On my honor, I pledge that I have neither recieved nor provided any improper assistance in the completion of this assignment.

—2190012 김석원

Homework

- When you rebuild an AVL tree from BST using recycling, how the left and right of the leaf nodes are to set to **`nullptr`**. Explain how they are being set in your code.
- You may add new pages of ppt to answer the question.

Building AVL tree from BST in O(n) – recycling method

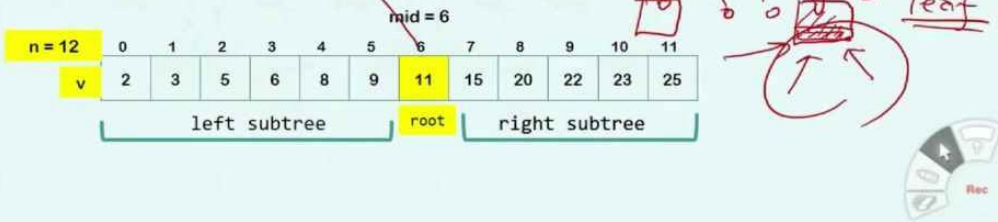
```
// rebuilds an AVL tree using a list of nodes sorted, no memory allocations
// v - an array of nodes sorted, n - the array size
tree buildAVL(tree* v, int n) {
  if (n <= 0) return nullptr;
  int mid = n / 2;

  tree root = v[mid];                    // mid becomes the root; don't call new TreeNode.

  // recursive buildAVL() calls for left & right, return it to root->left & root->right



  return root;
}
```



```
tree buildAVL(vector<tree>& v, int n){
    if (n <= 0) return nullptr;
    DPRINT(cout << ">buildAVL v[0]=" << v[0] << " n="
    int mid=n/2;

    tree root=v[mid];

    vector<tree> left(v.begin(), v.begin()+mid);
    vector<tree> right(v.begin()+mid+1, v.end());

    root->left=buildAVL(left,left.size());
    root->right=buildAVL(right,right.size());

    DPRINT(cout << "<buildAVL" << n << endl;);
    return root;
}
```

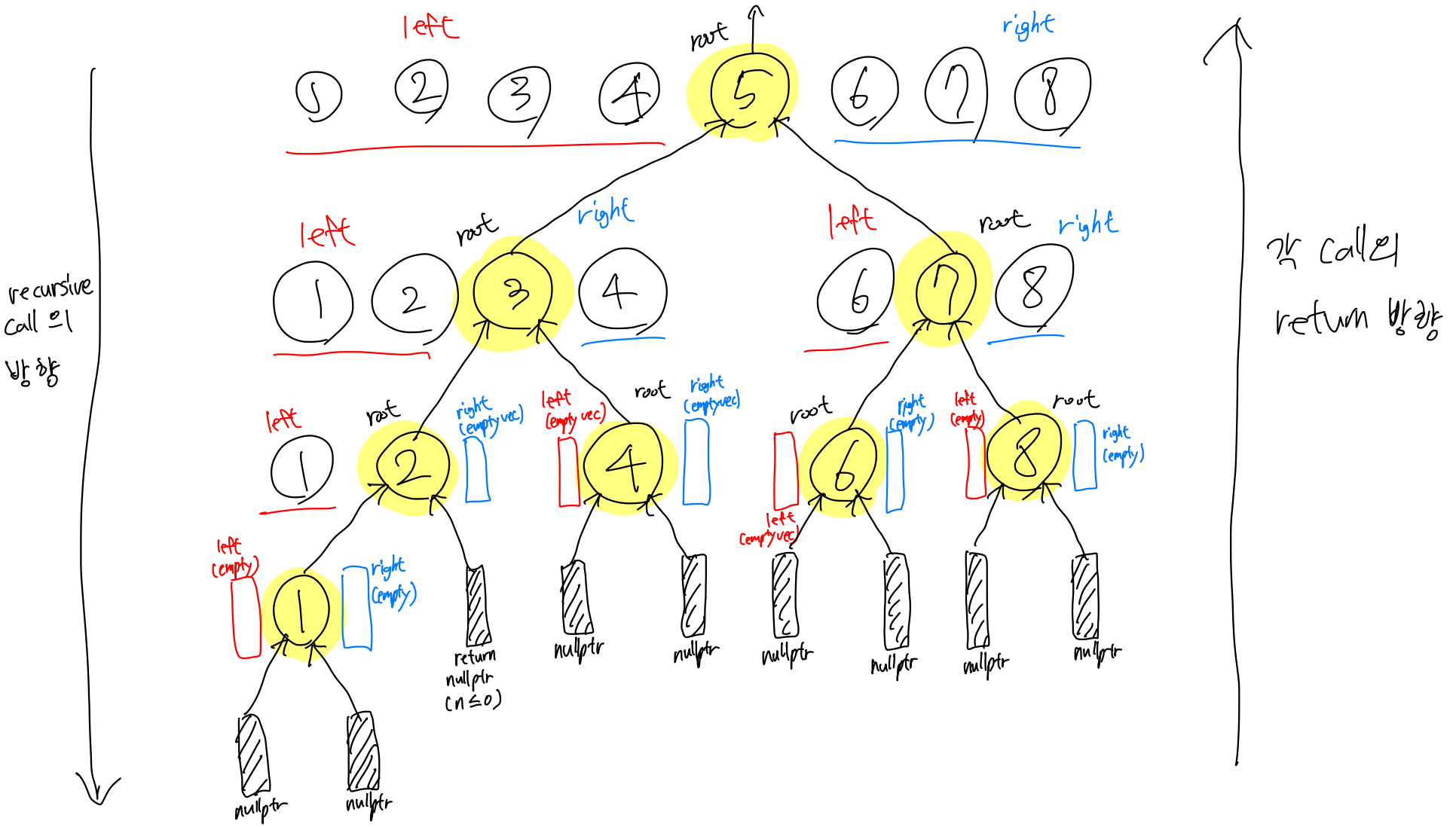
윗쪽의 코드가 직접 구현한 코드이다. recycle 하기 위해 key값이 아니라 node 자체로 구성된 vector를 parameter로 받아야 한다.

벡터의 요소 중, 중간 노드를 root로 정하고, 나머지 앞, 뒤 남은 요소들을 iterator을 사용하여 새로운 벡터 left, right에 배정하게 된다.

이제, parameter로 받은 vector는 처음부터 중간 전까지 요소가 있는 left, 가운데 노드(root), 중간 다음부터 마지막 요소까지 있는 right 노드, 총 세 가지로 나누어지게 된다.

이제 root->left에는 left vector, root->right에는 right vector을 parameter로 각각 배정하여 recursive call로 tree을 build하게 된다. 여기서 base case을 살펴본다면, N스0, 즉 vector의 size가 0보다 작거나 같으면 nullptr을 return 한다. 그런데 vector의 길이가 0이 되려면 빈 벡터여야만 한다. 벡터가 비는 경우는 요소가 하나 뿐인 벡터가 세 부분으로 나누면서 recursive call이 되는 경우 뿐이다. 벡터의 요소가 하나 뿐이라는 것은 마지막 leaf node라는 것이고, 이 노드의 left와 right는 vector.size()==0 이기 때문에 각각 nullptr이 리턴되고, system stack에 저장되어 있던 이전 노드들에게 차례로 return 되면서 tree가 완성되게 된다.

ex)



이런 식으로, 모든 leaf node의 left와 right가 nullptr로 초기화된다.