

给定一个只包含三种字符的字符串：(,) 和 *，写一个函数来检验这个字符串是否为有效字符串。有效字符串具有如下规则：

1. 任何左括号 (必须有相应的右括号)。
2. 任何右括号) 必须有相应的左括号 (。
3. 左括号 (必须在对应的右括号之前)。
4. * 可以被视为单个右括号)，或单个左括号 (，或一个空字符串。
5. 一个空字符串也被视为有效字符串。

示例 1:

输入: "()"

输出: True

示例 2:

输入: "(*)"

输出: True

示例 3:

输入: "(*))"

输出: True

注意:

1. 字符串大小将在 [1, 100] 范围内。

思路这题的主要思路是用两个数组来保存 '(' 和 '*' 这两个符号的位置，然后每当出现 ')' 的时候就对两个数组进行比较，如果两个数组为空那么这个字符串为非法，如果有则优先将 '(' 符号匹配，然后再和 '*' 符号匹配，到最后时如果 '(' 数量多余 '*' 数量，那么必为非法，当小于时则判断是否存在与之匹配的 '*'，即每一个 '(' 符号都要有一个存在于该字符后面的 '*' 符号才可以匹配。

by <https://blog.csdn.net/curry3030/article/details/80879375>

代码

```
class Solution {
public:
    bool checkValidString(string s) {
        int star=0;
        vector<int> starnum(s.size(),0);
```

```

vector<int> leftnum(s.size(),0);
int left=0;
for(int i=0;i<s.size();i++){
    if(s[i]=='('){
        leftnum[left++]=i;
    }else if(s[i]=='*'){
        starnum[star++]=i;
    }else{
        if(left>0){
            left--;
        }else if(star>0){
            star--;
        }else{
            return false;
        }
    }
}
if(left>star){
    return false;
}
for(;left>0;left--,star--){
    if(leftnum[left-1]>starnum[star-1]){
        return false;
    }
}
return true;
}
};

```