TURING 图灵程序设计丛书

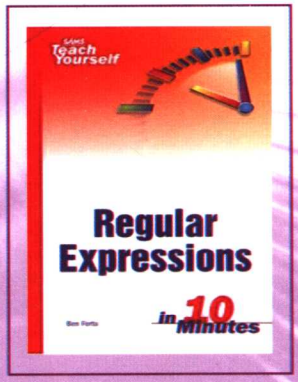
SAMS

Sams Teach Yourself Regular Expressions in 10 Minutes

正则表达式必知必会

[美] Ben Forta 著
杨涛 等译

- 正则表达式经典著作
- 轻松迅捷的学习环境
- 示例丰富，涵盖所有主流平台



人民邮电出版社
POSTS & TELECOM PRESS

TURING 图灵程序设计丛书

正则表达式必知必会

Sams Teach Yourself Regular Expressions in 10 Minutes

[美] Ben Forta 著
杨涛 等译

人民邮电出版社
北 京

图书在版编目（CIP）数据

正则表达式必知必会 /（美）福达（Forta,B.）著；
杨涛等译. —北京：人民邮电出版社，2007.12
（图灵程序设计丛书）
ISBN 978-7-115-16474-2

Ⅰ. 正… Ⅱ. ①福…②杨… Ⅲ. 正则表达式—教材
Ⅳ. TP301.2

中国版本图书馆 CIP 数据核字（2007）第 096517 号

内 容 提 要

正则表达式是一种威力无比强大的武器，几乎在所有的程序设计语言里和计算机平台上都可以用它来完成各种复杂的文本处理工作。本书从简单的文本匹配开始，循序渐进地介绍了很多复杂内容，其中包括回溯引用、条件性求值和前后查找，等等。每章都为读者准备了许多简明又实用的示例，有助于全面、系统、快速掌握正则表达式，并运用它们去解决实际问题。

本书适合各种语言和平台的开发人员。

图灵程序设计丛书

正则表达式必知必会

◆ 著 ［美］Ben Forta
译 杨 涛 等
责任编辑 陈兴璐

◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 http://www.ptpress.com.cn
北京顺义振华印刷厂印刷
新华书店总店北京发行所经销

◆ 开本：800×1230 1/32
印张：4.75
字数：146 千字 2007 年 12 月第 1 版
印数：1～5 000 册 2007 年 12 月北京第 1 次印刷
著作权合同登记号 图字：01-2007-1961 号

ISBN 978-7-115-16474-2/TP

定价：29.00 元

读者服务热线：(010)88593802 印装质量热线：(010)67129223

版 权 声 明

引　言

正则表达式（regular expression）和正则表达式语言已经出现很多年了。正则表达式的专家们早就掌握了这种威力无比强大的武器，它可以用来完成各种复杂的文本处理工作。更重要的是，这种武器可以在几乎所有的程序设计语言里和几乎所有的计算机平台上使用。

这是个好消息，但我还要告诉你一个坏消息：长期以来，只有一些真正的专家才能真正掌握正则表达式。甚至有很多人根本没有听说过正则表达式这个概念，更不用说用它们来解决问题了。至于少数勇于涉猎正则表达式领域的人们，又往往会因为正则表达式难以理解而浅尝辄止或总是在原地徘徊。这不能不说是一种悲哀，因为正则表达式其实并没有人们想像中的那么复杂。只要你能清晰地理解你想要解决的问题并学会如何使用正则表达式，就可以轻而易举地解决这些问题。

正则表达式不为大多数人所掌握的原因之一是关于这方面的好资料太少了。虽然有很多网站在吹嘘它们的正则表达式教程如何全面，但实际情况却是高质量的正则表达式学习资源相当稀缺。即便能够找到几本介绍正则表达式的书籍，它们又往往过于偏重语法而显得不够实用——知道如何定义{或是知道+与*之间的区别并不等于真正掌握了正则表达式的用法。在笔者看来，那些书籍反而把简单的问题弄得更复杂了：在学习和使用正则表达式的时候，重要的并不是你知道多少个特殊字符，而是你会不会运用它们去解决实际问题。

你拿在手里的这本书并不打算成为一本正则表达式的大全。如果你想要的是那样一本书，你应该去阅读Jeffrey Friedl编写的*Mastering Regular Expressions*（O'Reilly出版公司，ISBN 0596002890）。Friedl先生是业内公认的正则表达式专家，他的书绝对是这方面最权威和全面的著

作。本人对Friedl先生没有丝毫成见，但他的书不适合初学者也是实情；如果你只打算尽快完成手头的工作而不是要钻研正则表达式的内部原理的话，他的书也不很适用。这并不是说那本书里的信息没有用，只是它在你想要给HTML表单添加一些验证功能或者只想对解析的文本进行替换的时候派不上什么用场。如果你想尽快学会正则表达式的基本用法，你将发现自己陷入了一个两难境地：要么找不到简明易学的参考资料，要么找到的参考资料过于深奥而让你不知该如何起步。

这正是促使笔者编写本书的原因。本书所讲授的关于正则表达式知识正是你们在刚起步时最需要的，我们将从简单的文本匹配开始循序渐进地向大家介绍许多复杂的专题，其中包括回溯引用（backreference，或译为后向引用）、条件性求值（conditional evaluation）和前后查找（looking-around），等等。本书最大的优势是所学到的知识可以立即运用于实践中：我们在每章里都为大家准备了许多简明又实用的示例，它们可以帮助你全面、系统、快速地掌握正则表达式并运用它们去解决实际问题，而每章在10分钟甚至更短的时间里就可以学完。

还等什么，赶快翻到第1章开始今天的学习吧，你肯定会立刻感受到正则表达式的强大威力。

目标读者

本书的目标读者是以下几类人员：

- 第一次接触正则表达式。
- 希望自己能够快速掌握正则表达式的基本用法。
- 想使用一种强大的工具（虽然它不那么容易掌握）去解决实际问题。
- 正在开发Web应用软件并需要进行复杂的表单和文本处理。
- 正使用着Perl、ASP、Visual Basic、.NET、C#、Java、JSP、PHP、ColdFusion语言（或更多其他程序设计语言），希望在开发的应用程序里使用正则表达式。
- 希望在不求助于其他人的前提下尽快掌握正则表达式。

致谢

首先，我要感谢正则表达式专家和我以前的合作者Michael Dinowitz，他对本书的技术细节进行了严格的审校并提供了许多宝贵的意见和反馈。

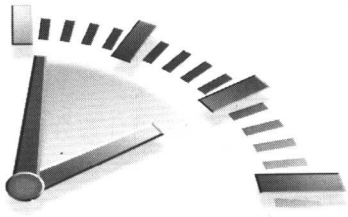
本书的附录C向大家介绍了一种基于Web的正则表达式测试器，而我必须在此感谢这个测试器的原始作者Nate Weiss（它最初是为*ColdFusion Web Application Construction Kit*一书而编写的）。在Nate的许可和支持下，我对他用ColdFusion编写的正则表达式测试软件进行了改写以配合本书使用，开发了相应的JavaScript版本。感谢Qasim Rasheed为这个测试器编写ASP和JSP版本，感谢Scott Van Vliet为这个测试器编写ASP.NET版本。

最后，我还要感谢Sams出版公司里帮助我把本书从概念变成现实的人们，尤其是Michael Stephens和Mark Renfrow。没有他们的帮助和支持，本书是不可能与大家见面的。

谢谢大家。

——Ben Forta

目　录



第1章 正则表达式入门

在本章里，你将学习何为正则表达式以及它们可以帮助你做些什么。

1.1 正则表达式的用途

正则表达式（regular expression，简称regex）是一种工具，和其他工具一样，它是人们为了解决某一类专门的问题而发明的。要想理解正则表达式及其功用，最好的办法是了解它们可以解决什么样的问题。

请考虑以下几个场景：

- 你正在搜索一个文件，这个文件里包含着单词car（不区分字母大小写），但你并不想把包含着字符串car的其他单词（比如scar、carry和incarcerate，等等）也找出来。
- 你打算用一种应用服务器来动态地生成一个Web网页以显示从某个数据库里检索出来的文本。在那些文本里可能包含着一些URL地址字符串，而你希望那些URL地址在最终生成的页面里是可点击的（也就是说，你打算生成一些合法的HTML代码——<A HREF></A>——而不仅仅是普通的文本）。
- 你创建了一份包含着一张表单的Web页面，这张表单用来收集用户信息，其中包括一个电子邮件地址。你需要检查用户给出的电子邮件地址是否符合正确的语法格式。
- 你正在编辑一段源代码并且要把所有的size都替换为isize，但这种替换仅限于单词size本身而不涉及那些包含着字符串size的其他单词。

- 你正在显示一份计算机文件系统中所有文件的清单，但你只想把文件名里包含着Application字样的文件列举出来。
- 你正在把一些数据导入应用程序。那些数据以制表符作为分隔符，但你的应用程序要支持CSV格式（每条记录独占一行，同一条记录里的各项数据之间用逗号分隔并允许被括在引号里面）。
- 你需要在文件里搜索某个特定的文本，但你只想把出现在特定位置的（比如每行的开头或是每条语句的结尾）找出来。

以上场景都是大家在编写程序时经常会遇到的问题，用任何一种支持条件处理和字符串操作的编程语言都可以解决它们——但问题是你的解决方案将会变得十分复杂。比较容易想到的办法是，用一些循环来依次遍历那些单词或字符并在循环体里面用一系列 if 语句来进行测试，这往往意味着你需要使用大量的标志来标记你已经找到了什么，你还没有找到什么，还需要检查空白字符和特殊字符，等等。而这一切都需要以手工方式来进行。

另一种解决方案是使用正则表达式。上述问题都可以用一些精心构造的语句——或者说一些由文本和特殊指令构成的高度简练的字符串来解决，比如像下面这样的语句：

```
\b[Cc][Aa][Rr]\b
```

注意　如果你现在还看不懂这一行，先别着急。你很快就会知道它的含义是什么。

1.2　如何使用正则表达式

如果认真思考一下那些问题场景，你就会发现它们不外乎两种情况：一种是查找特定的信息（搜索），另一种是查找并编辑特定的信息（替换）。事实上，从根本上来讲，那正是正则表达式的两种基本用途：搜索和替换。给定一个正则表达式，它要么匹配一些文本（进行一次搜索），要么匹配并替换一些文本（进行一次替换）。

1.2.1 用正则表达式进行搜索

正则表达式的主要用途之一是搜索变化多端的文本，比如刚才描述的搜索单词car的场景：你要把car、CAR、Car，或CaR都找出来，但这只是整个问题比较简单的一部分（有许多搜索工具都可以完成不区分字母大小写的搜索）。比较困难的部分是确保scar、carry和incarcerate之类的单词不会被匹配到。一些比较高级的编辑器提供了“Match Only Whole Word（仅匹配整个单词）”选项，但还有许多编辑器并不具备这一功能，而你往往无法在你正在编辑的文档里做出这种调整。使用正则表达式而不是纯文本car进行搜索就可以解决这个问题。

提示 想知道如何解决这个问题吗？你们其实已经见过答案了——它就是我们刚才给出的示例语句：\b[Cc][Aa][Rr]\b

请注意，“等于”比较（比如说，用户给出的电子邮件地址是否匹配这个正则表达式？）本质上也是一种搜索操作，这种搜索操作会对用户所提供的整个字符串进行搜索以寻找一个匹配。与此相对的是子字符串搜索，子字符串搜索是“搜索”这个词的普通含义。

1.2.2 用正则表达式进行替换

正则表达式搜索的威力非常强大，非常有用，而且比较容易学习和掌握。本书的许多章节和示例都与“匹配”有关。不过，正则表达式的真正威力体现在替换操作方面，比如我们刚才所描述的需要把URL地址字符串替换为可点击URL地址的场景：这需要先把相关文本里的URL地址字符串找出来（比如说，通过搜索以http://或https://开头、以句号、逗号或空白字符结尾的字符串），再把找到的URL地址字符串替换为HTML语言的“<A HREF=…> … </A>”元素，如下所示：


```
http://www.forta.com/
```

替换结果：

```
<A HREF="http://www.forta.com">http://www.forta.com/</A>
```

绝大多数应用程序的“Search and Replace”（搜索和替换）选项都可

以完成这种替换操作，但使用一个正则表达式来完成这个任务将简单得让人难以置信。

1.3　什么是正则表达式

现在，你已经知道正则表达式是用来干什么的了，我们再来给它下个定义。简单地说，正则表达式是一些用来匹配和处理文本的字符串。正则表达式是用正则表达式语言创建的，这种语言的用途就是为了解决我们前面所描述的种种问题。与其他程序设计语言一样，正则表达式语言也有需要你们去学习的特殊语法和指令，它们正是本书要教给大家的东西。

正则表达式语言并不是一种完备的程序设计语言，它甚至算不上是一种能够直接安装并运行的程序。更准确地说，正则表达式语言是内置于其他语言或软件产品里的“迷你”语言。好在现在几乎所有的语言或工具都支持正则表达式，但是正则表达式与你正在使用的语言或工具可以说毫无相似之处。正则表达式语言虽然也被称为一种语言，但它与人们对语言的印象相去甚远。

注意　正则表达式起源于1950年代在数学领域的一些研究工作。几年之后，计算机领域借鉴那些研究工作的成果和思路开发出了Unix世界里的Perl语言和grep等工具程序。在许多年里，正则表达式只流行于Unix平台（Unix程序员用它们来解决我们前面所描述的各种问题），但这种情况早已发生了变化，现在几乎所有的计算平台都支持正则表达式，只是具体方式和支持程度略有差异而已。

说完这些掌故，我们再来看几个例子。下面都是合法的正则表达式（我们稍后再解释它们的用途）：

- `Ben`
- `.`
- `www\ .forta\ .com`

- [a-zA-Z0-9_.]*
- <[Hh]1>.*</[Hh]1>
- \r\n\r\n
- \d{3,3}-\d{3,3}-\d{4,4}

请注意，语法是正则表达式最容易掌握的部分，真正的挑战是学会如何运用那些语法把实际问题分解为一系列正则表达式并最终解决。与学习其他程序设计语言一样，只靠读书是学不会如何灵活运用语法正则的，你必须通过亲身实践才能真正掌握它们。

1.4 使用正则表达式

正如前面解释的那样，不存在所谓的正则表达式程序；它既不是可以直接运行的应用程序，也不是可以从哪里购买或下载来的软件。在绝大多数的软件产品、编程语言、工具程序和开发环境里，正则表达式语言都已被实现。

正则表达式的使用方法和具体功能，在不同的应用程序/语言中各有不同。一般来说，应用程序大多使用菜单选项和对话框来访问正则表达式，而程序设计语言大都在函数或对象类中使用正则表达式。

此外，并非所有的正则表达式实现都是一样的。在不同的应用程序/语言里，正则表达式的语法和功能往往会有明显（有时也不那么明显）的差异。

附录A对支持正则表达式的许多应用程序和语言在这方面的细节进行了汇总。在继续学习下一章之前，你应该先熟悉一下附录A，看看你们正在使用的应用程序或语言在正则表达式方面都有哪些与众不同之处。

为了帮助大家尽快入门，我们在这本书的配套网页http：//www.forta.com/books/0672325667/上准备了一个名为“Regular Expression Tester（正则表达式测试器）”的工具软件供大家下载。这个基于Web的工具软件有好几种版本，它们分别对应着一些比较流行的应用服务器和编程语言，还有一个版本是专门用来直接测试用JavaScript语言编写出来的正则表达

式的。附录C对这个工具软件的用法进行了介绍，这个工具可以简便、快速地对你们构造出来的正则表达式进行测试，这对大家的学习肯定会有很大的帮助。

1.5　在继续学习之前

在继续学习之前，你还应该了解以下几个事实：

- 在使用正则表达式的时候，你将发现几乎所有的问题都有不止一种解决方案。它们有的比较简单，有的比较快速，有的兼容性更好，有的功能更全。这么说吧，在编写正则表达式的时候，只有对、错两种选择的情况是相当少见的——同一个问题往往会有多种解决方案。
- 正如我们前面讲过的那样，正则表达式的不同实现往往会有所差异。在编写本书的时候，我们已尽了最大努力来保证各章里的示例能适用于尽可能多的实现；但有些差异和不兼容是无法回避的，我们针对这种情况都尽可能地进行了注明。
- 与其他程序设计语言一样，学习正则表达式的关键是实践，实践，再实践。

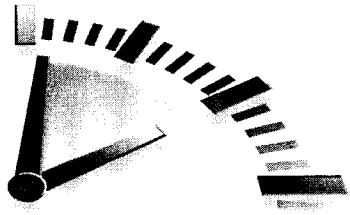
8

注意　我们强烈建议大家在学习本书的过程中能够亲自实践每一个示例。

1.6　小结

正则表达式是文本处理方面功能最强大的工具之一。正则表达式语言用来构造正则表达式（最终构造出来的字符串就称为正则表达式），正则表达式用来完成搜索和替换操作。

9



第2章

匹配单个字符

在本章里，你将学习如何对一个或多个字符进行简单的字符匹配。

2.1 匹配纯文本

Ben是一个正则表达式。因为本身是纯文本，所以看起来可能不像是一个正则表达式，但它的确是。正则表达式可以包含纯文本（甚至可以只包含纯文本）。当然，像这样使用正则表达式是一种浪费，但把它作为我们学习正则表达式的起点还是很不错的。

我们来看一个例子：

文本

```
Hello, my name is Ben. Please visit
my website at http://www.forta.com/.
```

正则表达式

```
Ben
```

结果

```
Hello, my name is Ben. Please visit
my website at http://www.forta.com/.
```

分析

这里使用的正则表达式是纯文本，它将匹配原始文本里的Ben。

我们再来看一个例子，它使用了与刚才相同的原始文本和另外一个正则表达式：