

剑指Offer（五十二）：正则表达式匹配

🕒 2018年1月22日 11:47:42 🗨 发表评论 🌡 4,097 °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

请实现一个函数用来匹配包括 '.' 和 '*' 的正则表达式。模式中的字符 '.' 表示任意一个字符，而 '*' 表示它前面的字符可以出现任意次（包含0次）。在本题中，匹配是指字符串的所有字符匹配整个模式。例如，字符串 "aaa" 与模式 "a.a" 和 "ab*ac*a" 匹配，但是与 "aa.a" 和 "ab*a" 均不匹配。

1、思路

这道题有些绕，需要好好思考下。

我们先来分析下如何匹配一个字符，现在只考虑字符 '.'，不考虑 '*' 看一下：

如果字符串和模式串的当前字符相等，那么我们继续匹配它们的下一个字符；如果模式串中的字符是 '.'，那么它可以匹配字符串中的任意字符，我们也可以继续匹配它们的下一个字符。

接下来，把字符 '*' 考虑进去，它可以匹配任意次的字符，当然出现0次也可以。

我们分两种情况来看：

- 模式串的下一个字符不是 '*'，也就是上面说的只有字符 '.' 的情况。

如果字符串中的第一个字符和模式串中的第一个字符相匹配，那么字符串的模式串都向后移动一个字符，然后匹配剩余的字符串和模式串。如果字符串中的第一个字符和模式串中的第一个字符不相匹配，则直接返回 false。

- 模式串的下一个字符是 '*'，此时就要复杂一些。

因为可能有多种不同的匹配方式。

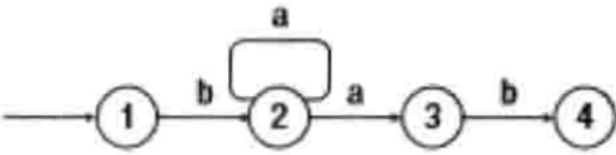
选择一：无论字符串和模式串当前字符相等不相等，我们都将模式串后移两个字符，相当于把模式串中的当前字符和 '*' 忽略掉，因为 '*' 可以匹配任意次的字符，所以出现0次也可以。

选择二：如果字符串和模式串当前字符相等，则字符串向后移动一个字符。而模式串此时有两个选择：

1、我们可以在模式串向后移动两个字符，继续匹配；

2、也可以保持模式串不变，这样相当于用字符 '*' 继续匹配字符串，也就是模式串中的字符 '*' 匹配字符串中的字符多个的情况。

用一张图表示如下：



如上图所示，当匹配进入状态2，并且字符串中的字符是'a'时，我们有两个选择：可以进入状态3（在模式串向后移动两个字符），也可以回到状态2（模式串保持不变）。

除此之外，还要注意对空指针的处理。

2、代码

C++:

C+

```
1 class Solution {
2 public:
3     bool match(char* str, char* pattern)
4     {
5         // 指针为空，返回false
6         if(str == NULL || pattern == NULL){
7             return false;
8         }
9         return matchCore(str, pattern);
10    }
11 private:
12     bool matchCore(char* str, char* pattern){
13         // 字符串和模式串都运行到了结尾，返回true
14         if(*str == '\0' && *pattern == '\0'){
15             return true;
16         }
17         // 字符串没有到结尾，模式串到了，则返回false
18         // 模式串没有到结尾，字符串到了，则根据后续判断进行，需要对'*'做处理
19         if((*str != '\0' && *pattern == '\0')){
20             return false;
21         }
22         // 如果模式串的下一个字符是'*'，则进入状态机的匹配
23         if(*(pattern + 1) == '*'){
24             // 如果字符串和模式串相等，或者模式串是'.', 并且字符串没有到结尾，则继续匹配
25             if(*str == *pattern || (*pattern == '.' && *str != '\0')){
26                 // 进入下一个状态，就是匹配到了一个
27                 return matchCore(str + 1, pattern + 2) ||
28                    // 保持当前状态，就是继续那个'*'去匹配
29                    matchCore(str + 1, pattern) ||
30                    // 跳过这个'*'
31                    matchCore(str, pattern + 2);
32             }
33             // 如果字符串和模式串不相等，则跳过当前模式串的字符和'*'，进入新一轮的匹配
34             else{
35                 // 跳过这个'*'
36                 return matchCore(str, pattern + 2);
37             }
38         }
39         // 如果字符串和模式串相等，或者模式串是'.', 并且字符串没有到结尾，则继续匹配
40         if(*str == *pattern || (*pattern == '.' && *str != '\0')){
41             return matchCore(str + 1, pattern + 1);
42         }
43         return false;
44     }
45 };
```



微信公众号

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

益者三友。友直，友谅，友多闻，益矣。--- 孔子