *head) {...

Given a linked list, reverse alternate nodes and append at the end · 3 hours ago

Neha I think that is what it should return as, in...

Find depth of the deepest odd level leaf node · 3 hours ago

- Check if a given Binary Tree is height balanced like a Red-Black Tree

6 Tweet 1 2

Writing code in comment? Please use **ideone.com** and share the link here.