

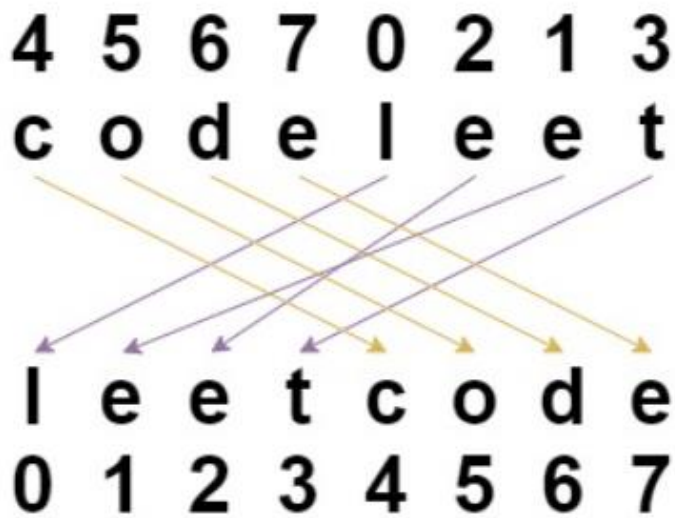
重新排列字符串

给你一个字符串 s 和一个 **长度相同** 的整数数组 $indices$ 。

请你重新排列字符串 s ，其中第 i 个字符需要移动到 $indices[i]$ 指示的位置。

返回重新排列后的字符串。

示例 1：



输入： $s = \text{"codeleet"}$, $indices = [4, 5, 6, 7, 0, 2, 1, 3]$

输出： "leetcode"

解释： 如图所示， "codeleet" 重新排列后变为 "leetcode" 。

示例 2：

输入： $s = \text{"abc"}$, $indices = [0, 1, 2]$

输出： "abc"

解释： 重新排列后，每个字符都还留在原来的位置上。

示例 3：

输入： $s = \text{"aiohn"}$, $indices = [3, 1, 4, 2, 0]$

输出： "nihao"

示例 4：

输入: s = "aaiougrt", indices = [4,0,2,6,7,3,1,5]

输出: "arigatou"

示例 5:

输入: s = "art", indices = [1,0,2]

输出: "rat"

```
class Solution {
public:
    string restoreString(string s, vector<int>& indices) {
        string str(s.size(),0);
        for(int i=0;i<indices.size();i++)
        {
            str[indices[i]]=s[i];
        }
        return str;
    }
};
```