

Trie

#G面试准备

Word Search II

Word Search II - LeetCode

做了两次 每次都有很多蠢的错误

第一次

- 写了完整的trie 却不知道怎么写search
 - 其实只需要写insert 遍历的时候就在当前node的map里去找是否containsKey(c)就行
- 看了答案才知道不是去得到一个完整string才能去trie里找是否存在这个string或是否存在这个前缀
- search DFS递归函数
 - 参数：board, x, y, TrieNode node, List ans
 - 外部调用的时候传入的node是root 从root开始
 - 递归的出口：node.isEnd
 - 其实不算是出口 里面的return不要写 防止当前找到的词是字典里更长的一个词的前缀 我们要找到所有的词 所以不退
 - ans list里是否已经找到过这个词？没找到过再加进去
 - 判断边界和hash是否遍历过要在出口的下面 否则跟word search第一题一样 如果棋盘只有一个字母的话 是不会走到出口的 → WA在此1
 - 判断board[x][y]在不在当前node的map里 如果在才会进行下面的四向DFS遍历
- TrieNode里如果isEnd == true的话 要存一个s变量 存这个位置是什么单词 这样递归出口的时候就能直接把该单词加入结果list里
- 要有hash表
 - 直接改了board[x][y] == '0'
 - 先用个临时变量c存board[x][y]
 - search后面再改回board[x][y] = c

第二次

- Trie只写了insert
- 一些蠢的错：
 - $x > m$ $y > n$ 应该是 $x \geq m$
 - `board[x][y] = 'c';` 应该是 `board[x][y] = c;`
 - `root.map.get(board[x][y])` `root.map.get(c)`
 - 把'0' 传进去了 导致递归下一轮null pointer