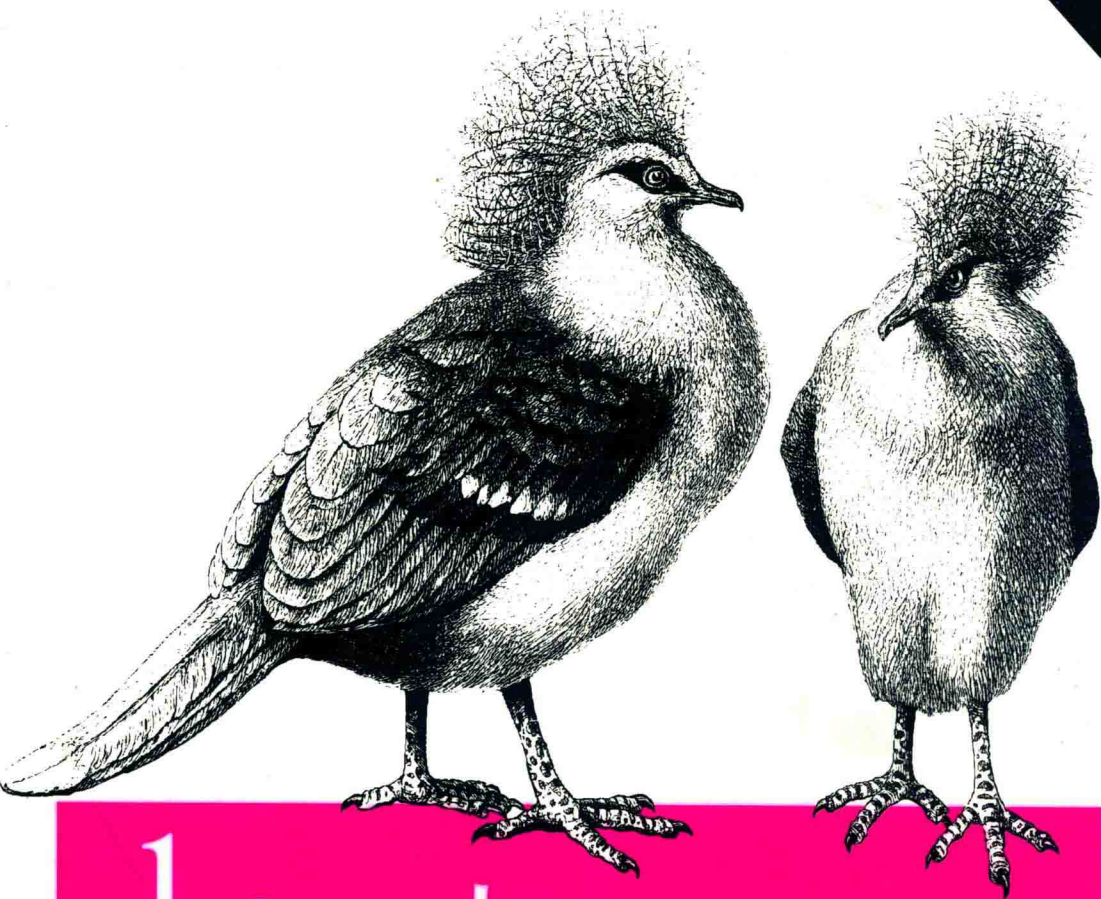


lex & yacc

第二版



lex与yacc

O'REILLY®



机械工业出版社
China Machine Press



John R. Levine, Tony Mason
& Doug Brown 著

杨作梅 张旭东 等译

lex 与 yacc

第二版

John R. Levine, Tony Mason & Doug Brown 著

杨作梅 张旭东 等译

O'REILLY®

Beijing • Cambridge • Farnham • Köln • Paris • Sebastopol • Taipei • Tokyo

O'Reilly & Associates, Inc. 授权机械工业出版社出版

机械工业出版社

图书在版编目 (CIP) 数据

lex 与 yacc (第二版) / (美) 莱弗恩 (Levine, J. R.) 等著; 杨作梅等译. - 北京: 机械工业出版社, 2003.1

书名原文: lex & yacc, Second Edition

ISBN 7-111-10721-7

I. I... II. ①莱... ②杨... III. UNIX 操作系统 - 程序设计 IV. TP316.81

中国版本图书馆 CIP 数据核字 (2002) 第 060101 号

北京市版权局著作权合同登记

图字: 01-2002-1829 号

©1992 by O'Reilly & Associates, Inc.

Simplified Chinese Edition, jointly published by O'Reilly & Associates, Inc. and China Machine Press, 2003. Authorized translation of the English edition, 1992 O'Reilly & Associates, Inc., the owner of all rights to publish and sell the same.

All rights reserved including the rights of reproduction in whole or in part in any form.

英文原版由 O'Reilly & Associates, Inc. 出版 1992。

简体中文版由机械工业出版社出版 2003。英文原版的翻译得到 O'Reilly & Associates, Inc. 的授权。此简体中文版的出版和销售得到出版权和销售权的所有者 —— O'Reilly & Associates, Inc. 的许可。

版权所有, 未得书面许可, 本书的任何部分和全部不得以任何形式重制。

书 名 / lex 与 yacc (第二版)

书 号 / ISBN 7-111-10721-7

责任编辑 / 刘立卿

封面设计 / Edie Freedman, 张健

出版发行 / 机械工业出版社

地 址 / 北京市西城区百万庄大街 22 号 (邮政编码 100037)

经 销 / 新华书店北京发行所发行

印 刷 / 北京牛山世兴印刷厂

开 本 / 787 毫米 × 1092 毫米 16 开本 24.5 印张 419 千字

版 次 / 2003 年 1 月第 1 版 2003 年 1 月第 1 次印刷

印 数 / 0001-4000 册

定 价 / 45.00 元

(凡购本书, 如有倒页、脱页、缺页, 由本社发行部调换)

lex 与 yacc

O'Reilly & Associates 公司介绍

为了满足读者对网络 and 软件技术知识的迫切需求,世界著名计算机图书出版机构 O'Reilly & Associates 公司授权机械工业出版社,翻译出版一批该公司久负盛名的英文经典技术专著。

O'Reilly & Associates 公司是世界上在 UNIX、X、Internet 和其他开放系统图书领域具有领导地位的出版公司,同时是联机出版的先锋。

从最畅销的《The Whole Internet User's Guide & Catalog》(被纽约公共图书馆评为二十世纪最重要的 50 本书之一)到 GNN (最早的 Internet 门户和商业网站),再到 WebSite (第一个桌面 PC 的 Web 服务器软件),O'Reilly & Associates 一直处于 Internet 发展的最前沿。

许多书店的反馈表明,O'Reilly & Associates 是最稳定的计算机图书出版商——每一本书都一版再版。与大多数计算机图书出版商相比,O'Reilly & Associates 公司具有深厚的计算机专业背景,这使得 O'Reilly & Associates 形成了一个非常不同于其他出版商的出版方针。O'Reilly & Associates 所有的编辑人员以前都是程序员,或者是顶尖级的技术专家。O'Reilly & Associates 还有许多固定的作者群体——他们本身是相关领域的技术专家、咨询专家,而现在编写著作,O'Reilly & Associates 依靠他们及时地推出图书。因为 O'Reilly & Associates 紧密地与计算机业界联系着,所以 O'Reilly & Associates 知道市场上真正需要什么图书。

目录

前言	1
第一章 lex 和 yacc	9
最简单的 lex 程序	10
用 lex 识别单词	11
语法	23
运行 lex 和 yacc	31
lex 和手写的词法分析程序	32
练习	35
第二章 使用 lex	36
正则表达式	37
单词计数程序	42
分析命令行	48
C 源代码分析程序	55
小结	58
练习	59

第三章 使用 yacc	60
语 法	60
移进 / 归约分析	62
yacc 语法分析程序	65
词法分析程序	68
算术表达式和歧义性	69
变量和有类型的标记	74
符号表	77
函数和保留字	81
用 make 构建语法分析程序	88
小 结	89
练 习	89
第四章 菜单生成语言	91
MGL 的概述	91
开发 MGL	93
构建 MGL	103
屏幕处理	109
结 束	112
MGL 代码示例	115
练 习	120
第五章 分析 SQL	122
SQL 的要点概述	123
语法检查程序	127
语法分析程序	133
嵌入式 SQL	158
练 习	163

第六章 lex 规范参考	164
lex 规范的结构	164
BEGIN	166
程序错误	166
字符变换	168
上下文相关	169
定义（替换）	171
ECHO	172
包含操作（文件的逻辑嵌套）	172
从字符串中输入	174
input()	176
内部表（%N 声明）	177
lex 库	178
行号和 yylineno	179
文字块	179
一个程序中的多个词法分析程序	179
output()	184
lex 词法分析程序的可移植性	185
正则表达式语法	187
REJECT	190
从 yylex() 中返回值	190
起始状态	191
unput()	193
yyinput()、yyoutput()、yyunput()	194
yylen	194
yyless()	194
yylex()	195
yyomore()	196
yytext	197
yywrap()	199

第七章 yacc 语法参考	200
yacc 语法的结构.....	200
动作	201
歧义和冲突	204
yacc 中的程序错误.....	205
结束标记	208
错误标记和错误恢复	208
%ident 声明	209
继承的属性 (\$0).....	209
词汇的反馈.....	211
文字块	212
文字标记	213
yacc 语法分析程序的可移植性.....	213
优先级、结合性和操作符声明	215
递归规则	217
规则	219
特殊字符	220
开始声明	221
符号值	222
标记	223
%type 声明	226
%union 声明	226
变体和多重语法	227
y.output 文件.....	232
yacc 库.....	233
YYABORT	234
YYACCEPT	235
YYBACKUP	235
yyclearin	235
yydebug 和 YYDEBUG	236

yyerror	237
YYERROR	237
yyerror()	237
yyvsparse()	238
YYRECOVERING()	239
第八章 yacc 歧义和冲突	240
指针模型和冲突	240
冲突的普通示例	253
如何修复冲突	258
小结	266
练习	266
第九章 错误报告和恢复	267
错误报告	267
错误恢复	272
练习	276
附录一 AT&T lex	279
附录二 AT&T yacc	287
附录三 Berkeley yacc	297
附录四 GNU bison	303
附录五 flex	306
附录六 MKS lex 和 yacc	318

附录七 Abraxas lex 和 yacc	320
附录八 POSIX lex 和 yacc	322
附录九 MGL 编译程序代码	325
附录十 SQL 分析程序代码	341
参考文献	369
词汇表	371

前言

lex 和 yacc 是特意为编写编译程序和解释程序的人设计的工具，它对非编译程序编写人员所感兴趣的许多应用程序也非常有用。在输入中查找模式或者拥有输入或命令语言的任何应用程序都适于采用 lex 和 yacc。而且，它们允许快速应用程序原型设计，容易修改，而且程序的维护简单。为了激发读者的想像力，下面给出了几个用 lex 和 yacc 开发的应用：

- 台式计算器 *bc*。
- 工具 *eqn* 和 *pic*，用于数学公式和复杂图片的排版预处理器。
- *PCC* 和 *GCC*，*PCC* 是和许多 UNIX 系统一起使用的可移植 C 编译程序，*GCC* 是 GNU C 编译程序。
- 菜单编译程序。
- SQL 数据库语言语法检查程序。
- *lex* 程序本身。

第二版的新特点

在第二版中，我们已经做了大量的修正。主要变化包括：

- 完全重写的介绍性的第一章到第三章。
- 全新的第五章介绍了完整的 SQL 语法。
- 全新的详细的参考性的第六章和第七章。
- 包含了 lex 和 yacc 的所有主要的 MS-DOS 和 UNIX 版本的完整介绍, 包括 AT&T lex 和 yacc、Berkeley yacc、flex、GNU bison、MKS lex 和 yacc、Abraxas PCYACC。
- 包含了 lex 和 yacc 的新的 POSIX 1003.2 标准版本的介绍。

本书范围

第一章“lex 和 yacc”介绍了如何以及为什么使用 lex 和 yacc 来创建编译程序和解释程序, 并且演示了一些小的 lex 和 yacc 应用程序。还介绍贯穿全书的基本术语。

第二章“使用 lex”描述了如何使用 lex。开发了几个 lex 应用程序, 它们分别用于计算文件中单词的个数, 分析程序命令开关和参数和计算 C 程序上的统计数据。

第三章“使用 yacc”给出了使用 lex 和 yacc 开发功能完整的台式计算器的完整示例。

第四章“菜单生成语言”演示了如何使用 lex 和 yacc 开发菜单生成程序。

第五章“分析 SQL”开发了针对完整的 SQL 关系数据库语言的分析程序。首先, 我们将分析程序作为一个语法检查程序, 然后将它扩展为内置于 C 程序中的 SQL 的简单预处理程序。

第六章“lex 规范参考”和第七章“yacc 语法参考”详细描述了 lex 和 yacc 程序设计人员可用的特性和选项。当开发新的 lex 和 yacc 应用程序时, 这两章以及随后的两章为有经验的 lex 和 yacc 程序设计人员提供了丰富的技术信息。

第八章“yacc 歧义和冲突”解释了 yacc 歧义和冲突，它们是造成 yacc 不能正确分析语法的原因。然后介绍了定位和纠正这些问题的方法。

第九章“错误报告和恢复”讨论了编译程序和解释程序设计人员定位、识别和报告编译程序输入中的错误的技术。

附录一“AT&t lex”描述了 AT&T lex 的命令行语法、报告的错误消息以及建议的解决方案。

附录二“AT&T yacc”描述了 AT&T yacc 的命令行语法，并且列出了由 yacc 报告的错误，提供了能导致这种错误的代码示例和建议的解决方案。

附录三“Berkeley yacc”描述了 Berkeley yacc 的命令行语法，以及随 Berkeley UNIX 一起发布的、已广泛使用的免费版本的 yacc，并且列出了由 Berkeley yacc 报告的错误及其建议的解决方案。

附录四“GNU bison”讨论了 bison 中的不同之处。bison 是自由软件基金会实现的 yacc。

附录五“flex”讨论了 flex (广泛使用的免费版本的 lex)，列出了它与其他版本的差别，以及由 flex 报告的错误及其解决方案。

附录六“MKS lex 和 yacc”讨论了来自 Mortice Kern Systems 的 lex 和 yacc 的 MS-DOS 以及 OS/2 版本。

附录七“Abraxas lex 和 yacc”讨论了来自 Abraxas Software 的 lex 和 yacc 的 MS-DOS 以及 OS/2 版本 PCYACC。

附录八“POSIX lex 和 yacc”讨论了由 IEEE POSIX 1003.2 标准定义的 lex 和 yacc 版本。

附录九“MGL 编译程序代码”为第四章讨论的菜单生成语言编译程序提供了完整的源代码。

附录十“SQL分析程序代码”提供了第五章讨论的SQL分析程序的完整的源代码和交叉引用。

词汇表列出了关于语言和编译理论的技术术语。

参考文献列出了有关lex和yacc的其他文档,以及对编译程序设计有帮助的书籍。

我们假定读者都熟悉C语言,因为大多数示例都是用C、lex或yacc编写的,书中有些示例是用特殊目的语言开发的。

lex和yacc的可用性

lex和yacc都是在20世纪70年代由贝尔实验室开发的。yacc的开发早于lex,它是由Stephen C. Johnson开发的。为了与yacc一起工作, Mike Lesk和Eric Schmidt开发了lex。自第7版UNIX以来, lex和yacc已经成为标准的UNIX实用程序。System V和老式的BSD版本使用原始的AT&T版本,而最新版本的BSD使用flex(见下文)和Berkeley yacc。开发者撰写的文章是lex和yacc的主要信息源。

自由软件基金会的GNU工程组发布了**bison**,即yacc的替代品;bison是由Robert Corbett和Richard Stallman编写的。Charles Donnelly和Richard Stallman编写的bison手册特别好,尤其是特殊特性参考部分。附录四论述了bison。

BSD和GNU工程组还发布了**flex**(快速词法分析发生器, *Fast Lexical Analyzer Generator*),其参考页中称:“lex的重写纠正了那个工具的部分不足”。flex最初由Jef Poskanzer编写;Vern Paxson和Van Jacobson又对它进行了重大改善,而且Vern目前仍在维护它。附录五概述了有关flex的特定主题。

至少有两种版本的lex和yacc可用于MS-DOS和OS/2计算机。MKS(Mortice Kern Systems Inc.)是MKS工具包的出版商,他将lex和yacc作为一种独立的产品提出来以满足许多PC的C语言编译。MKS lex和yacc有一本非常好的手册。附录六概述了MKS lex和yacc。Abraxas Software公司发布了PCYACC,它是

另一种 `lex` 和 `yacc` 版本，该版本带有许多广泛使用的编程语言的示例分析程序。附录七概述了 `Abraxas` 的 `lex` 和 `yacc` 版本。

示例程序

本书的程序可以从 UUNET 上免费得到（指的是除了 UUNET 普通连接时间费用之外免费）。如果访问 UUNET，就可以使用 UUCP 或 FTP 检索到源代码。对于 UUCP，只需找到一种能直接访问 UUNET 的机器，并且键入下面的命令：

```
uucp uUNET\!~/nutshell/lexyacc/progs.tar.Z yourhost\!~/yourname/
```

如果用 Bourne shell (`sh`) 代替 C shell (`csh`)，可以省略反斜线。等一段时间（一天或更长时间）后，文件就会出现在 `/usr/spool/uncppublic/yourname` 目录下。如果你还没有账户，但是却想要一个账户接收电子邮件，则可以打电话 703-204-8000 与 UUNET 联系。

要使用 `ftp`，就要找到一种能直接访问 Internet 的机器。下面是一个示例会话，黑体字表示命令。

```
% ftp ftp.oreilly.com
Connected to ftp.oreilly.com.
220 FTP server (Version 5.99 Wed May 23 14:40:19 EDT 1990) ready.
Name (ftp.oreilly.com:yourname): anonymous
331 Guest login ok, send ident as password.
Password: ambar@ora.com (use your user name and host here)
230 Guest login ok, access restrictions apply.
ftp> cd published/oreilly/nutshell/lexyacc
250 CWD command successful.
ftp> binary (you must specify binary transfer for compressed files)
200 Type set to I.
ftp> get progs.tar.Z
200 PORT command successful.
150 Opening BINARY mode data connection for progs.tar.Z.
226 Transfer complete.
ftp> quit
221 Goodbye.
%
```

该文件是一个压缩的 tar 档案文件。一旦得到这个档案文件就要进行解压缩，键入：

```
% zcat progs.tar.Z | tar xf -
```

System V 系统需要换用下面的 tar 命令：

```
% zcat progs.tar.Z | tar xof -
```

排版约定

本书使用下列排版约定：

黑体 (**Bold**)

用于语句、函数、标识符和程序名。

斜体 (*Italic*)

在段落中表示文件、目录和命令名以及数据类型，当介绍新的术语和概念时也用斜体来强调。

等宽 (Constant Width)

用在示例中表示文件的内容或命令的输出。

等宽黑体 (**Constant Bold**)

用在示例中表示逐字键入的命令和选项。

引号

用于标识解释性文本中的代码段。系统消息、记号和符号也用引号引起。

%

shell 提示符。

[]

在程序语法描述中用来表示可选的元素（不要键入括号本身）。