



“基于系统能力培养的计算机专业课程建设研究”项目规划教材
普通高等教育“十一五”国家级规划教材

编译 技术

Compiler Technology

张莉 史晓华 杨海燕 金茂忠 编著

高等教育出版社



“基于系统能力培养的计算机专业课程建设研究”项目规划教材
普通高等教育“十一五”国家级规划教材

编译 技术

Compiler Technology

张莉 史晓华 杨海燕 金茂忠 编著

高等教育出版社·北京

内容提要

本书为“基于系统能力培养的计算机专业课程建设研究”项目规划教材,同时也是国家精品资源共享课程主讲教材。本书对传统编译技术课程内容进行了结构性改革,首先暂时抛开大量形式化方法,先给学生一个完整的编译过程,以及这个过程中涉及的编译技术,在该过程中同时介绍相关的理论和方法。对于编译过程中涉及的形式化方法、编译自动生成技术、编译优化技术等,则将其放在了一个完整的编译过程之后,作为必要的补充。

全书共分三部分。其中,第一部分基础篇(1~10章),包含编译技术概述、语言与文法基础、一个简单编译器的构造(一个完整的编译过程)。第二部分提高篇(11~16章),重点介绍编译程序的自动化生成技术和代码优化及面向目标机的代码生成技术。第三部分实例篇(17~18章),给出两个小型编译系统的完整设计。与教材配套的课程网站(易课程网、爱课程网)包括课程教学视频、电子教案、案例源代码等教学资源。

本书可作为本科计算机类专业编译技术课程教材,也可供相关技术人员参考使用。

图书在版编目(CIP)数据

编译技术 / 张莉等编著. -- 北京:高等教育出版社,2016.9

ISBN 978-7-04-046317-0

I. ①编… II. ①张… III. ①编译程序-程序设计-高等学校-教材 IV. ①TP314

中国版本图书馆 CIP 数据核字(2016)第 196396 号

策划编辑 倪文慧
插图绘制 杜晓丹

责任编辑 倪文慧
责任校对 刘 莉

封面设计 张申申
责任印制 田 甜

版式设计 张 杰

出版发行 高等教育出版社
社 址 北京市西城区德外大街 4 号
邮政编码 100120
印 刷 北京市昌平百善印刷厂
开 本 787mm×1092mm 1/16
印 张 31
字 数 590 千字
购书热线 010-58581118
咨询电话 400-810-0598

网 址 <http://www.hep.edu.cn>
<http://www.hep.com.cn>
网上订购 <http://www.hepmall.com.cn>
<http://www.hepmall.com>
<http://www.hepmall.cn>

版 次 2016 年 9 月第 1 版
印 次 2016 年 9 月第 1 次印刷
定 价 48.00 元

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换
版权所有 侵权必究
物 料 号 46317-00

数字课程资源使用说明

与本书配套的数字课程资源发布在高等教育出版社易课程网站,请登录网站后开始课程学习。

一、注册/登录

访问 <http://abook.hep.com.cn/18671010>,点击“注册”,在注册页面输入用户名、密码及常用的邮箱进行注册。已注册的用户直接输入用户名和密码登录即可进入“我的课程”页面。

二、课程绑定

点击“我的课程”页面右上方“绑定课程”,正确输入教材封底防伪标签上的 20 位密码,点击“确定”完成课程绑定。

三、访问课程

在“正在学习”列表中选择已绑定的课程,点击“进入课程”即可浏览或下载与本书配套的课程资源。刚绑定的课程请在“申请学习”列表中选择相应课程并点击“进入课程”。

四、与本书配套的易课程数字课程资源包括重难点教学视频、电子教案和案例源代码,以便读者学习使用。

账号自登录之日起一年内有效,过期作废。

如有账号问题,请发邮件至: abook@hep.com.cn。

序

从技术的角度看,现代计算机工程呈现出系统整体规模日趋庞大、子系统数量日趋增长且交联关系日趋复杂等特征。这就要求计算机工程技术人才必须从系统的高度多维度地研究与构思,综合运用多种知识进行工程实施,并在此过程中反复迭代以寻求理想的系统平衡性。上述高素质计算机专业人才的培养,是当前我国高校计算机类专业教育的重要目标。

经过半个多世纪的建设,我国计算机专业课程体系完善、课程内容成熟,但在高素质计算机专业人才的培养方面还存在一些普遍性问题。

(1)突出了课程个体的完整性,却缺乏课程之间的融通性。每门课程教材都是一个独立的知识体,强调完整性,相关知识几乎面面俱到,忽略了前序课程已经讲授的知识以及与课程之间知识的相关性。前后课程知识不能有效地整合与衔接,学生难以系统地理解课程知识体系。

(2)突出了原理性知识学习,却缺乏工程性实现方法。课程教学往往突出原理性知识的传授,注重是什么、有什么,却缺乏一套有效的工程性构建方法,学生难以实现具有一定规模的实验。

(3)突出了分析式教学,却缺乏综合式教学。分析式教学方法有利于学习以往经验,却难于培养学生的创新能力,国内高校计算机专业大多是分析式教学。从系统论观点看,分析式方法是对给定系统结构,分析输入输出关系;综合式方法是对给定的输入输出关系,综合出满足关系的系统结构。对于分析式教学方法来说,虽然学生理解了系统原理,但是仍然难于重新构造系统结构。只有通过综合式教学方法,才能使得学生具有重新构造系统结构的能力。

在此背景下,教育部高等学校计算类专业指导委员会提出了系统能力培养的课题研究。这里所说的“系统能力”,是指能理解计算机系统的整体性、关联性、动态性和开放性,掌握硬软件协同工作及相互作用机制,并综合运用多种知识与技术完成系统开发的能力。以系统能力培养为目标的教学改革,是指将本科生自主设计“一台功能计算机、一个核心操作系统、一个编译系统”确立为教学目标,并据此重构计算机类课程群,即注重离散数学的基础,突出“数字逻辑”“计算机组成”“操作系统”“编译原理”4门课程群的融合,形成边界清晰且有序衔接的课群知识体系。在教学实验上,强调按工业标准、工程规模、工程方法以及工具环境设计与开发系统,提高学生设计开发复杂工程问

题解决方案的系统能力。

在课题研究的基础上,计算机类教学指导委员会研制了《高校计算机类专业系统能力培养研究报告》(以下简称《研究报告》)。其总体思路是:通过对系统能力培养的课程体系教学工作凝练总结,明确系统能力培养目标,展现各学校已有的实践和探索经验;更重要的是总结出一般性方法,推动更多高校开展计算机类专业课程改革。国内部分高校通过长期的系统能力培养教学改革探索与实践,不仅提高了学生的系统能力,同时还总结出由顶层教学目标驱动“课程群为中心”的课程体系建设模式,为计算机专业改革提供了有益参考。这些探索与实践成果,也为计算机类专业工程教育认证中的复杂工程问题凝练,以及解决复杂工程问题能力提供了很好的示范。

高水平的教材是一流专业教育质量的重要保证。在总结系统能力培养教学改革探索与实践经验的基础上,国内部分高校也组织了计算机专业教材编写。高等教育出版社为《研究报告》的研制以及出版这批具有创新实践性的系列教材提供了支持。这些教材以强化基础、突出实践、注重创新为原则,体现了计算机专业课程体系的整体性与融通性特点,突出了教学分析方法与综合方法的结合,以及系统能力培养教学改革的新成果。相信这些教材的出版,能够对我国高校计算机专业课程改革与建设起到积极的推动作用,对计算机专业工程教育认证实践起到很好的支撑作用。

教育部高等学校计算机类专业教学指导委员会秘书长

马殿富

2016年7月

前 言

本书是在 2011 年版教材《编译原理及编译程序构造》(张莉、杨海燕、史晓华、金茂忠、高仲仪编著)基础上改编而成。2011 版教材未来得及对教材整体结构进行修改,仅在前言中给出授课建议,本次成书则进行了较大调整。为简洁起见,我们将教材更名为《编译技术》,意在强调编译系统的构造及其相关技术,突出工科类院校对人才培养的要求。

本书仍然定位为本科生的第一门编译课程的教材,因此着重介绍编译领域涉及的基础和通用原理,对于现代编译器的一些工程化方法和优化技术,本书涉及不多。

我们认为编译技术是计算机及软件工程专业的一门非常重要的课程。虽然我们重在介绍编译系统的构造和相关原理与技术,但是在这个过程中,我们介绍并讨论了计算机领域三个非常重要的原理、概念和技术:

(1) 高级程序设计语言的工作原理。

(2) 程序/模型间转换的方法:实现模型从一种语言表达形式到另一种语言表达形式的(语义)等价转换方法及其相关技术。

(3) 软件系统的概念。

毕竟对于大多数本科生而言,编译系统是他们构造的第一个相对完整的软件系统,因此我们非常强调编译过程和编译系统的完整性。

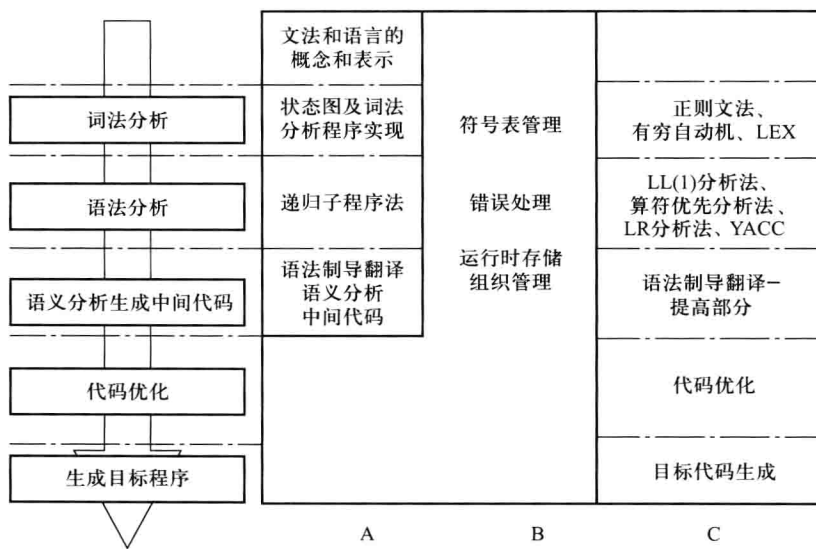
编译原理是公认的比较难学亦难教的一门课,因为有太多理论性的东西,同时实践性也很强。经过十多年的教学,我们认为这门课程应该不那么难学,因为编译的主线已了然在心。可为什么大家觉得难学呢?因为我们开始讲了太多的理论,从文法基础到词法分析部分的自动机,再到各种语法分析方法,学生被灌输了很多算法,反而忽略了编译的根本任务——翻译,即程序从一种语言形式到另一种语言形式的等价翻译。这就是为什么学生往往在完成课程设计(一个小型编译系统)之后,才恍然大悟:我有点明白编译是干什么的了。

这也促使我们下决心对编译课程的讲解进行结构性改革。首先暂时抛开大量形式化方法,先讲一个完整的编译过程,以及这个过程中涉及的编译技术,以便学生深刻地理解高级程序语言的运行机理,理解一个完整地将高级程序设计语言编写的程序翻译为低级语言程序的过程。因为编译技术作为一门重要的专业基础课程,它讲述的是计算机领域中一个重要的主题:程序从一种语言形式到另一种语言形式的等价转换。同

时,它还是软件和硬件之间的一个桥梁,帮助学生理解高级语言在计算机上运行的机理。因此,我们有责任给学生展示一个完整的编译过程。在这个过程的基础上或其过程中,介绍相关的理论和方法。对于编译过程中涉及的形式化方法、编译自动生成技术、编译优化技术等,我们将其放在一个完整的编译过程之后,作为必要的补充。

为此,本书按照以下结构进行组织,并建议第一部分作为必修环节;第二部分,即提高部分,各学校可根据情况选讲全部或者部分;第三部分的两个实践案例可作为学生完成编译实践作业的参考资料。

第一部分基础篇(1~10章),包含编译技术概述、语言与文法基础、一个简单编译器的构造(一个完整的编译过程),重点讲解一个完整的高级语言程序翻译过程。因此,我们选用了—个伪代码形式 P-code 作为中间代码,完成高级语言程序到低级语言程序的翻译,从而给学生展示一个完整的最简单的编译过程(翻译过程),见下图中 A 和 B 列对应的知识点。



编译过程和知识点组织

第二部分提高篇(11~16章),重点介绍编译程序的自动化生成技术和代码优化及面向目标机的代码生成技术。各学校可以根据培养要求选讲本部分内容。

本部分内容对应于上图中 C 列,包括以下几方面:

- (1) 词法分析的自动生成(正则文法、有穷自动机和 LEX)。
- (2) 语法分析方法及分析器的自动生成技术(LL(1)分析法、算符优先分析法、LR

分析法、YACC)。

(3) 代码优化及面向目标机的代码生成技术。

(4) 语法制导翻译(提高部分)作为选修内容。

第三部分实践案例(17~18章),给出两个小型编译系统的完整设计。实现代码见教材配套的课程网站。

建议采用本书的任课教师根据本校的培养定位选择讲课内容。

本书凝聚了北京航空航天大学编译课程组全体成员的心血,由几代师生共同完成。具体编写分工为:张莉负责全书的整体结构设计;金茂忠、张莉负责编写第1~4章及第11、12章,张莉负责编写第9、10章,杨海燕负责编写第5、8、16~18章,史晓华负责编写第6、7、13~15章;最后由张莉对全书进行了审校。

本书在编写过程中参考和引用了国内外优秀编译教材和著作中的相关内容,在此谨向原书著(译)者深表敬意和感谢。同时感谢邵兵老师为本书提出的宝贵意见。

本书难免有疏漏和差错之处,欢迎读者批评指正,以便再版时及时修正。作者的联系方式为 lily@buaa.edu.cn。本书相应的国家精品资源共享课程网站地址为 http://www.icourses.cn/coursestatic/course_3288.html。

张莉

2016年7月

目 录

第 1 部分 基 础 篇

第 1 章 编译概述	3	2.2.2 符号串及其集合的 运算	27
1.1 什么是程序设计语言	3	练习 2-2	29
1.1.1 程序设计语言的定义 方法	4	2.3 文法和语言的形式定义	29
1.1.2 程序设计语言的处理 系统	5	2.3.1 文法的形式定义	29
1.1.3 编译程序和解释程序	5	2.3.2 推导的形式定义	31
1.2 与编译程序相关的处理 系统	8	2.3.3 语言的形式定义	32
1.3 编译程序和程序设计 环境	10	2.3.4 递归规则与递归文法	35
1.4 编译程序的构造	12	2.3.5 短语、简单短语和 句柄	36
1.5 编译技术在软件工程中的 应用	19	练习 2-3	38
练习 1	21	2.4 语法树和二义性	39
第 2 章 文法和语言的概念和 表示	22	2.4.1 推导与语法树	39
2.1 文法的非形式讨论	22	2.4.2 文法的二义性	43
2.1.1 语法树	22	练习 2-4	46
2.1.2 规则	23	2.5 符号串的分析	48
2.1.3 由规则推导句子	24	2.5.1 自顶向下分析	49
练习 2-1	26	2.5.2 自底向上分析	50
2.2 符号、符号串及其集合 的运算	27	2.6 有关文法的实用限制	51
2.2.1 字母表和符号串	27	练习 2-5	53
		2.7 扩充的 BNF 表示和 语法图	53
		2.7.1 扩充的 BNF 表示	53
		2.7.2 语法图	56
		2.8 文法和语言分类	57

第3章 词法分析程序的设计	60	5.2 符号表的组织和内容	97
3.1 词法分析程序的功能及 实现方案	60	5.2.1 符号表的结构与内容 ...	97
3.2 单词的种类及词法分析 程序的输出形式	61	5.2.2 符号表的组织方式	99
3.3 正则文法及其状态图	63	5.3 非分程序结构语言的符号 表组织	101
3.3.1 状态图	63	5.3.1 标识符的作用域及基本 处理方法	101
3.3.2 状态图的使用	64	5.3.2 符号表的组织方式	102
3.4 词法分析程序的设计与 实现	65	5.4 分程序结构语言的符号表 组织	109
3.4.1 文法及其状态图	65	5.4.1 标识符的作用域及基本 处理方法	109
3.4.2 词法分析程序的构造 ...	67	5.4.2 定位和重定位操作	110
3.4.3 词法分析程序的实现 ...	70	5.4.3 符号表的组织方式	112
练习3	73	练习5	115
第4章 语法分析(一)	74	第6章 运行时的存储组织及管理 ...	117
4.1 自顶向下分析方法	74	6.1 静态存储分配	117
4.1.1 带回溯的自顶向下 分析方法	74	练习6-1	119
4.1.2 存在的问题及解决 办法	76	6.2 动态存储分配	120
练习4-1	84	6.2.1 活动记录	121
4.2 递归下降分析法	84	6.2.2 参数区	121
4.3 基于递归下降分析法的 语法分析程序构造	85	6.2.3 display 区	122
练习4-2	91	6.2.4 运行时的地址计算	125
第5章 符号表管理技术	93	6.2.5 递归过程的处理	125
5.1 概述	93	6.3 内存垃圾回收器	128
5.1.1 符号表的概念及建立和 访问时间	93	6.3.1 引用计数	128
5.1.2 符号表的重要性和 作用	95	6.3.2 标记和清除垃圾回 收器	129
5.1.3 在符号表上的操作	95	6.3.3 标记紧缩算法	129
		6.3.4 拷贝回收算法	130
		6.3.5 分代垃圾回收器	130
		练习6-2	133

第 7 章 源程序的中间形式	135	翻译	168
7.1 波兰表示	135	练习 9-2	172
7.2 N -元表示	137	9.4.2 属性翻译文法的自顶 向下翻译	173
7.3 抽象语法树	139	练习 9-3	180
7.4 抽象机代码	140	第 10 章 语义分析和代码生成	182
7.4.1 可移植性和抽象机	141	10.1 语义分析的概念	182
7.4.2 Pascal 的 P-code 抽象机	142	10.2 栈式抽象机及其汇编 指令	184
7.4.3 P-code 指令	143	10.3 声明语句的处理	186
练习 7	144	10.3.1 常量类型	188
第 8 章 错误处理	145	10.3.2 简单变量	188
8.1 概述	145	10.3.3 数组变量	190
8.2 错误的分类	146	10.3.4 记录变量	192
8.3 错误的检查与报告	146	10.3.5 过程声明	193
8.4 错误处理技术	148	10.4 表达式语句	194
8.4.1 错误改正	149	10.5 赋值语句	201
8.4.2 错误局部化处理	149	10.6 控制语句	203
8.4.3 目标程序运行时错误 检测与处理	152	10.6.1 if 语句	203
8.4.4 遏止重复的错误信息	152	10.6.2 分情形语句	205
第 9 章 语法制导翻译技术	153	10.6.3 repeat-while 语句	208
9.1 翻译文法	154	10.6.4 for 循环语句	209
9.2 语法制导翻译	156	10.7 过程调用和返回语句	211
9.3 属性翻译文法	158	10.7.1 参数的基本传递形式	211
9.3.1 综合属性	158	10.7.2 过程调用	213
9.3.2 继承属性	160	10.7.3 返回语句和过程 终止	217
9.3.3 属性翻译文法	162	10.8 输入/输出语句	218
9.3.4 属性翻译文法举例:算术 表达式的翻译	164	10.8.1 输入语句	218
练习 9-1	166	10.8.2 输出语句	221
9.4 自顶向下语法制导翻译	168	10.9 编译程序的辅助功能	222
9.4.1 翻译文法的自顶向下 练习 10	223		

第 2 部分 提 高 篇

第 11 章 词法分析程序的自动生成	练习 12-1	264
技术	227	
11.1 正则文法与正则表达式	227	
11.1.1 正则表达式	227	
11.1.2 正则文法转换为正则表达式	230	
11.1.3 正则表达式转换为正则文法	231	
11.2 有穷自动机	231	
11.2.1 确定的有穷自动机	232	
11.2.2 不确定的有穷自动机	233	
11.2.3 NFA 的确定化	235	
11.2.4 确定有穷自动机的化简(最小化)	238	
11.2.5 正则表达式与有穷自动机的等价性	242	
11.2.6 正则文法与有穷自动机的等价性	246	
11.3 词法分析程序的自动生成器	248	
11.3.1 Lex 源程序(Lex 的输入文件)	248	
11.3.2 Lex 的实现	250	
练习 11	254	
第 12 章 语法分析(二)	256	
12.1 LL(1)分析方法	256	
12.1.1 LL(1)分析器的逻辑结构及工作过程	256	
12.1.2 LL(1)分析表的构造方法	259	
12.2 自底向上分析方法	265	
12.3 算法优先分析法	268	
12.3.1 方法概述	268	
12.3.2 直观算符优先分析法	270	
12.3.3 算符优先分析法的进一步讨论	273	
练习 12-2	279	
12.4 LR 语法分析方法	280	
12.4.1 概念和术语	281	
练习 12-3	282	
12.4.2 LR 分析算法	282	
练习 12-4	288	
12.4.3 LR 文法	288	
12.4.4 构造 SLR 语法分析表	289	
练习 12-5	297	
12.4.5 构造规范 LR 语法分析表	299	
练习 12-6	304	
12.4.6 构造 LALR 语法分析表	305	
练习 12-7	309	
第 13 章 语法制导翻译技术(二)*	311	
13.1* LL(1)文法的语法制导翻译	311	
13.1.1* 翻译文法的自顶向下翻译——LL(1)翻		

译器	311	14.3.6 死代码删除	353
练习 13-1	313	14.4 循环优化	353
13.1.2* 属性文法自顶向下 翻译的实现——下 推机法	314	14.4.1 循环交换	354
练习 13-2	320	14.4.2 循环展开	354
13.2* 自底向上语法制导 翻译	321	14.4.3 代码外提和循环强度 削弱	355
13.2.1 波兰翻译	322	练习 14	356
13.2.2 S-属性文法	323	第 15 章 目标代码生成及优化	357
练习 13-3	325	15.1 微处理器体系结构 简介	358
第 14 章 代码优化	327	15.1.1 指令集架构	359
14.1 基本块和流图	328	15.1.2 存储层次架构	362
14.2 基本块内优化	330	15.1.3 流水线	364
14.2.1 基本块的 DAG 图 表示	331	15.2 地址空间	367
14.2.2 消除局部公共子 表达式	331	15.2.1 程序地址空间的 实例分析	369
14.2.3 数组、指针及函数 调用	332	15.2.2 程序运行栈的设计	371
14.2.4 从 DAG 图重新导出 中间代码	334	15.3 寄存器的分配和指派	374
14.2.5 窥孔优化	336	15.3.1 全局寄存器分配	374
14.2.6 常数合并和传播	337	15.3.2 临时寄存器分配	377
14.3 全局优化	338	15.4 指令选择	379
14.3.1 数据流分析	338	练习 15	381
14.3.2 活跃变量分析	344	第 16 章 编译程序生成方法和 工具	383
14.3.3 定义-使用链、网和 冲突图	346	16.1 编译程序的书写语言	383
14.3.4 消除全局公共子表 达式	351	16.2 自展	384
14.3.5 复制传播	352	16.3 移植	385
		16.4 编译程序的生成工具	386
		16.4.1 语法分析器的生成器 YACC	387
		16.4.2 用 YACC 处理二义 文法	390

16.4.3 用 Lex 建立 YACC 的 词法分析器	393	16.4.4 YACC 的错误恢复	394
练习 16		练习 16	396

第 3 部分 实 例 篇

第 17 章 PL/0 简单编译系统	399	18.3.1 表格	439
17.1 PL/0 语言	399	18.3.2 编译初启	446
17.2 PL/0 编译系统结构	404	18.3.3 实用程序	447
17.3 PL/0 的词法分析	406	18.3.4 词法分析及处理	448
17.4 PL/0 的语法分析	407	18.3.5 语法分析处理	449
17.5 出错处理	409	18.3.6 出错处理	455
17.6 目标代码的生成和解释 执行	411	18.4 Pascal-S 解释执行 程序	458
17.7 PL/0 程序编译和运行 举例	414	18.4.1 P 代码指令系统	458
第 18 章 Pascal-S 编译系统	427	18.4.2 运行栈	461
18.1 Pascal-S 语言	427	18.4.3 运行时的 display	462
18.2 Pascal-S 编译程序的 结构	434	18.4.4 运行出错处理和现场 剖析打印	464
18.3 Pascal-S 编译程序	439	18.5 编译及运行的例子	465
参考文献	479		

第 1 部分 基 础 篇

把高级程序设计语言翻译成汇编语言或机器语言的工作称为编译,完成这项翻译工作的软件系统称为编译程序或编译器。本书介绍编译原理及编译器的设计和实现技术,学习编译原理和技术也是深入理解程序设计语言的工作原理和设计高质量程序的基础。

基础篇为必修部分,介绍编译的基础概念和一个完整的翻译过程,即将高级语言编写的程序翻译为一个汇编伪代码(P-code)的完整过程。在此基础上,培养学生具备编写一个简单编译程序的能力,以及深入理解高级语言程序运行原理的能力。

