

7_自增ID

已知有如下一个树状数据结构：

```
let tree = {
  id: '1',
  type: 'View',
  name: 'view',
  children: [
    {
      id: '2',
      type: 'Button',
      name: 'button'
    },
    {
      id: '3',
      type: 'View',
      name: 'view_1'
      children: [
        {
          id: '4',
          type: 'Button',
          name: 'button_1'
        },
        {
          id: '5',
          type: 'View',
          name: 'view_2'
        }
      ]
    }
  ]
};
```

name 字段的规则为整棵树不能重复，如果遇到重复，则添加 _n 作为结尾。如果因为删除某些节点，名字出现了空隙，比如 Tree 中有 button_1 和 button_3，但是没有 button_2，下一

次插入时，需要优先使用 button_2，而不是 button_4。

请实现一个唯一名称算法，当向整棵树的任意children插入一个新的Node时，可以保证 name 不重复。

srcName 是计划向 Tree 中插入节点的 name，比如 button、view,
rootTreeNode 是整棵树

```
function getIncName(srcName: string, rootTreeNode : TreeNode): string;
```