5441. Making File Names Unique

My Submissions (/contest/weekly-contest-194/problems/making-file-names-unique/submissions/)

Back to Contest (/contest/weekly-contest-194/)

Given an array of strings `names` of size `n`. You will create `n` folders in your file system **such that**, at the `ith` minute, you will create a folder with the name `names[i]`.

Since two files **cannot** have the same name, if you enter a folder name which is previously used, the system will have a suffix addition to its name in the form of `(k)`, where, `k` is the **smallest positive integer** such that the obtained name remains unique.

Return *an array of strings of length* `n` where `ans[i]` is the actual name the system will assign to the `ith` folder when you create it.

User Accepted:	387
User Tried:	1213
Total Accepted:	387
Total Submissions:	1772
Difficulty:	Medium

Example 1:

```
Input: names = ["pes","fifa","gta","pes(2019)"]
Output: ["pes","fifa","gta","pes(2019)"]
Explanation: Let's see how the file system creates folder names:
"pes" --> not assigned before, remains "pes"
"fifa" --> not assigned before, remains "fifa"
"gta" --> not assigned before, remains "gta"
"pes(2019)" --> not assigned before, remains "pes(2019)"
```

Example 2:

```
Input: names = ["gta","gta(1)","gta","avalon"]
Output: ["gta","gta(1)","gta(2)","avalon"]
Explanation: Let's see how the file system creates folder names:
"gta" --> not assigned before, remains "gta"
"gta(1)" --> not assigned before, remains "gta(1)"
"gta" --> the name is reserved, system adds (k), since "gta(1)" is also reserved, systems
"avalon" --> not assigned before, remains "avalon"
```

Example 3:

```
Input: names = ["onepiece","onepiece(1)","onepiece(2)","onepiece(3)","onepiece"]
Output: ["onepiece","onepiece(1)","onepiece(2)","onepiece(3)","onepiece(4)"]
Explanation: When the last folder is created, the smallest positive valid k is 4, and it k
```
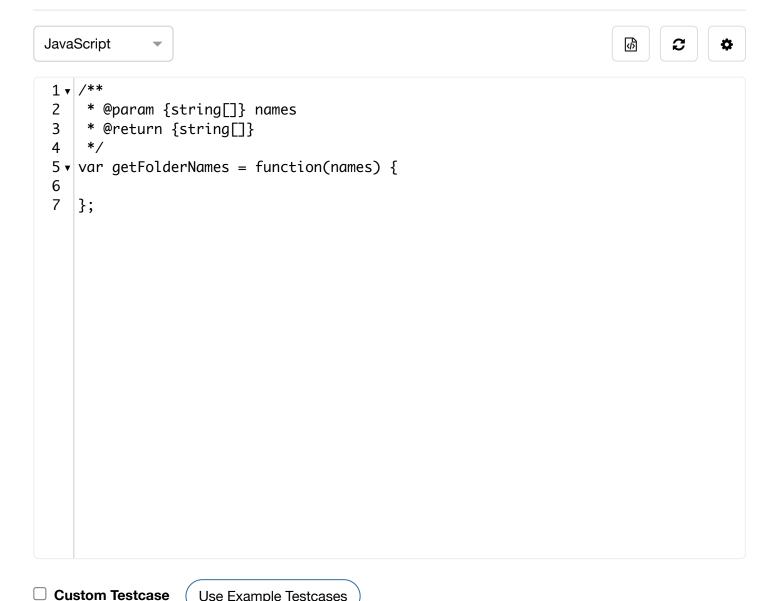
Example 4:

```
Input: names = ["wano","wano","wano","wano"]
Output: ["wano","wano(1)","wano(2)","wano(3)"]
Explanation: Just increase the value of k each time you create folder "wano".
```

Example 5:

```
Input: names = ["kaido","kaido(1)","kaido","kaido(1)"]
Output: ["kaido","kaido(1)","kaido(2)","kaido(1)(1)"]
Explanation: Please note that system adds the suffix (k) to current name even it contained
```

Constraints:

- `1 <= names.length <= 5 * 10^4`
- `1 <= names[i].length <= 20`
- `names[i]` consists of lower case English letters, digits and/or round brackets.

JavaScript ▾

```javascript
1 ▾ /**
2    * @param {string[]} names
3    * @return {string[]}
4    */
5 ▾ var getFolderNames = function(names) {
6
7   };
```

☐ **Custom Testcase**   (Use Example Testcases)