正则表达式

电子系科协软件部
TUNA协会

课前准备：确认你的编辑器搜索/替换能用正则表达式
或者下载
Sublime Text 3
sublimetext.com/3
或 Atom
atom.io
或者拿出纸和笔orz

fuli.eesast.com

李思涵
lisihan969@gmail.com

2015-2016软件部讲座分享交流群

This group QR Code will expire on 2015-10-24

2015-2016软件福利讲座二群

This group QR Code will expire on 2015-10-24

正则表达式

电子系科协软件部

李思涵
lisihan969@gmail.com

什么是正则表达式?

Regular expression

什么是正则表达式?

Regular expression

Regex

正则表达式 / Pattern

(?<=\.) {2,}(?=[A-Z])

匹配

本质：定义一种语法，描述了一系列字符串

什么是正则表达式?

Regular expression

Regex

正则表达式 / Pattern

(?<=\.) {2,}(?=[A-Z])

匹配

为什么用它?

wildcard　　re*l?r

描述过于简单，不适合复杂匹配

xxxxx_8或9位数字.xxx

在哪里能用它?

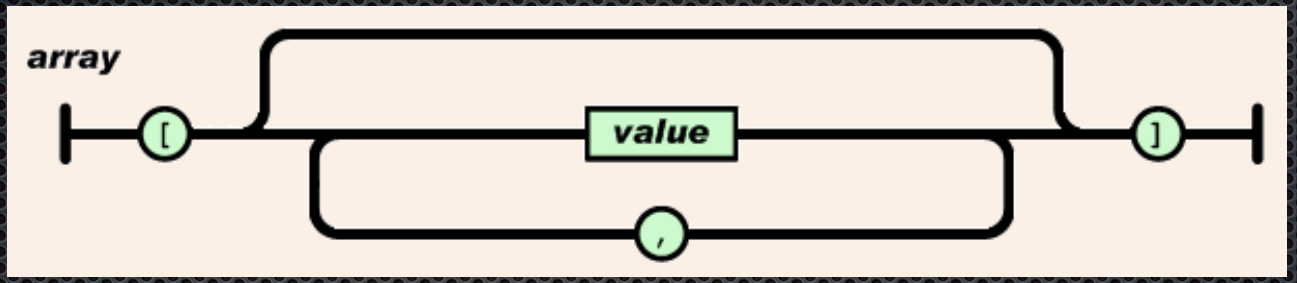
	内置:	标准库:
编程时	JavaScript Ruby Perl	C++ (从C++11起) Python Java .NET

日常　　一个正常的代码编辑器/命令行工具

grep, sed, …

语法图 syntax diagram

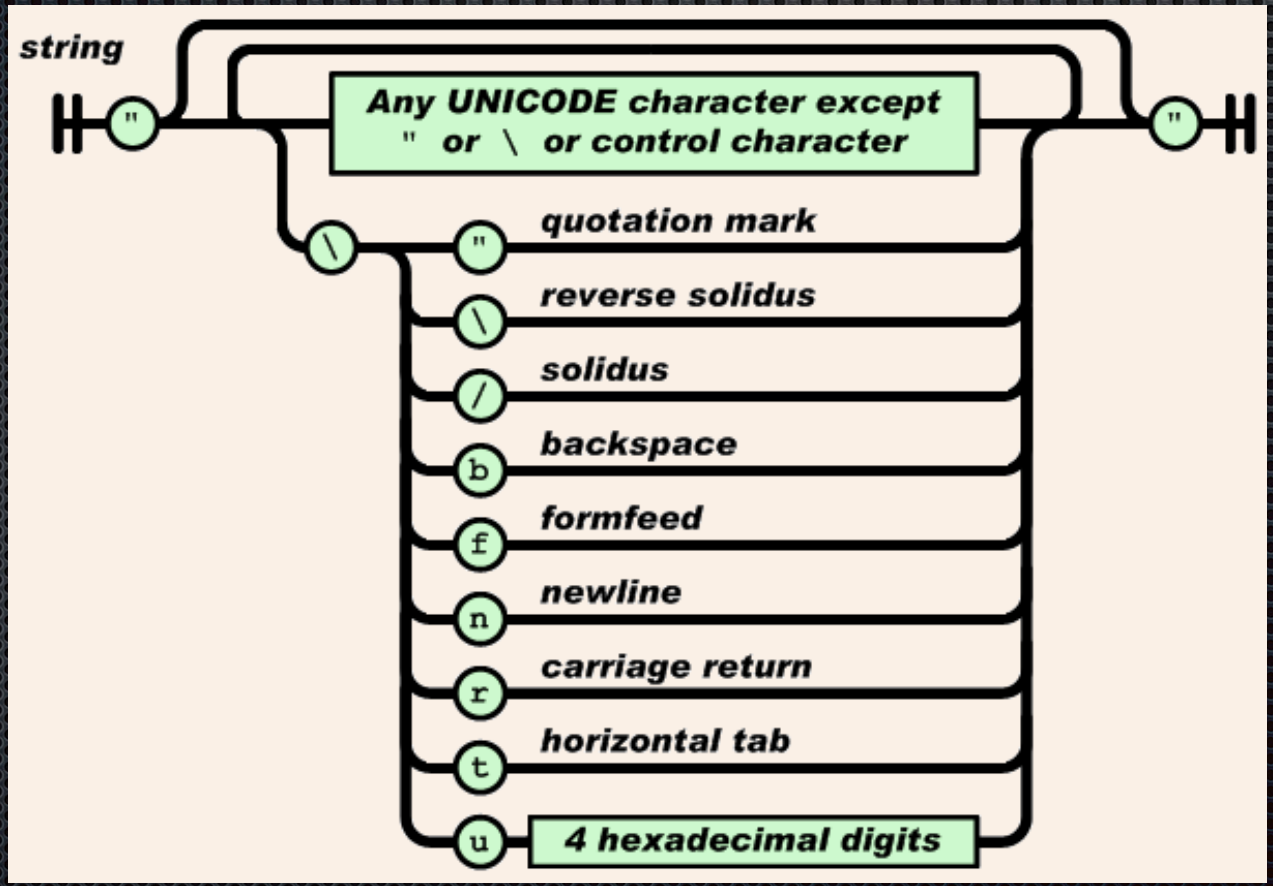
```
[value1, value2, value3, …]
```



1. 左边界到右边界
2. 圆框字面量，方框描述
3. 只有沿轨道走合法
4. 两端双线不能插空白，双线不能

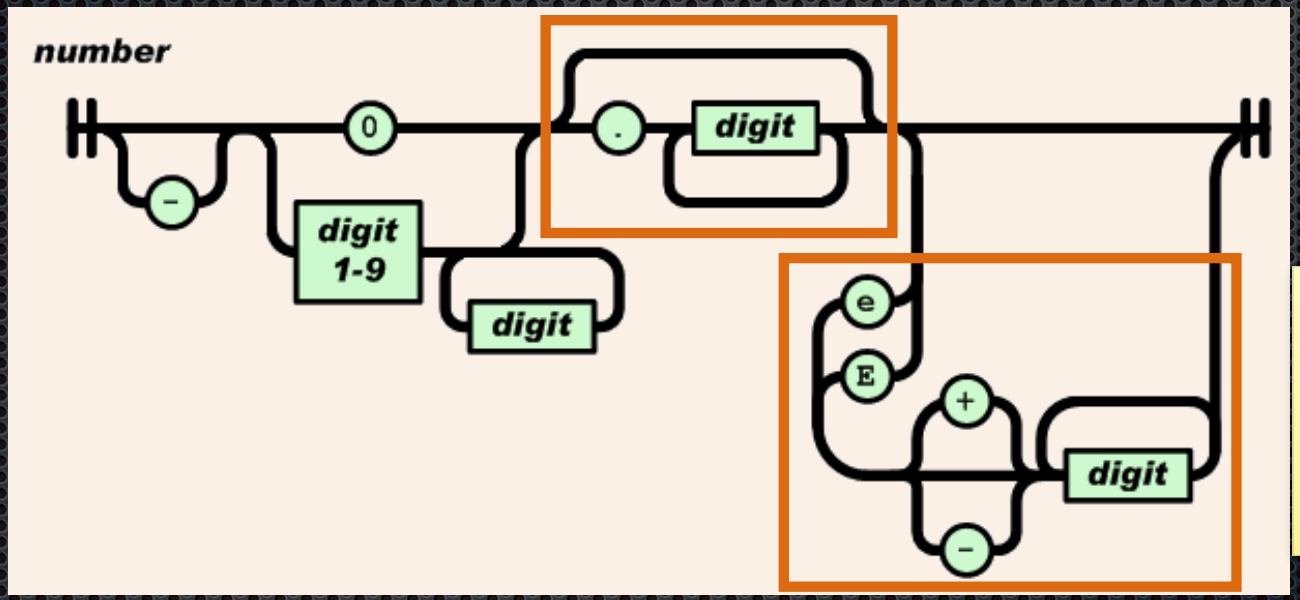
语法图 syntax diagram

```
"Hello World!\n"
```



语法图 syntax diagram

小数部分



指数部分

基本语法

- 元字符 Metacharacter（用\转义）
- 正常字符

. 任意字符

一般不包括 '\n'

. 任意字符

[ ] 任意一个

]在第一个不用转义
-在第一或最后不用转义

[qwerty]

[^qwerty]

[a-zA-Z]

注意：[.] 匹配的是 .

. 任意字符
[] 任意一个

^ 行首

注意：长度为0

. 任意字符
[] 任意一个
^ 行首

$ 行尾

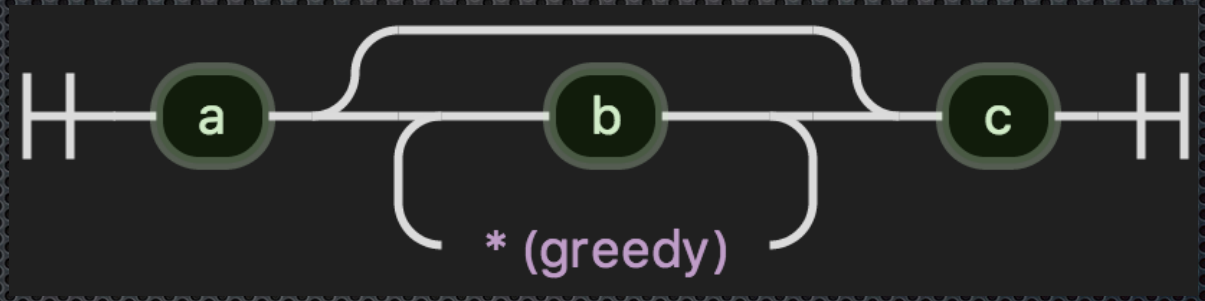
注意：长度为0

. 任意字符
[] 任意一个
^ 行首 $ 行尾

* 0次或更多次

针对前一个元素

ab*c　　ac, abc, abbc, abbbc, ...



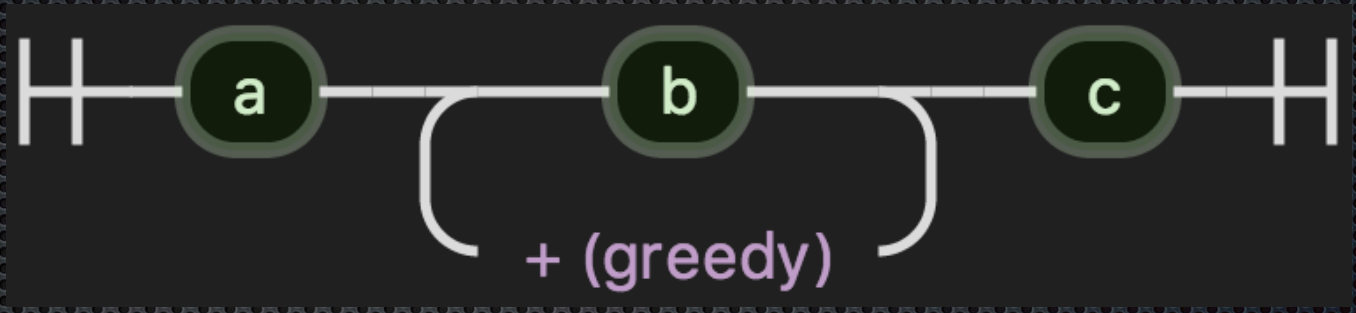
ab*　　abbbbbbbbb　　贪婪

ab*?　　abbbbbbbbb　　懒惰

+ 1次或更多次

. 任意字符
[] 任意一个
^ 行首 $ 行尾
* 0次或更多次

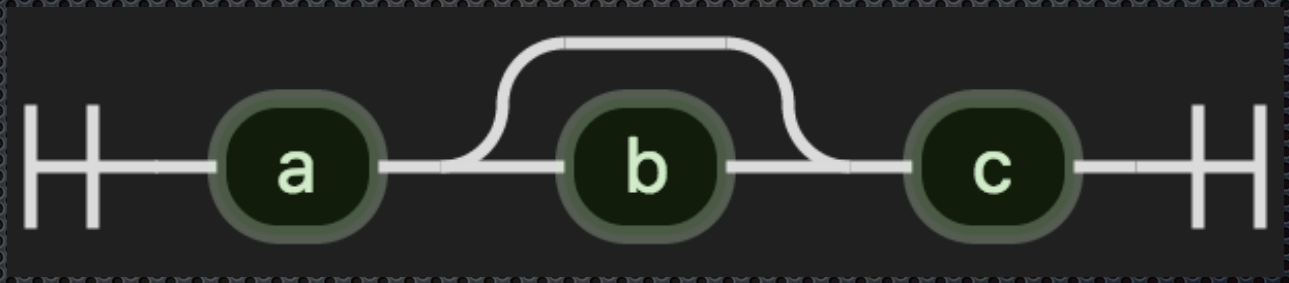
ab+c　　abc, abbc, abbbc, ...



? 0次或1次

. 任意字符
[] 任意一个
^ 行首 $ 行尾
* 0次或更多次
+ 1次或更多次

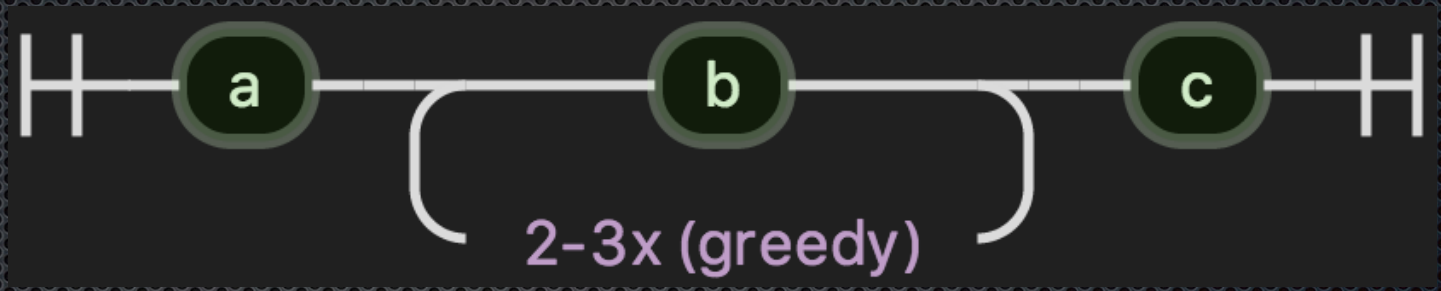
ab?c　　ac, abc



{m,n} m到n次

. 任意字符
[] 任意一个
^ 行首 $ 行尾
* 0次或更多次
+ 1次或更多次
? 0次或1次

ab{2, 3}c　　abbc, abbbc



{m} 恰好 m 次
{m,} 至少 m 次

() 子表达式

. 任意字符
[] 任意一个
^ 行首 $ 行尾
* 0次或更多次
+ 1次或更多次
? 0次或1次
{m,n} m到n次

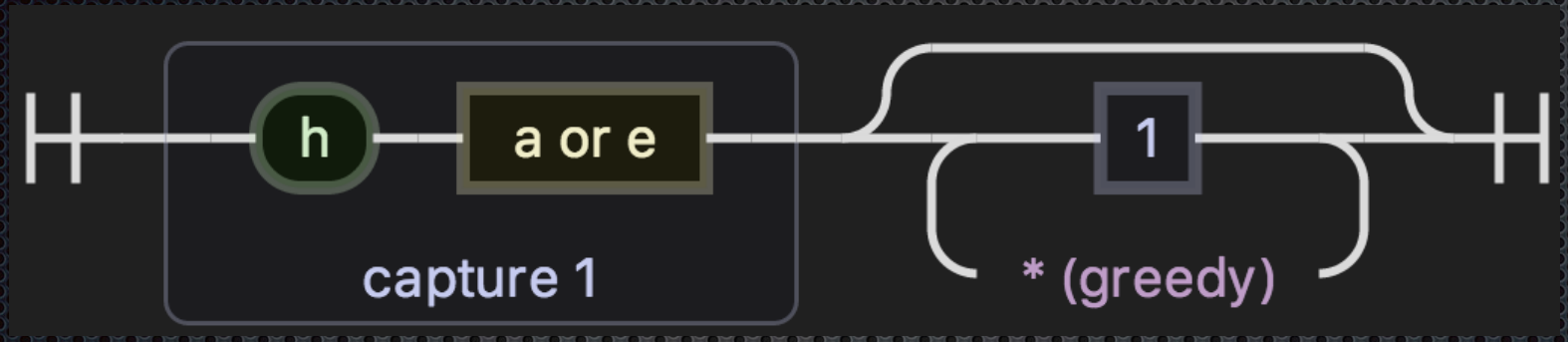
(h[ae])+ ha, hehe

hahehahe?

\1, \2, ...引用子表达式

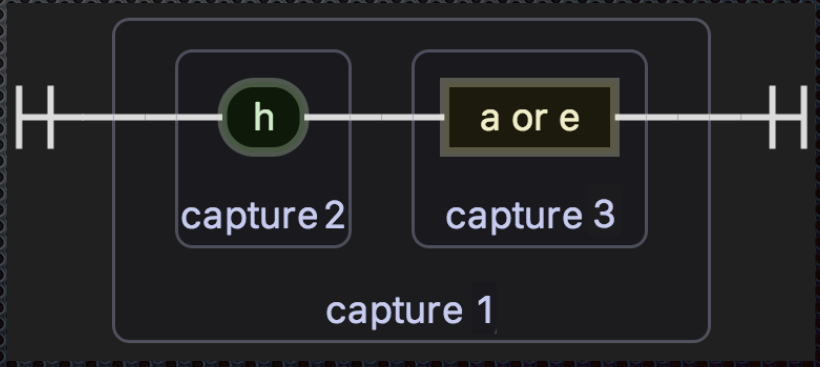
9个以上，有的实现支持

(h[ae])\1*　　ha, hehe, hanana



. 任意字符
[] 任意一个
^ 行首 $ 行尾
* 0次或更多次
+ 1次或更多次
? 0次或1次
{m,n} m到n次
() 子表达式

嵌套：((h)([ae]))
按左括号的次序

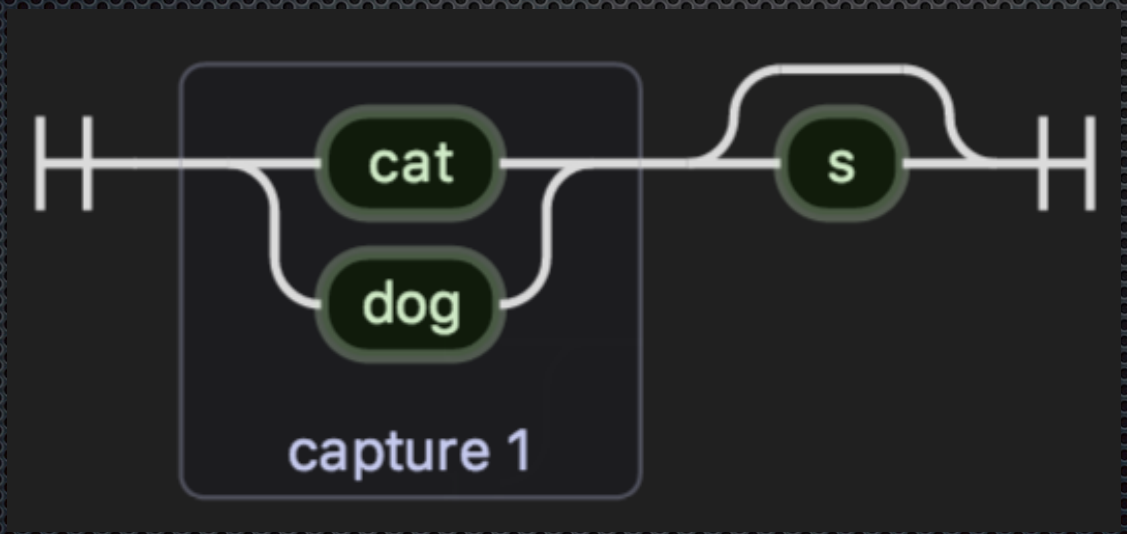


p过，工具有问题

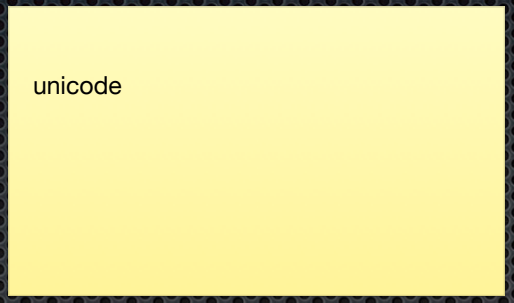
| 前面或后面的表达式

(cat|dog)s?　　cat, cats, dog, dogs

. 任意字符
[] 任意一个
^ 行首 $ 行尾
* 0次或更多次
+ 1次或更多次
? 0次或1次
{m,n} m到n次
() 子表达式
\1, \2, ...引用子表达式



其他




```
\d      [0-9]
\w      [A-Za-z0-9_]
\s      [ \t\r\n\v\f]
\b      word boundary
```

(无长度，前面是\w后面不是，
或后面 \w 前面不是)

```
\D      [^0-9]
\W      [^A-Za-z0-9_]
\S      [^ \t\r\n\v\f]
```

. 任意字符
[] 任意一个
^ 行首 $ 行尾
* 0次或更多次
+ 1次或更多次
? 0次或1次
{m,n} m到n次
() 子表达式
\1, \2, ...引用子表达式
| 前面或后面的表达式

小练习

请大家在纸上/电脑上写下能匹配ipv4 地址的正则表达式

匹配

- 255.255.255.255
- 192.168.0.0
- 0.0.0.0
- 01.02.003.004

不匹配

- 256.255.255.255
- 0001.0.0.0

. 任意字符
[] 任意一个
^ 行首 $ 行尾
* 0次或更多次
+ 1次或更多次
? 0次或1次
{m,n} m到n次
() 子表达式
\1, \2, ...引用子表达式
| 前面或后面的表达式

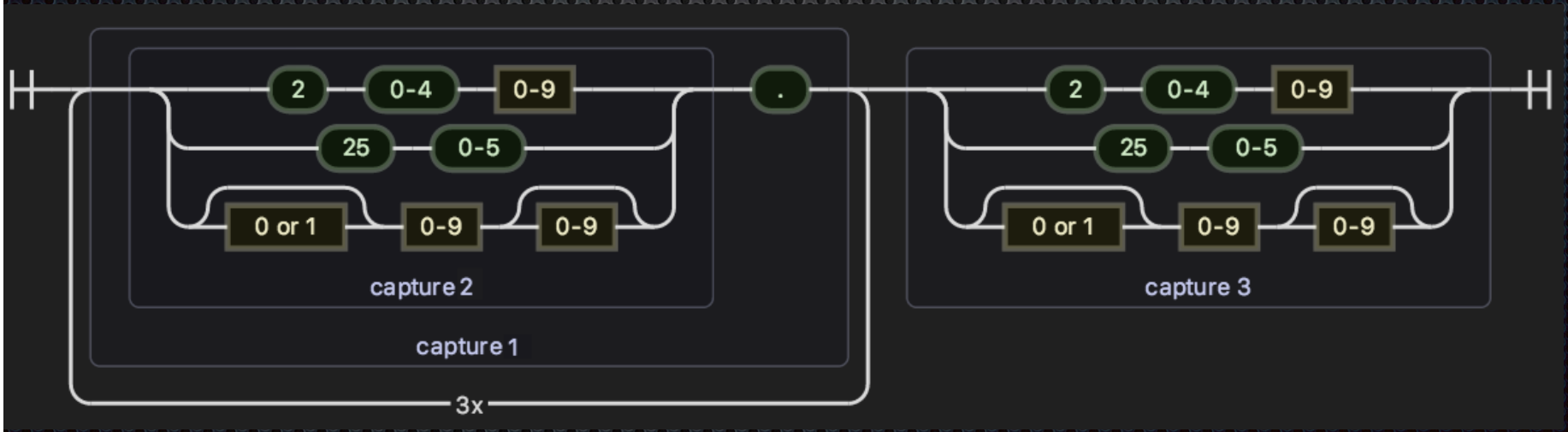
```
\d    [0-9]
\w    [A-Za-z0-9_]
\s    [ \t\r\n\v\f]
\b    word boundary
```

小练习

((2[0-4]\d|25[0-5]|[01]?\d\d?)\.){3}
(2[0-4]\d|25[0-5]|[01]?\d\d?)

关键：(2[0-4]\d|25[0-5]|[01]?\d\d?)

. 任意字符
[] 任意一个
^ 行首 $ 行尾
* 0次或更多次
+ 1次或更多次
? 0次或1次
{m,n} m到n次
() 子表达式
\1, \2, ...引用子表达式
| 前面或后面的表达式



小练习

网络学堂下载文件的文件，已知后缀数字 8-9 位

匹配

- 《工程制图》MOOC课程选课说明_4256027 34.pdf
- Calibration_393802155.pdf
- 03连续信源数字化_40101875.pdf

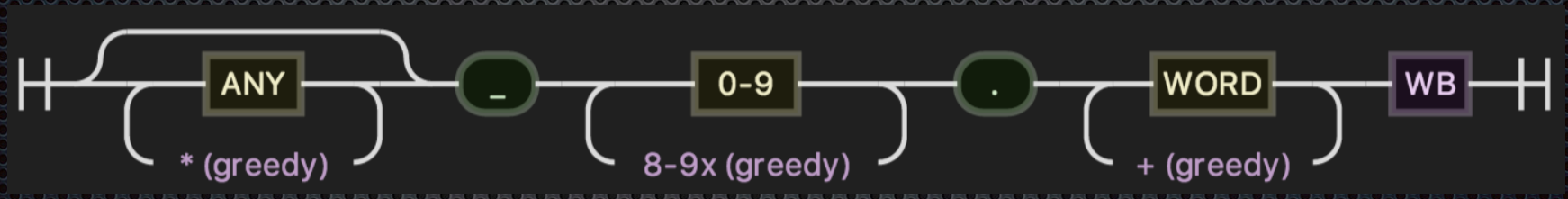
. 任意字符
[] 任意一个
^ 行首 $ 行尾
* 0次或更多次
+ 1次或更多次
? 0次或1次
{m,n} m到n次
() 子表达式
\1, \2, ...引用子表达式
| 前面或后面的表达式

\d	[0-9]
\w	[A-Za-z0-9_]
\s	[\t\r\n\v\f]
\b	word boundary

小练习

.*_\d{8,9}\.\w+\b

. 任意字符
[] 任意一个
^ 行首 $ 行尾
* 0次或更多次
+ 1次或更多次
? 0次或1次
{m,n} m到n次
() 子表达式
\1, \2, ...引用子表达式
| 前面或后面的表达式



进阶

Look-ahead / Look-behind

(?=pattern)
后面匹配 pattern
(?!pattern)
后面不匹配 pattern
(?<=pattern)
前面匹配 pattern
(?<!pattern)
前面不匹配 pattern

不一定支持

例如：最后一个 hehe

进阶

Look-ahead / Look-behind

(?=pattern)
后面匹配 pattern

(?!pattern)
后面不匹配 pattern

(?<=pattern)
前面匹配 pattern

(?<!pattern)
前面不匹配 pattern

hehe(?!.*hehe)

例如：最后一个 hehe

每当我学会一项新技能，我总是幻想出极其复杂的场景，使得这项技能让我危难时刻显身手。

每当我学会一项新技能，我总是幻想出极其复杂的场景，使得这项技能让我危难时刻显身手。

哦闹！那杀手一定是跟踪她去度假地了！

每当我学会一项新技能，我总是幻想出极其复杂的场景，使得这项技能让我危难时刻显身手。

哦闹！那杀手一定是跟踪她去度假地了！

可是要找到她，我们得搜寻两百多兆的电子邮件，在里面寻找看起来像地址一样的东西！

我们毫无希望！

每当我学会一项新技能，我总是幻想出极其复杂的场景，使得这项技能让我危难时刻显身手。

哦闹！那杀手一定是跟踪她去度假地了！

可是要找到她，我们得搜寻两百多兆的电子邮件，在里面寻找看起来像地址一样的东西！

我们毫无希望！

所有人都让开。

每当我学会一项新技能，我总是幻想出极其复杂的场景，使得这项技能让我危难时刻显身手。

哦闹！那杀手一定是跟踪她去度假地了！

可是要找到她，我们得搜寻两百多兆的电子邮件，在里面寻找看起来像地址一样的东西！

我们毫无希望！

所有人都让开。

我知道正则表达式。

每当我学会一项新技能，我总是幻想出极其复杂的场景，使得这项技能让我危难时刻显身手。

哦闹！那杀手一定是跟踪她去度假地了！

可是要找到她，我们得搜寻两百多兆的电子邮件，在里面寻找看起来像地址一样的东西！

我们毫无希望！

所有人都让开。

我知道正则表达式。

每当我学会一项新技能，我总是幻想出极其复杂的场景，使得这项技能让我危难时刻显身手。

哦闹！那杀手一定是跟踪她去度假地了！

可是要找到她，我们得搜寻两百多兆的电子邮件，在里面寻找看起来像地址一样的东西！

我们毫无希望！

所有人都让开。

我知道正则表达式。

每当我学会一项新技能，我总是幻想出极其复杂的场景，使得这项技能让我危难时刻显身手。

哦闹！那杀手一定是跟踪她去度假地了！

可是要找到她，我们得搜寻两百多兆的电子邮件，在里面寻找看起来像地址一样的东西！

我们毫无希望！

所有人都让开。

我知道正则表达式。

原作xkcd 汉化Ent