297. Serialize and Deserialize Binary Tree

link

这道题跟serialize binary search tree很像, 但是那个题要求string越短越好, 这个题必须存储null, 两个地方要注意

1. 用的`node[i] == "#"`结果总是返回false, 应该用String.equals
2. 后面循环i++的时候得重新检查index是不是越界.

```
public String serialize(TreeNode root) {
    if(root == null) return "";
    StringBuilder sb = new StringBuilder();
    Queue<TreeNode> queue = new LinkedList();
    queue.offer(root);
    while(!queue.isEmpty()){
        int size = queue.size();
        for(int i = 0; i < size; ++i){
            TreeNode node = queue.poll();
            if(node == null)
                sb.append("# ");
            else{
                sb.append(String.valueOf(node.val) + " ");
                queue.offer(node.left);
                queue.offer(node.right);
            }
        }
    }
    while(sb.charAt(sb.length() - 2) == '#'){
        sb.setLength(sb.length() - 2);
    }
    sb.setLength(sb.length() - 1);
    return sb.toString();
}

// Decodes your encoded data to tree.
public TreeNode deserialize(String data) {
    if(data.equals("")) return null;
    String[] nodes = data.split(" ");
    Queue<TreeNode> queue = new LinkedList();
    TreeNode root = new TreeNode(Integer.parseInt(nodes[0]));
    queue.offer(root);
    int i = 1;
    while(i < nodes.length){
        TreeNode node = queue.poll();
        if(!"#".equals(nodes[i])){
            node.left = new TreeNode(Integer.parseInt(nodes[i]));
            queue.offer(node.left);
            }
            ++i;
//这里得重新检查是不是越界.
            if(i < nodes.length && !"#".equals(nodes[i])){
                node.right = new TreeNode(Integer.parseInt(nodes[i]));
                queue.offer(node.right);
            }
            ++i;
        }
        return root;
    }
```

也可以用preOrder传, 一样的.

link