

Link nodes at same level in a binary tree

Definition of data structures

You are given a binary tree:

```
// C or C++  
  
struct node  
{  
  
    int n; // value of node  
    struct node *left; // left subtree  
    struct node *right; // right subtree  
    struct node *level; // level pointer (node "to the right")  
  
}
```

Initially, the level field is set to NULL.

Tasks to complete during coding test

1. Write a function that will link all the nodes at the same level in a tree (not only from example, but any given tree).

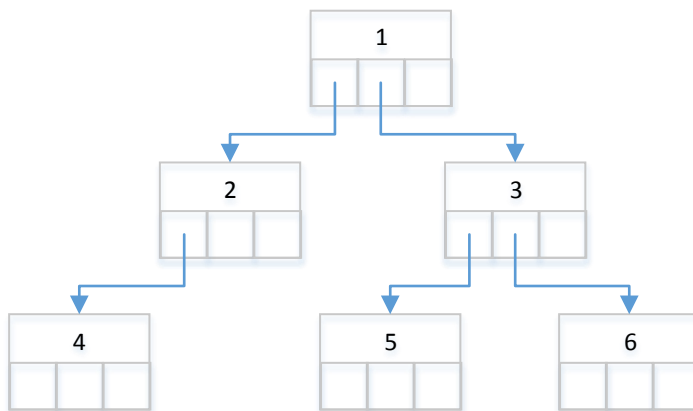
```
// C or C++  
void linkSameLevel(struct node *t);
```

Your solution will be tested using automated tests so do not change signature of linkSameLevel method or Node class!

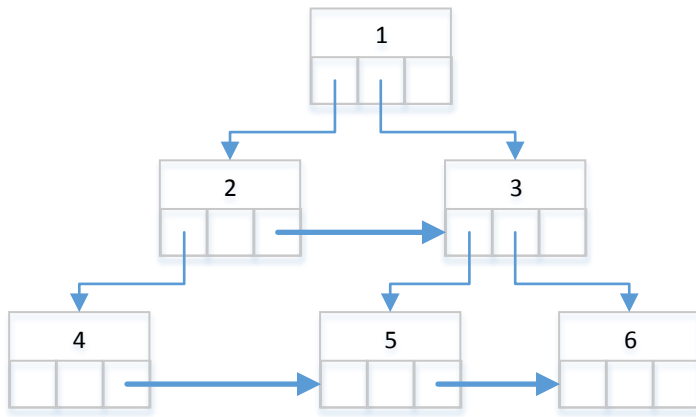
2. Please explain what the running time and memory usage of your function are for a tree of depth d containing n nodes.

Example

If you are given the tree:



your implementation of `linkSameLevel` function should create the links (i.e. pointers) illustrated in bold below:



For Java, C# or Objective-C developers

Please consider following definitions:

// C# or Java

```
public class Node
```

```
{
```

```
    public int n; // value of node
```

```
    public Node left; // left subtree
```

```
    public Node right; // right subtree
```

```
    public Node level; // level pointer (node "to the right")
```

```
}
```

```
public static void linkSameLevel(Node t);
```

// Objective-C

```
@interface Node : NSObject
```

```
@property (nonatomic) NSInteger n;
```

```
@property (nonatomic) Node *left;
```

```
@property (nonatomic) Node *right;
```

```
@property (nonatomic) Node *level;
```

```
@end
```

```
void linkSameLevel(Node* node);
```