

可下载教学资料

<http://www.tup.tsinghua.edu.cn>



高等学校教材
计算机科学与技术

编译原理及实践教程

黄贤英 王柯柯 编著



清华大学出版社

TP314/72

2008

高等学校教材
计算机科学与技术

编译原理及实践教程

黄贤英 王柯柯 编著

清华大学出版社
北京

内 容 简 介

本书以提高学生的软件开发能力为目标,除了介绍与编译有关的基础知识外,更注重介绍设计、开发编译程序的过程、方法和技术。为达到此目标,本书以一个编译器的设计实现案例贯穿全书,以 Sample 语言为代表,利用软件工程的思想,采取自顶向下分析设计、逐步求精的方法,从词法分析开始,随着教学的深入逐步对编译程序及相关部分进行分析、设计,直至完成整个编译器的设计。

本书强调理论和实践相结合,主要从实用性和可操作性的角度把握;内容丰富,文字精练,重点突出,不求大而全,但求简而精。本书面向普通高校计算机专业的本科学学生,也可作为自学考试教材及读者自学的参考书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13501256678 13801310933

图书在版编目(CIP)数据

编译原理及实践教程/黄贤英,王柯柯编著. —北京:清华大学出版社,2008.2

(高等学校教材·计算机科学与技术)

ISBN 978-7-302-16422-7

I. 编… II. ①黄… ②王… III. 编译程序—程序设计—高等学校—教材 IV. TP314

中国版本图书馆 CIP 数据核字(2007)第 170722 号

责任编辑:付弘宇 徐跃进

责任校对:梁毅

责任印制:杨艳

出版发行:清华大学出版社 地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn> 邮 编:100084

c-service@tup.tsinghua.edu.cn

社总机:010-62770175 邮购热线:010-62786544

投稿咨询:010-62772015 客户服务:010-62776969

印刷者:北京国马印刷厂

装订者:北京市密云县京文制本装订厂

经 销:全国新华书店

开 本:185×260 印 张:14.25 字 数:346 千字

版 次:2008 年 2 月第 1 版 印 次:2008 年 2 月第 1 次印刷

印 数:1~3000

定 价:25.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系
调换。联系电话:(010)62770177 转 3103 产品编号:026680-01

编审委员会成员

(按地区排序)

清华大学	周立柱	教授
	覃征	教授
	王建民	教授
	刘强	副教授
	冯建华	副教授
北京大学	杨冬青	教授
	陈钟	教授
	陈立军	副教授
北京航空航天大学	马殿富	教授
	吴超英	副教授
	姚淑珍	教授
中国人民大学	王珊	教授
	孟小峰	教授
	陈红	教授
北京师范大学	周明全	教授
北京交通大学	阮秋琦	教授
北京信息工程学院	孟庆昌	教授
北京科技大学	杨炳儒	教授
石油大学	陈明	教授
天津大学	艾德才	教授
复旦大学	吴立德	教授
	吴百锋	教授
	杨卫东	副教授
华东理工大学	邵志清	教授
华东师范大学	杨宗源	教授
	应吉康	教授
东华大学	乐嘉锦	教授
上海第二工业大学	蒋川群	教授
浙江大学	吴朝晖	教授
	李善平	教授
南京大学	骆斌	教授
南京航空航天大学	秦小麟	教授
南京理工大学	张功萱	教授

南京邮电学院	朱秀昌	教授
苏州大学	龚声蓉	教授
江苏大学	宋余庆	教授
武汉大学	何炎祥	教授
华中科技大学	刘乐善	教授
中南财经政法大学	刘腾红	教授
华中师范大学	叶俊民	教授
	玉林平	副教授
	魏开平	副教授
国防科技大学	赵克佳	教授
	肖 侬	副教授
中南大学	陈松乔	教授
	刘卫国	教授
湖南大学	林亚平	教授
	邹北骥	教授
西安交通大学	沈钧毅	教授
	齐 勇	教授
长安大学	巨永峰	教授
西安石油学院	方 明	教授
西安邮电学院	陈莉君	副教授
哈尔滨工业大学	郭茂祖	教授
吉林大学	徐一平	教授
	毕 强	教授
长春工程学院	沙胜贤	教授
山东大学	孟祥旭	教授
	郝兴伟	教授
山东科技大学	郑永果	教授
中山大学	潘小轰	教授
厦门大学	冯少荣	教授
福州大学	林世平	副教授
云南大学	刘惟一	教授
重庆邮电学院	王国胤	教授
西南交通大学	杨 燕	副教授

改革开放以来,特别是党的十五大以来,我国教育事业取得了举世瞩目的辉煌成就,高等教育实现了历史性的跨越,已由精英教育阶段进入国际公认的大众化教育阶段。在质量不断提高的基础上,高等教育规模取得如此快速的发展,创造了世界教育发展史上的奇迹。当前,教育工作既面临着千载难逢的良好机遇,同时也面临着前所未有的严峻挑战。社会不断增长的高等教育需求同教育供给特别是优质教育供给不足的矛盾,是现阶段教育发展面临的基本矛盾。

教育部一直十分重视高等教育质量工作。2001年8月,教育部下发了《关于加强高等学校本科教学工作,提高教学质量的若干意见》,提出了十二条加强本科教学工作提高教学质量的措施和意见。2003年6月和2004年2月,教育部分别下发了《关于启动高等学校教学质量与教学改革工程精品课程建设工作的通知》和《教育部实施精品课程建设提高高校教学质量和人才培养质量》文件,指出“高等学校教学质量和教学改革工程”是教育部正在制定的《2003—2007年教育振兴行动计划》的重要组成部分,精品课程建设是“质量工程”的重要内容之一。教育部计划用五年时间(2003—2007年)建设1500门国家级精品课程,利用现代化的教育信息技术手段将精品课程的相关内容上网并免费开放,以实现优质教学资源共享,提高高等学校教学质量和人才培养质量。

为了深入贯彻落实教育部《关于加强高等学校本科教学工作,提高教学质量的若干意见》精神,紧密配合教育部已经启动的“高等学校教学质量与教学改革工程精品课程建设工作”,在有关专家、教授的倡议和有关部门的大力支持下,我们组织并成立了“清华大学出版社教材编审委员会”(以下简称“编委会”),旨在配合教育部制定精品课程教材的出版规划,讨论并实施精品课程教材的编写与出版工作。“编委会”成员皆来自全国各类高等学校教学与科研第一线的骨干教师,其中许多教师为各校相关院、系主管教学的院长或系主任。

按照教育部的要求,“编委会”一致认为,精品课程的建设工作从开始就要坚持高标准、严要求,处于一个比较高的起点上;精品课程教材应该能够反映各高校教学改革与课程建设的需要,要有特色风格、有创新性(新体系、新内容、新手段、新思路,教材的内容体系有较高的科学创新、技术创新和理念创新的含量)、先进性(对原有的学科体系有实质性的改革和发展,顺应并符合新世纪教学发展的规律、代表并引领课程发展的趋势和方向)、示范性(教材所体现的课程体系具有较广泛的辐射性和示范性)和一定的前瞻

性。教材由个人申报或各校推荐(通过所在高校的“编委会”成员推荐),经“编委会”认真评审,最后由清华大学出版社审定出版。

目前,针对计算机类和电子信息类相关专业成立了两个“编委会”,即“清华大学出版社计算机教材编审委员会”和“清华大学出版社电子信息教材编审委员会”。首批推出的特色精品教材包括:

(1) 高等学校教材·计算机应用——高等学校各类专业,特别是非计算机专业的计算机应用类教材。

(2) 高等学校教材·计算机科学与技术——高等学校计算机相关专业的教材。

(3) 高等学校教材·电子信息——高等学校电子信息相关专业的教材。

(4) 高等学校教材·软件工程——高等学校软件工程相关专业的教材。

(5) 高等学校教材·信息管理与信息系统。

(6) 高等学校教材·财经管理与计算机应用。

清华大学出版社经过 20 多年的努力,在教材尤其是计算机和电子信息类专业教材出版方面树立了权威品牌,为我国的高等教育事业做出了重要贡献。清华版教材形成了技术准确、内容严谨的独特风格,这种风格将延续并反映在特色精品教材的建设中。

清华大学出版社教材编审委员会

E-mail: dingl@tup.tsinghua.edu.cn

“编译原理”是计算机及其相关专业的重要专业基础课,主要研究设计和构造编译程序的原理和方法。编译原理蕴涵着计算机学科中解决问题的思路、形式化问题和解决问题的方法,对应用软件和系统软件的设计与开发有一定的启发和指导作用,编译程序构造的原理和技术在软件工程、语言转换等许多领域中有着广泛应用。

本书主要面向一般本科院校,理论学时为 40~48 学时,压缩了编译课程中的理论部分,删除了实用意义不大的编译方法。以程序编译的五个主要阶段——词法分析、语法分析、中间代码生成、代码优化和目标代码生成成为线索,重点放在设计与构造编译程序及各个组成部分的软件技术和实用方法上。通过本课程的教学,可以使建立一个较为完整的编译系统的模型,掌握各个阶段的基本算法、常用的编译技术和方法,为今后从事应用软件和系统软件的开发打下一定的理论和实践基础。

本书的特色主要有:

(1) 力求将基本概念、基本原理和实现方法的思路阐述清楚,条理清晰,通俗易懂;

(2) 为便于自学,书中引入典型例题,以实例形式讲解理论,加强学生对理论的理解,并配以大量习题,以巩固所学的知识,并提供了参考答案;

(3) 为切实做到理论联系实际,便于读者更深刻地理解编译程序的实现过程,以 Sample 语言为例,贯穿本书的各个章节,重点介绍 Sample 语言的编译程序在各个阶段的具体实现技术和构造方法,并给出了部分程序框架;

(4) 本书注重实际应用,配套软件实现了 Sample 语言的词法分析、语法分析、语法制导的翻译以及本书涉及到的各种核心算法,形象生动地展示了编译程序的分析过程。教师可将该软件用作课堂教学演示,也可用作“编译原理”课程作业的参考实例,作为学生编译课程的实训内容;学生可以通过该软件自学,在课后反复观看揣摩,并尝试实现其中的部分内容。加 * 的章节为可选内容,请教师根据具体情况选择。

本书的第 1、3~5、8 章由黄贤英编写,第 2、6、7 章由王柯柯编写。刘恒洋、曹琼老师为本书的配套软件做了大量工作,杨武、刘洁老师为本书的编写提供了宝贵建议,在此一并表示衷心感谢。

本书配套的电子教案已经过作者多年教学试用,重点突出,脉络清晰,教学效果良好。此外,本书提供的配套软件可在 Windows 2000/XP 等环境下运行,下载后直接安装即可使用。这些资源均可从清华大学出版社网站(<http://www.tup.tsinghua.edu.cn>)的相关网页上下载。

限于作者的水平和精力,书中难免存在不足甚至错误,欢迎广大读者指正。如果在本书及其配套资源的使用中有任何问题或建议,请发邮件至 fuhy@tup.tsinghua.edu.cn。

作者

2007年8月

第 1 章 概述	1
1.1 程序设计语言与翻译程序	1
1.2 编译过程和编译程序的结构	3
1.2.1 编译过程概述	3
1.2.2 编译程序的结构	7
1.2.3 编译阶段的组合	8
1.3 编译程序的构造	9
1.3.1 编译程序的构造与生成	9
1.3.2 Sample 语言编译程序的设计	10
1.4 编译程序的发展及应用	11
1.4.1 编译程序的发展	11
1.4.2 为什么要学习编译原理及其构造技术	11
1.4.3 编译技术的应用	12
1.5 小结	13
1.6 习题	13
第 2 章 形式语言基本知识	14
2.1 符号和符号串	14
2.2 上下文无关文法及其语言	15
2.2.1 上下文无关文法	15
2.2.2 推导、句型和语言	17
2.3 语法树和文法的二义性	19
2.3.1 语法分析树	19
2.3.2 文法的二义性	20
2.4 文法的分类	21
2.5 类 Pascal 语言 Sample 的简单介绍	22
2.5.1 Sample 语言字符集的定义	22
2.5.2 Sample 语言单词的定义	23

2.5.3	Sample 语言数据类型的定义	23
2.5.4	Sample 语言表达式的定义	24
2.5.5	Sample 语言语句的定义	24
2.5.6	Sample 语言程序的定义	25
2.5.7	符合 Sample 语言定义的源程序举例	25
2.6	小结	26
2.7	习题	26
第 3 章	词法分析	28
3.1	词法分析的任务和功能	28
3.1.1	词法分析的功能	28
3.1.2	单词的类型和类别码	29
3.2	词法分析器的设计	31
3.2.1	词法分析器的接口	31
3.2.2	词法分析程序的总体设计	31
3.2.3	词法分析程序的详细设计	33
3.2.4	单词的识别和状态转换图	33
3.2.5	符号表及其操作	36
3.2.6	词法分析阶段的错误处理	37
3.3	正规文法、正规式与有穷自动机	38
3.3.1	正规文法	38
3.3.2	正规式	39
3.3.3	有穷自动机	40
3.3.4	正规文法与有穷自动机的等价性	47
3.3.5	正规式与有穷自动机的等价性	48
3.4	词法分析器的自动产生	50
3.4.1	Lex 的概述	51
3.4.2	Lex 源文件的格式	51
3.4.3	Lex 源文件中正规式的约定	52
3.4.4	Lex 源文件的翻译规则	54
3.4.5	Lex 源程序设计举例	54
3.4.6	Lex 使用中的一些注意事项	58
3.4.7	使用 Lex 自动生成 Sample 语言的词法分析器	59
3.5	小结	60
3.6	习题	60
第 4 章	语法分析	64
4.1	语法分析概述	64
4.2	自上而下分析面临的问题及其解决方法	65

4.2.1	确定的自上而下分析面临的两个问题	65
4.2.2	左递归的消除	67
4.2.3	回溯的消除	68
4.2.4	LL(1)文法	69
4.3	确定的自上而下的分析方法	73
4.3.1	递归下降分析法	73
4.3.2	预测分析法	77
4.4	Sample 语言自上而下语法分析程序的设计	82
4.5	自下而上语法分析的思想	84
4.5.1	“移进-归约”分析方法	84
4.5.2	规范归约与句柄	86
4.6	算符优先分析	88
4.6.1	算符优先文法	88
4.6.2	算符优先表的构造	89
4.6.3	算符优先分析方法	93
*4.6.4	算符优先函数	95
4.7	LR 分析法	97
4.7.1	LR 分析概述	97
4.7.2	LR(0)分析	100
4.7.3	SLR(1)分析	108
*4.7.4	LR(1)分析	111
*4.7.5	LALR(1)分析	115
*4.7.6	二义文法在 LR 分析中的应用	117
4.8	语法分析器的自动生成工具 Yacc	119
4.8.1	Yacc 概述	119
4.8.2	Yacc 源文件的格式	120
4.8.3	Yacc 的翻译规则	122
4.8.4	Yacc 的辅助程序	123
4.9	语法分析程序中的错误处理	124
4.9.1	语法分析中的错误处理的一般原则	124
4.9.2	自上而下语法分析的错误处理	125
4.9.3	自下而上语法分析的错误处理	127
4.10	小结	131
4.11	习题	132
第 5 章	语义分析和中间代码生成	136
5.1	静态语义检查	137
5.2	中间代码的形式	137
5.2.1	逆波兰式	137

5.2.2	三地址代码	138
5.3	属性文法和语法制导的翻译	140
5.3.1	属性文法	140
5.3.2	综合属性	141
5.3.3	继承属性	142
5.3.4	语法制导的翻译方法	143
5.4	常见语句的语法制导的翻译	144
5.4.1	常量说明语句的翻译	145
5.4.2	变量说明语句的翻译	146
5.4.3	简单算术表达式和赋值语句的翻译	147
5.4.4	布尔表达式的翻译	149
5.4.5	IF 语句的翻译	155
5.4.6	REPEAT 语句的翻译	157
5.4.7	FOR 语句的翻译	158
5.5	Sample 语言语法制导的翻译程序的设计	160
5.6	小结	161
5.7	习题	162
第 6 章	运行时存储空间的组织	164
6.1	源程序及其目标程序执行时的活动	164
6.1.1	源程序中的过程	164
6.1.2	过程执行时的活动	165
6.1.3	名字的作用域	166
6.1.4	参数的传递	166
6.1.5	名字的绑定	167
6.2	程序执行时的存储器组织	168
6.2.1	程序执行时存储器的划分	168
6.2.2	局部数据区的内容	169
6.2.3	存储分配策略	170
6.3	静态存储分配	171
6.3.1	静态存储分配的性质	171
6.3.2	静态存储分配的实现	172
6.3.3	临时变量的地址分配	173
6.4	栈式存储分配	173
6.5	堆式存储分配	175
6.6	小结	176
6.7	习题	176

第 7 章 代码优化	178
7.1 概述	178
7.1.1 代码优化器的结构及所处的地位	178
7.1.2 优化的原则	179
7.1.3 优化的主要种类	180
7.1.4 基本块的概念及流图	180
7.2 局部优化	182
7.2.1 删除公共子表达式	182
7.2.2 复写传播	182
7.2.3 删除无用代码	183
7.2.4 对程序进行代数恒等变换	184
7.2.5 利用基本块的 DAG 进行优化	185
7.3 循环优化	188
7.3.1 循环的定义	188
7.3.2 代码外提	189
7.3.3 强度削弱	190
7.3.4 删除归纳变量	191
7.4 小结	192
7.5 习题	192
第 8 章 目标代码生成	194
8.1 概述	194
8.2 目标机器	196
8.3 简单的代码生成算法	197
8.3.1 中间代码的简单翻译方法	198
8.3.2 引用信息和活跃信息	199
8.3.3 寄存器描述和地址描述	201
8.3.4 基本块的代码生成算法	202
8.4 从 DAG 生成目标代码	204
8.5 Sample 语言代码优化及目标代码生成器的设计	207
8.6 小结	209
8.7 习题	209
参考文献	210

概 述

任何一门高级语言都需要有相应的翻译程序将其翻译为机器语言,才能真正在计算机上执行。编译程序是一种翻译程序,编写编译程序所涉及的一些原理、技术和方法在计算机相关的各个领域中都有广泛的应用,学习它具有非常重要的意义。本章首先介绍编译原理的基本概念及与编译程序相关的一些工具,然后介绍编译的过程以及编译程序的结构、组成以及构造方法,最后简单介绍了编译技术的发展及其应用。

1.1 程序设计语言与翻译程序

在计算机系统中,语言分三个层次:机器语言(machine language)、汇编语言(assembly language)和高级语言(high-level language),如图 1.1 所示。只有机器语言程序可直接在计算机上执行,因此用高级语言和汇编语言编写的程序必须被翻译为对应的机器语言程序才能执行。人们把能将一种语言(称为源语言,source language)书写的程序翻译成另一种语言(称为目标语言,target language)书写的程序的程序统称为翻译程序(translator),翻译前后的程序在逻辑上是等价的。

计算机刚出现时,人们用机器语言编写程序,如要编写“将 2 放到一个存储单元中”,机器代码为“C7 06 0000 0002”。用机器语言编写程序费时、乏味,开发难度大,周期长,很快就被汇编语言代替了。

汇编语言以助记符的形式表示指令和地址。如与上述指令等价的汇编代码为“MOV X,2”,它不能直接执行,需要汇编程序(assembly)将其翻译为机器语言程序才能执行。与汇编程序有关的翻译程序有:反汇编程序(disassembler)是把机器语言程序逆向翻译为汇编语言程序的程序;交叉汇编程序(cross assembler)是把甲计算机上的汇编语言程序翻译为乙计算机上的机器语言程序的程序。

然而,汇编语言仍与人类的思维相差甚远,不易阅读和理解,它严格依赖于机器,在一种机器上编写的代码应用于另一种机器时必须完全重写。高级语言的出现缩短了人类思维和计算机语言间的差距,上述指令用 Pascal 语言写为 $x:=2$ 。编写高级语言程序类似于定义数学公式或书写自然语言,与机器无关。起初人们担心这不可能,或者即使可能,目标代码也会因效率不高而没有多大用处。编译程序(compiler)的出现解除了人们的这种担忧,它能够直接将高级语言(如 C 或 Java、Pascal 等)程序翻译为对应的低级语言(如汇编语言或

机器语言)程序,又称为编译器。

实际上,除了上述程序之间的翻译之外,同一种机器上的不同语言和不同种机器上的相同或不同语言书写的程序之间都可以进行翻译,图 1.1 给出了常见语言程序之间的翻译模式。高级语言之间可以相互转换,把一种高级语言程序转换为另一种高级语言程序的程序,统称为转换程序(converter)。图 1.1 中的交叉编译程序(cross compiler)是指能够将甲计算机上的高级语言程序翻译为乙计算机上的机器语言或汇编语言程序的程序。

用高级语言编写程序简单方便,多数程序都可用高级语言编写。在一台计算机上要运行某一种高级语言程序,至少要为该语言配有一个编译程序。对有些高级语言甚至配置了几个不同性能的编译程序,以实现用户的不同需求。

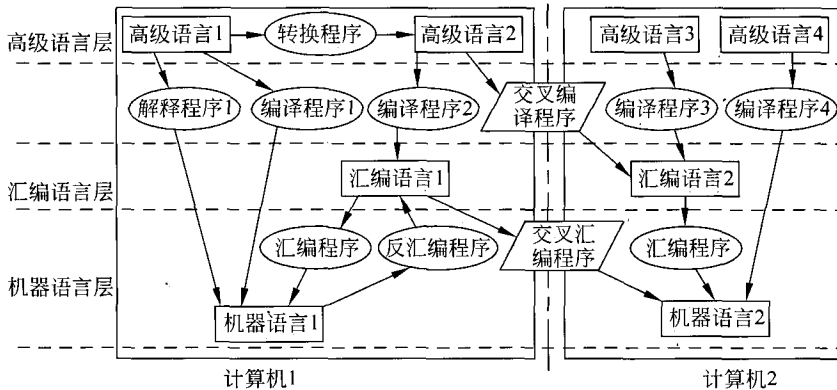


图 1.1 语言层次和转换关系

根据用途和侧重点的不同,编译程序可进一步分为几类。专门用于帮助程序开发和调试的编译程序称为诊断编译程序(diagnostic compiler);着重于提高目标代码效率的编译程序称为优化编译程序(optimizing compiler);运行编译程序的计算机称为宿主机,运行编译程序所产生的目标代码的计算机称为目标机;如果不需要重写编译程序中与机器无关的部分就能改变目标机,则称该编译程序为可变目标编译程序(retargetable compiler)。

编译程序的重要性在于它使得多数计算机用户不必考虑与机器有关的烦琐细节,使程序员独立于机器硬件。除编译程序外,还需要其他一些程序相互配合才能使高级语言程序生成能在计算机上执行的目标程序。图 1.2 给出了一个典型的高级语言程序的处理过程。

图 1.2 中的预处理程序(preprocessor)是指:当源程序以几个模块的形式存放在不同的文件中,将这些源程序汇集在一起的程序。预处理程序也能够把源程序中称为宏的缩写语句展开为原始语句加入到源程序中。源程序由编译程序编译后生成目标程序,图中的目标代码是汇编代码,需要经过汇编程序汇编转换为可装配的机器代码。再经装配连接程序连接成真正能在机器上运行的代码。装配连接程序(loader-linker)是指将可再装配的机器代码进行装配

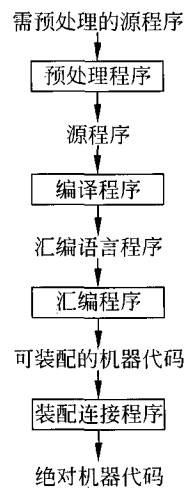


图 1.2 高级语言程序处理过程

连接形成绝对机器代码的程序。

在实际使用中,高级语言除了通过编译程序将其翻译为机器语言执行外,解释程序也可以把高级语言翻译为机器语言。**解释程序(interpreter)**是指按高级语言程序的语句执行的顺序边翻译边执行相应功能的程序。编译程序和解释程序的主要区别是:

- (1) 编译程序先把全部源程序翻译为目标程序,然后再执行,该目标程序可以反复执行;
- (2) 解释程序对源程序逐句地翻译执行,目标代码只能执行一次,若需重新执行,则必须重新解释源程序。

编译过程类似笔译,笔译后的结果可以反复阅读,而解释过程则类似于同步翻译,别人说一句,就译一句,翻译的结果没有保存。图 1.3 是两个过程的比较。

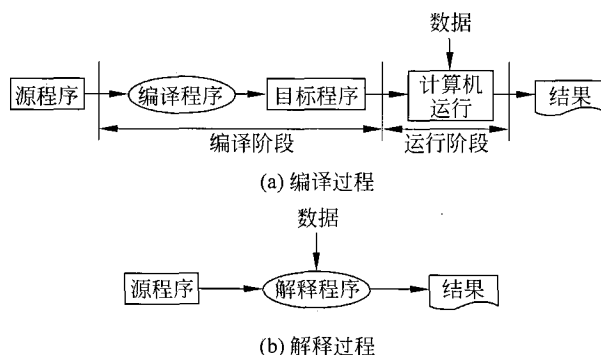


图 1.3 编译与解释过程对比

本书重点介绍编译程序及相关原理、方法,关于解释程序不作深入探讨,许多编译程序的构造与实现技术同样适用于解释程序。

1.2 编译过程和编译程序的结构

编译程序完成从源程序到目标程序的翻译工作,这是一个复杂的过程。整个工作过程需要分阶段完成,每个阶段完成不同的任务,各个阶段进行的操作在逻辑上是紧密相关的,每个阶段的工作都通过相应的程序来完成。编译程序就是由完成这些功能的全部程序组成的。

1.2.1 编译过程概述

为了研究和学习,根据各个阶段的复杂程度、理论基础和实现方法的不同,通常将编译程序的工作过程划分为词法分析、语法分析、语义分析与中间代码生成、代码优化、目标代码生成五个阶段(如图 1.4 所示),这是一种普遍的划分方法。

下面用例 1.1 的 Pascal 程序来扼要介绍编译程序如何将其翻译为目标程序,以及源程序在各个不同阶段被转换后的表示形式、各个阶段的任务。

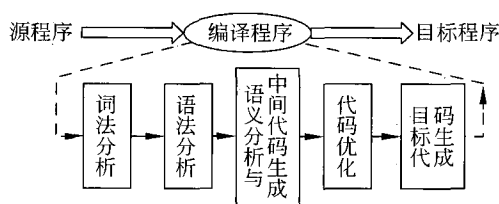


图 1.4 编译的各个阶段