

## 正则表达式 ( 2 ) : 连续次数的匹配

在本博客中, "正则表达式"为一系列文章, 如果你想要从头学习怎样在Linux中使用正则, 可以参考此系列文章, 直达链接如下:

[在Linux中使用正则表达式](#)

"正则"系列的每篇文章都建立在前文的基础之上, 所以, [请按照顺序阅读这些文章](#), 否则有可能在阅读中遇到障碍。

上一篇正则表达式的文章中, 我们总结了跟"位置匹配"有关的正则, 此处, 我们来认识一些跟"连续次数匹配"有关的正则。

"连续次数匹配"是什么意思呢? 空口白话说不容易明白, 看完下例就能明白, 首先, 我们准备一个测试文件, 文件内容如下。

```
[www.zsythink.net]#cat regex.txt
a a
aa
a aa
bb
bbb
c cc ccc
dddd d dd ddd
ab abc abcc
ef eef eeef
[www.zsythink.net]#
```

测试文本regex.txt的内容如上图所示。

如果我们想要从regex.txt文本中找出哪些行包含两个连续的字母a, 我们应该怎样去查找呢? 我们可以使用如下方法

```
[www.zsythink.net]#cat -n regex.txt
 1 a a
 2 aa
 3 a aa
 4 bb
 5 bbb
 6 c cc ccc
 7 dddd d dd ddd
 8 ab abc abcc
 9 ef eef eeef
[www.zsythink.net]#
[www.zsythink.net]#grep --color -n "aa" regex.txt
 2:aa
 3:a aa
[www.zsythink.net]#
```

没错, 我们直接使用grep命令, 在文本中搜索"aa"即可, 因为"aa"就是两个连续的a字母。

可以看到, 文本中的第二行和第三行中都包含两个连续的a, 所以第二行与第三行被打印了出来。

但是, 如果我们要在文本中搜索10个连续的a字母呢? 好吧, 我们可以搜索"aaaaaaaa"字符串

如果我们想要在文本中搜索100个连续的a字母呢? 难道还要写100个连续的a? 这样显然有点累, 我们可以利用正则解决这个问题, 示例如下。

利用grep命令和正则表达式, 即可找出哪些行包含2个连续的字母a, 示例如下

```
[www.zsythink.net]#cat -n regex.txt
 1 a a
 2 aa
 3 a aa
 4 bb
 5 bbb
 6 c cc ccc
 7 dddd d dd ddd
 8 ab abc abcc
 9 ef eef eeef
[www.zsythink.net]#
[www.zsythink.net]#grep --color -n "a{2}" regex.txt
 2:aa
 3:a aa
[www.zsythink.net]#
```

聪明如你一定看懂了, 没错, "{2}"就表示"连续出现2次", 所以, "a{2}"就表示a连续出现两次, 可以看到, 包含2个连续字母a的行只有第二行, 所以, 当我们使用"a{2}"时, 只能匹配到第二行, 由于第一行中的两个字母a中间存在"空格", 所以并不能算作两个连续的字母a, 所以没有被匹配到。

你肯定已经学会举一反三了, "{2}"表示连续出现2次, 那么, "{5}"就表示连续出现5次, "{100}"就表示连续出现100次, 没错, 我们只要替换其中的数字, 即几次。

我们总结一下刚才的语法

使用"(x)"表示之前的字符连续出现x次将会被匹配到。

不过需要注意, 如果字符连续出现的次数大于指定的次数, 也是可以匹配到的, 示例如下:

```
[www.zsythink.net]#cat -n regex.txt
1  a a
2  aa
3  a aa
4  bb
5  bbb
6  c cc ccc
7  dddd d dd ddd
8  ab abc abcc
9  ef eef eeef
[www.zsythink.net]#
[www.zsythink.net]#grep --color -n "b\{2\}" regex.txt
4:bb
5:bbb
[www.zsythink.net]#
```

zsythink.net 宋双印博客

正则表达式中,我们指定, b字母连续出现2次则会被匹配到,所以,第4行被匹配到了,同时,第5行也被匹配到了,因为第5行中, b字母连续出现了3次,包含2个连续的字母b也被匹配到了。

如果你不想出现上述情况,只是想要精准的匹配连续出现2次且只出现了2次的字母b,应该怎么办呢?其实我们在前文中已经学到了解决问题的方法,示例如下

```
[www.zsythink.net]#grep --color -n "\<b\{2\}\>" regex.txt
4:bb
[www.zsythink.net]#
```

没错,就是结合了上次介绍到的单词定界符,锚定词首与锚定词尾,如果你没有看出来上述正则表达式什么意思,那么请回顾上一篇文章。

那么现在,我们来

延伸一下,你来猜猜"\{x,y}"表示什么?

"\{x,y}"表示之前的字符至少连续出现x次,最多连续出现y次,都能被匹配到,换句话说,只要之前的字符连续出现的次数在x与y之间,即可被匹配到,示例如下。

```
[www.zsythink.net]#grep --color -n "d\{2,4\}" regex.txt
7:dddd d dd ddd
[www.zsythink.net]#
```

如上图所示,连续出现2次的d字母、连续出现3次的d字母、连续出现4次的d字母都被匹配到了。

好了,现在我们已经了解了两种语法。

\{x\} 表示之前的字符连续出现x次时会被匹配到。

\{x,y\} 表示之前的字符至少连续出现x次,至多连续出现y次,都可以被匹配到,x与y之间用逗号隔开。

那么,我们再延伸一下,你猜猜... \{x,\} 与 \{,y\} 分别表示什么意思?

没错,你肯定已经猜到了

\{x,\}表示之前的字符至少连续出现x次,或者连续出现次数大于x次,即可被匹配到,上不封顶。

\{,y\}表示之前的字符至多连续出现y次,或者连续出现次数小于y次,即可被匹配到,最小次数为0次,换句话说,之前的字符连续出现0次到y次,都会被匹配到。

示例如下:

```
[www.zsythink.net]#cat -n regex.txt
1  a a
2  aa
3  a aa
4  bb
5  bbb
6  c cc ccc
7  dddd d dd ddd
8  ab abc abcc
9  ef eef eeef
[www.zsythink.net]#
[www.zsythink.net]#grep --color -n "d\{2,\}" regex.txt
7:dddd d dd ddd
[www.zsythink.net]#
```

zsythink.net 宋双印博客

如上图所示,字母d连续出现2次以及2次以上的都被匹配到了。

```
[www.zsythink.net]#grep --color -n "abc\{,2\}" regex.txt
8:ab abc abcc
[www.zsythink.net]#
```

如上图所示,abc、abcc都被匹配到了,因为"c\{2\}"表示只要c字母连续出现的次数小于等于2,即可被匹配到,再配合之前的"ab"字符,所以,abc

、abcc都被匹配到了,ab为什么也被匹配到了呢?之前说过,"\{,y\}"表示之前的字符连续出现0次到y次,都会被匹配到,所以,ab被匹配到了,相当于c被匹配到

现在我们来认识一个用于匹配次数的正则符号,它就是\*

如果你之前使用过通配符,那么你肯定对\*非常熟悉,在通配符中,\*表示匹配任意长度的任意字符。

但是，在正则表达式中，\*代表另一个意思，在正则表达式中，\*表示之前的字符连续出现任意次数（包括0次），不要与通配符中的\*搞混淆了。

示例如下

```
[www.zsythink.net]#cat -n regex.txt
1  a a
2  aa
3  a aa
4  bb
5  bbb
6  c cc ccc
7  dddd d dd ddd
8  ab abc abcc
9  ef eef eeef
[www.zsythink.net]#
[www.zsythink.net]#grep --color -n "e*f" regex.txt
9:ef eef eeef
[www.zsythink.net]#
[www.zsythink.net]#
```

如上图所示，"e\*f"表示e出现任意次，f必须跟在e的后头。

注意，\*表示之前的字符连续出现任意次数，包括0次，即可被匹配到，理解了这一点，再看如下示例，就简单了。

```
[www.zsythink.net]#grep --color -n "d*" regex.txt
1:a a
2:aa
3:a aa
4:bb
5:bbb
6:c cc ccc
7:dddd d dd ddd
8:ab abc abcc
9:ef eef eeef
[www.zsythink.net]#
```

如上图所示，"d\*"表示d连续出现任意次数，即可被匹配到，所以，第7行高亮显示了。

但是其他行为什么也被打印出来了呢？这是因为\*表示连续出现任意次数，包括0次。

其他行中，根本不包含字母d，换句话说就是，d连续出现了0次，所以其他行也符合条件，最终也被grep输出了。

那么，在通配符中，\*表示匹配任意长度的任意字符，在正则中，怎样表示任意长度的任意字符呢？

在正则表达式中，使用"."表示任意长度的任意字符。

我们先看示例，回头再解释为什么"."表示任意长度的任意字符，示例如下。

```
[www.zsythink.net]#cat -n regex.txt
1  a a
2  aa
3  a aa
4  bb
5  bbb
6  c cc ccc
7  dddd d dd ddd
8  ab abc abcc
9  ef eef eeef
[www.zsythink.net]#
[www.zsythink.net]#grep --color -n "a.*" regex.txt
1:a a
2:aa
3:a aa
8:ab abc abcc
[www.zsythink.net]#
```

上图中的正则表达式表示，a字母后面存在任意长度的任意字符，都可以被匹配到，如上图所示，的确都被匹配到了。

其实，在正则表达式中，"."表示匹配任意单个字符，示例如下。

```
[www.zsythink.net]#grep --color -n "ee." regex.txt
9:ef eef eeef
[www.zsythink.net]#grep --color -n "ee.." regex.txt
9:ef eef eeef
[www.zsythink.net]#
```

如上图所示

"ee."表示"ee"后面跟随任意一个单个字符，都会被匹配到

"ee.."表示"ee"后面跟随任意两个字符，都会被匹配到，由于"空格"也算作单个字符，所以，"eef空格"也被匹配到了，因为"f"和"空格"被看做了两个字符。

理解完上述示例，再回过头来理解".\*"，就容易多了，".\*"可以理解为"."与"\*"的结合，"."在正则中表示"连续出现任意次的任意单个字符"，换句话说就是，任意长字符，正则表达式中的"."与通配符中的"\*"所表达的意思一样。

理解完上述符号以后，再来认识两个新符号，"\?"与"\+"

\?

表示匹配其前面的字符0或1次，换句话说，就是前面的字符要么没有，要么有一个。

\+表示匹配其前面的字符至少1次，换句话说，就是前面的字符必须有至少一个。

我们来看看示例，如下。

```
[www.zsythink.net]#grep --color "abc\?" regex.txt
ab abc abcc
[www.zsythink.net]#
```

如上图所示，"c\?"表示c出现0次或者1次，都会被匹配到，所以ab和abc都被匹配到了，ab被匹配到是因为c出现了0次，abc被匹配到是因为c出现了1次。

看完上述示例后，再看另外一个例子，如下：

```
[www.zsythink.net]#grep --color "abc\+" regex.txt
ab abc abcc
[www.zsythink.net]#
```

可以看到，abc与abcc都被匹配到了，这是因为"c\+"表示c至少要出现1次，至多可以连续出现多次，连续次数上不封顶，所以abc和abcc都会被匹配到。

好了，关于"连续次数匹配"的相关正则表达式就总结到这里吧。

## 小结

为了方便以后回顾，我们将上述正则总结如下。

- 1 \* 表示前面的字符连续出现任意次，包括0次。
- 2 . 表示任意单个字符。
- 3 .\* 表示任意长度的任意字符，与通配符中的\*的意思相同。
- 4 \? 表示匹配其前面的字符0或1次
- 5 \+ 表示匹配其前面的字符至少1次，或者连续多次，连续次数上不封顶。
- 6 \{n\} 表示前面的字符连续出现n次，将会被匹配到。
- 7 \{x,y\} 表示之前的字符至少连续出现x次，最多连续出现y次，都能被匹配到，换句话说，只要之前的字符连续出现的次数在x与y之间，即可被匹配到。
- 8 \{,n\} 表示之前的字符连续出现至多n次，最少0次，都会匹配到。
- 9 \{n,\} 表示之前的字符连续出现至少n次，才会被匹配到。

希望这篇文章能够帮助到你，下次再见哦~~亲~~~

