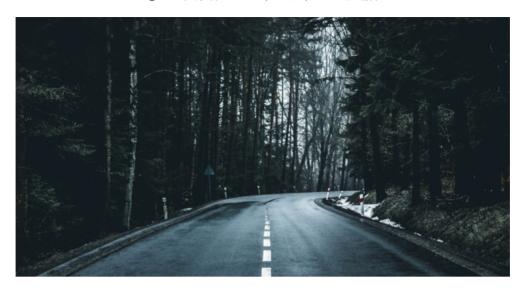
剑指Offer(六十五):矩阵中的路径

⑤ 2018年2月3日 11:01:41 ♀ 6 ◎ 8,445 °C ♣ 编辑



一、前言

本系列文章为《剑指Offer》刷题笔记。

刷题平台: 牛客网

书籍下载: 共享资源

二、题目

请设计一个函数,用来判断在一个矩阵中是否存在一条包含某字符串所有字符的路径。路径可以从矩阵中的任意一个格子开始,每一步可以在矩阵中位左,向右,向上,向下移动一个格子。如果一条路径经过了矩阵中的某一个格子,则该路径不能再进入该格子。例如在下面的3x4的矩阵中包含一条字符串"bcced"的路径(路径中的字母用斜体表示)。但是矩阵中不包含"abcb"路径,因为字符串的第一个字符b占据了矩阵中的第一行第二个格子之后,路径不能再次进入这个格子。

$$a \quad b \quad c \quad e$$
 $s \quad f \quad c \quad s$
 $a \quad d \quad e \quad e$

1、思路

这是一个可以用回溯法解决的典型问题。

首先,遍历这个矩阵,我们很容易就能找到与字符串str中第一个字符相同的矩阵元素ch。然后遍历ch的上下左右四个字符,如果有和字符串str中下一字符相同的,就把那个字符当作下一个字符(下一次遍历的起点),如果没有,就需要回退到上一个字符,然后重新遍历。为了避免路径重叠,需要一个知助矩阵来记录路径情况。

下面代码中,当矩阵坐标为(row, col)的格子和路径字符串中下标为pathLength的字符一样时,从4个相邻的格子(row, col-1)、(row-1, col)、(row, col+1)以及(row+1, col)中去定位路径字符串中下标为pathLength+1的字符。

如果4个相邻的格子都没有匹配字符串中下标为pathLength+1的字符,表明当前路径字符串中下标为pathLength的字符在矩阵中的定位不正确,我们需要回到前一个字符串(pathLength-1),然后重新定位。

一直重复这个过程,直到路径字符串上所有字符都在矩阵中找到格式的位置(此时str[pathLength] == '\0')。

2、代码

C++:

```
1 class Solution {
2 public:
3  bool hasPath(char* matrix, int rows, int cols, char* str)
4
```

```
if(matrix == NULL || rows < 1 || cols < 1 || str == NULL){</pre>
                           return false;
 6
7
8
9
                    bool* visited = new bool[rows*cols];
memset(visited, 0, rows*cols);
int pathLength = 0;
for(int row = 0; row < rows; row++){
   for(int col = 0; col < cols; col++){
      if(hasPathCore(matrix, rows, cols, row, col, str, pathLength, visited)){</pre>
                                         delete[] visited;
return true;
                          }
                    delete[] visited;
return false;
             bool hasPathCore(char* matrix, int rows, int cols, int row, int col, char* str, int& pathLength, bool* visited){
    if(str[pathLength] == '\0'){
                           return true;
                    bool hasPath = false;
if(row >= 0 && row < rows && col >= 0 && col < cols && matrix[row*cols+col] == str[pathLength] && !visited[row*cols+col]){</pre>
                           ++pathLength;
visited[row*cols+col] = true;
                           Ntstted[row*cots+cot] = true;
hasPath = hasPathCore(matrix, rows, cols, row-1, col, str, pathLength, visited)
| hasPathCore(matrix, rows, cols, row+1, col, str, pathLength, visited)
| hasPathCore(matrix, rows, cols, row, col-1, str, pathLength, visited)
| hasPathCore(matrix, rows, cols, row, col+1, str, pathLength, visited);
                           if(!hasPath){
    --pathLength;
                                   visited[row*cols+col] = false;
40
                    return hasPath;
41
42
```



微信公众号

分享技术,乐享生活:微信公众号搜索「JackCui-Al」关注一个在互联网摸爬滚打的潜行者。

当我沉默的时候,我觉得很充实,当我开口说话,就感到了空虚。--- 鲁迅