

计算机专业考研指导丛书

编译原理

考点精要与解题指导

伍春香 编著

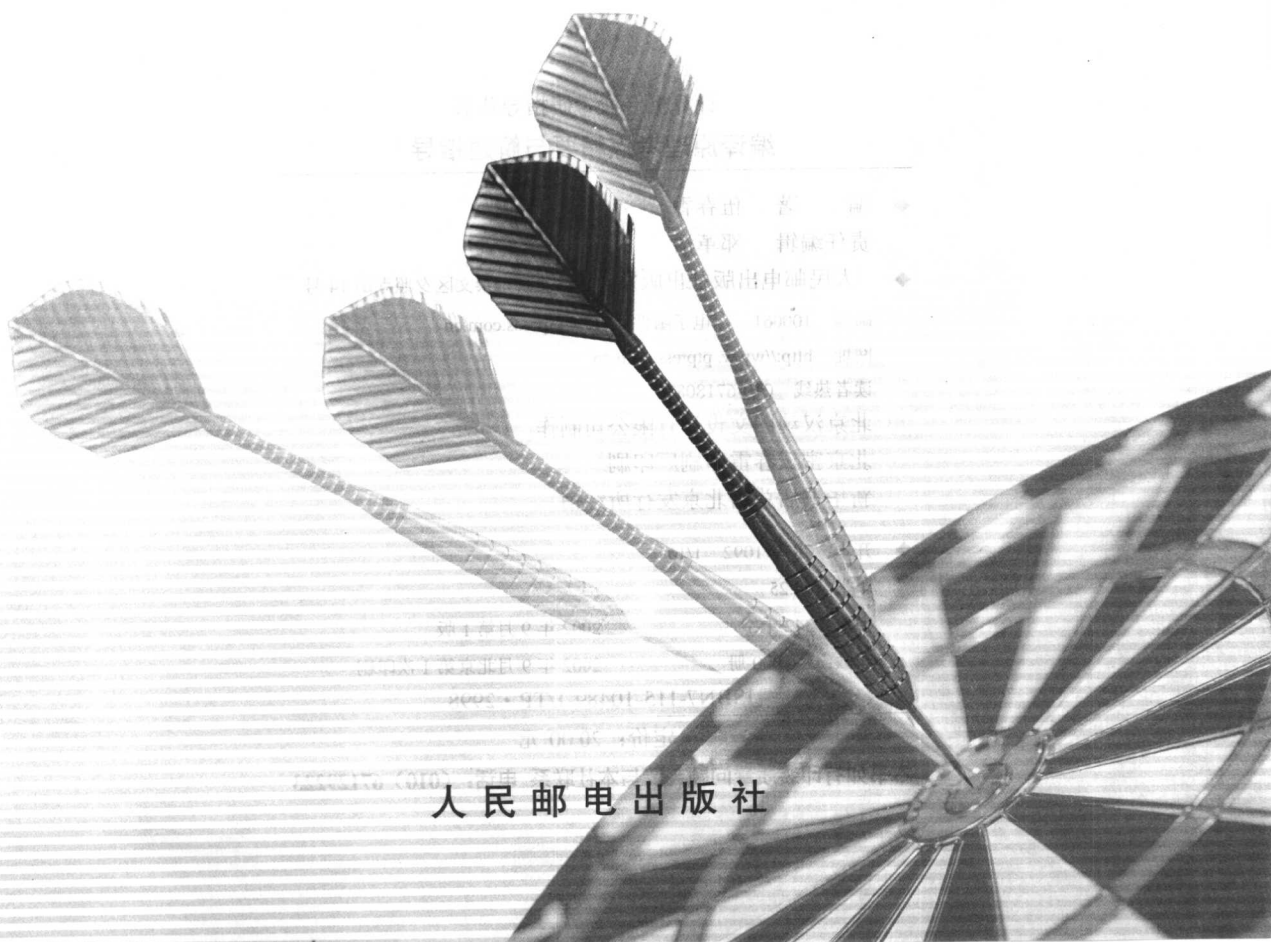
人民邮电出版社
POSTS & TELECOMMUNICATIONS PRESS

计算机专业考研指导丛书

编译原理

考点精要与解题指导

伍春香 编著



人民邮电出版社

图书在版编目(CIP)数据

编译原理考点精要与解题指导 / 伍春香编著. —北京: 人民邮电出版社, 2002.9

(计算机专业考研指导丛书)

ISBN 7-115-10489-1

I. 编... II. 伍... III. 编译程序—程序设计—研究生—入学考试—自学参考资料
IV. TP314

中国版本图书馆 CIP 数据核字(2002)第 063332 号

计算机专业考研指导丛书 编译原理考点精要与解题指导

- ◆ 编 著 伍春香
责任编辑 邓革浩
- ◆ 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>
读者热线 010-67180876
北京汉魂图文设计有限公司制作
北京密云春雷印刷厂印刷
新华书店总店北京发行所经销
- ◆ 开本: 787×1092 1/16
印张: 14.25
字数: 342 千字 2002 年 9 月第 1 版
印数: 1-6 000 册 2002 年 9 月北京第 1 次印刷

ISBN7-115-10489-1/TP · 2998

定价: 20.00 元

本书如有印装质量问题, 请与本社联系 电话: (010) 67129223

内容提要

本书针对课程内容的重点和难点，选取多年来各类教材以及各高校、科研院所考研试题中有代表性的题目进行了分析解答，力求从发现问题考点、理清解题思路、掌握基本方法等方面对学习者给予帮助。

本书以编译原理教学大纲为指导，针对考研的特点进行内容安排，全书共分 8 章，内容包括：编译程序概述、文法和语言的形式定义、词法分析与有穷自动机、自上而下语法分析、自下而上语法分析、语法制导翻译和中间代码生成、运行阶段的存储组织与分配和代码优化。每一章具体内容分为考点精要、例题解析、自测题与参考答案 3 部分。

本书的特点是概念准确，文字简洁明了，解题思路完整，极便于考研者短时间内掌握解题要点，提高考试成绩。

本书不仅可作为研究生入学考试的复习参考书，也可作为计算机专业本科生的学习辅导书，对于参加其他相关考试的人员来说，也有一定的参考价值。

计算机专业是当今最热门也是发展最迅速的学科之一，很多学生为了进一步提高专业水平和应用能力，纷纷报考计算机专业研究生。据统计，近几年报考计算机软件与理论、计算机应用和计算机与通信专业硕士研究生的考生远远超过报考其他专业的考生，其中有相当一部分考生原来所学专业并非计算机专业，还有很多考生是工作多年的在职人员。为了方便报考者复习计算机专业课程，我们特地组织一批计算机专业教学第一线的教授和副教授（其中大多数编写者多年参加硕士研究生入学试题命题工作）编写了本丛书。本丛书包含如下课程：

- 《C 程序设计考点精要与解题指导》
- 《离散数学考点精要与解题指导》
- 《数据结构考点精要与解题指导》
- 《操作系统考点精要与解题指导》
- 《编译原理考点精要与解题指导》
- 《计算机组成原理考点精要与解题指导》

本丛书具有以下特点：

◎ 讲述全面而详实

本丛书涵盖各门专业课程的内容，不是针对个别学校的命题特点，而是充分地讲授课程中的重点、难点和考点，并通过例题进行扩充与深化，使读者得以全面温习，不留“死角”。

◎ 阐述简洁而明了

不同于本、专科教材，本丛书的目的是使考生花较少的时间温习各门课程的内容，因此，不过多地解释简单的术语，只对基本知识点进行高度概括和总结，使读者将主要精力花在解题过程上。

◎ 重点突出解题思路

本丛书重点介绍解题的方式和方法，不仅授人以“鱼”，更在于授人以“渔”，选择的例题和习题大多是计算机专业研究生入学考试试题（题目前标有“★”号），并配上详解，具有很强的实战性。

◎ 强调内容的综合与提高

一般的教科书多是按照内容的先后顺序按部就班地介绍，这种方式有助于初学者学习，但不便于复习和综合，因为考研题一般都具有很强的综合性，往往一个题涉及好几章的概念，所以本丛书打破了一般教科书的教学模式，将相关的概念有机地融为一体，从而提高考生的解题能力。

◎ 答疑解惑

本丛书选择的例题和习题大部分具有较高的难度，书中不仅给出了答案，而且详细介绍了解题思路和解题过程，有助于考生纠正以往的概念误区。

本丛书希望在考研指导方面作一些探索和尝试，起到抛砖引玉的作用，书中的不妥之处敬请广大的读者和同行指教。

李春葆

2002.6

前 言

编译原理是计算机专业一门非常重要的专业基础课，通常作为计算机专业本科生的必修课或专业限选课来设置，在各高校、科研院所的计算机专业研究生入学考试科目中，它也占有比较重要的地位。编译原理的课程内容涉及形式语言、有穷自动机等较抽象的内容，对学习者的理论掌握和实践能力的要求都较高，使得众多学习者感到有一定的难度。

本书针对课程内容的重点和难点，选取多年来各类教材以及各高校、科研院所考研试题中有代表性的题目进行了分析解答，力求从发现问题考点、理清解题思路、掌握基本方法等方面对学习者的给予帮助。

本书以编译原理教学大纲为指导，针对考研的特点，将内容分成8章。第1章为编译程序概述，第2章介绍文法和语言的形式定义，第3章介绍词法分析与有穷自动机，第4章介绍自上而下语法分析，第5章介绍自下而上语法分析，第6章介绍语法制导翻译和中间代码生成，第7章介绍运行阶段的存储组织与分配，第8章介绍代码优化。

每一章除了基础知识点、重点和难点的提示外，具体内容分为考点精要、例题解析、自测题与参考答案3部分。考点精要部分简单摘要了各章知识点——基本概念、基本原理和基本方法；例题解析部分选取了对各章重点、难点进行考查的有代表性的试题和习题，并对每一道题的求解给出了详细的分析与解答，使读者从解题方法上得到有效的帮助；自测题部分完全针对考研的需要进行选取，主要来自历年各高校、科研院所的考研试题以及国内比较权威的教材。

本书不仅可作为研究生入学考试的复习参考书，也可作为计算机专业本科生的学习辅导书，对于参加其他相关考试的人员来说，也有一定的参考价值。

由于编者水平有限，书中难免有错漏之处，诚请广大读者批评指正！

编 者

2002. 6

目 录

第 1 章 编译程序概述	1
1.1 考点精要	1
1.1.1 程序设计语言的分类	1
1.1.2 翻译程序	1
1.1.3 编译方式	2
1.1.4 解释方式与解释程序	3
1.1.5 编译程序的工作过程	3
1.1.6 编译程序的逻辑结构	4
1.1.7 编译程序的构造	6
1.2 例题解析	6
1.3 自测题及参考答案	7
第 2 章 文法和语言的形式定义	8
2.1 考点精要	8
2.1.1 上下文无关文法	8
2.1.2 文法描述的语言	8
2.1.3 文法的分类	9
2.1.4 语法树与二义性	9
2.1.5 短语、简单短语与句柄	10
2.1.6 文法的实用限制和文法变换	10
2.2 例题解析	11
2.3 自测题及参考答案	18
第 3 章 词法分析与有穷自动机	33
3.1 考点精要	33
3.1.1 词法分析程序的任务	33
3.1.2 词法分析方法	33
3.1.3 状态转换图	33
3.1.4 正规表达式	34
3.1.5 有穷自动机	35
3.1.6 由正规表达式构造确定的有穷自动机	36
3.1.7 正规文法 G 到有穷自动机 A 的转换	38
3.1.8 有穷自动机 FA A 到正规文法 G 的转换	38
3.1.9 由有穷自动机到正规表达式的转换	38
3.1.10 由正规文法到正规表达式的转换	39



3.1.11 词法分析程序的设计方法	39
3.2 例题解析	40
3.3 自测题及参考答案	46
第4章 自上而下语法分析	73
4.1 考点精要	73
4.1.1 自上而下语法分析	73
4.1.2 自上而下语法分析方法遇到的问题	73
4.1.3 递归子程序法	75
4.1.4 预测分析法(LL(1)方法)	76
4.2 例题解析	76
4.3 自测题及参考答案	82
第5章 自下而上语法分析	95
5.1 考点精要	95
5.1.1 自下而上语法分析	95
5.1.2 简单优先分析法	96
5.1.3 算符优先分析法	97
5.1.4 优先函数	100
5.1.5 LR 分析法	101
5.2 例题解析	105
5.3 自测题及参考答案	118
第6章 语法制导翻译和中间代码生成	147
6.1 考点精要	147
6.1.1 语义分析	147
6.1.2 语法制导翻译	147
6.1.3 属性文法	148
6.1.4 常见中间代码形式	149
6.2 例题解析	149
6.3 自测题及参考答案	161
第7章 运行阶段的存储组织与分配	172
7.1 考点精要	172
7.1.1 基本思想	172
7.1.2 过程的活动与活动记录	172
7.1.3 静态存储分配	173
7.1.4 动态存储分配	173
7.1.5 简单的栈式存储分配	174
7.1.6 嵌套结构语言的栈式动态存储分配方案	174
7.1.7 堆式存储分配	177

7.1.8 参数的传递方式及其实现	177
7.2 例题解析	178
7.3 自测题及参考答案	185
第8章 代码优化	201
8.1 考点精要	201
8.1.1 程序的优化	201
8.1.2 代码优化的种类	201
8.1.3 基本块内的优化	202
8.1.4 循环优化	203
8.2 例题解析	204
8.3 自测题及参考答案	210
参考文献	218

第1章 编译程序概述

基本知识点：编译方式的特点，编译程序的功能、工作过程、结构及构造方法。

重点：编译方式的特点，编译程序的结构。

难点：编译程序的结构，编译方式与解释方式的区别。

1.1 考点精要

1.1.1 程序设计语言的分类

计算机必须接受以程序方式授予的指令，才能完成人们所要求的工作。用以书写程序的程序设计语言通常分成3种情形。

1. 机器语言

机器语言是指可以直接执行的代码形式的指令系统，它全部是由二进制数字0和1表示的数和指令构成。

2. 汇编语言

汇编语言是指计算机的符号形式的指令系统。例如用 `ADD X Y` 表示 X 与 Y 相加，用 `MUL X Y` 表示 X 与 Y 相乘，用 `CAL P` 表示调用 P 等。

3. 高级语言

高级语言是指与自然语言比较接近、具有丰富的数据结构和控制结构等特点的程序设计语言，诸如 `ALGOL`、`FORTRAN`、`PASCAL`、`Ada`、`C` 等。

1.1.2 翻译程序

计算机只能识别0和1两种状态。因此，计算机只懂用二进制数0、1表示的指令和数构成的本台机器的机器语言。所以，除了机器语言程序，其他语言书写的程序都必须翻译成某台机器的机器语言才能被这台计算机识别，这一过程由翻译程序来完成。

翻译程序能将用甲语言编写的程序翻译成与之等价的用乙语言书写的程序。甲语言称为该翻译程序的源语言，乙语言称为该翻译程序的目标语言。用源语言书写的程序称为源程序，与源程序等价的用目标语言编写的程序称为目标程序。



翻译程序对源程序的等价翻译，指的是程序功能上的等价翻译，即生成
注意 的目标程序与源程序在功能上是等价的。

翻译程序对程序的翻译通常有两种方式：一种是“编译”方式，另一种是“解释”方式。二者的根本区别在于是否生成目标代码。

1.1.3 编译方式

编译方式是一种分阶段进行的方式。一般说来，首先进行“翻译”，把用高级语言或汇编语言编写的程序翻译成与之等价的用机器语言书写的程序，然后对翻译出来的程序进行运行计算。前一阶段的翻译工作由翻译程序完成，后一阶段的运行计算需要有运行程序来配合完成。

如果翻译程序是将用高级语言书写的源程序翻译成与之等价的某计算机的汇编语言或机器语言的程序，则翻译程序称为编译程序。

如果翻译程序是将汇编语言的程序翻译成某台计算机的机器语言程序，则翻译程序称为汇编程序。

用机器语言构成的目标程序又称为目标代码程序或简称为代码程序，有时又称为目标代码或结果代码。

编译方式有如下一些特点。

(1) 在编译方式下，源程序的执行需要分阶段。

如果编译程序生成的目标程序是机器代码程序，则源程序的执行分为两大阶段：编译阶段和运行阶段，如图 1.1 所示。

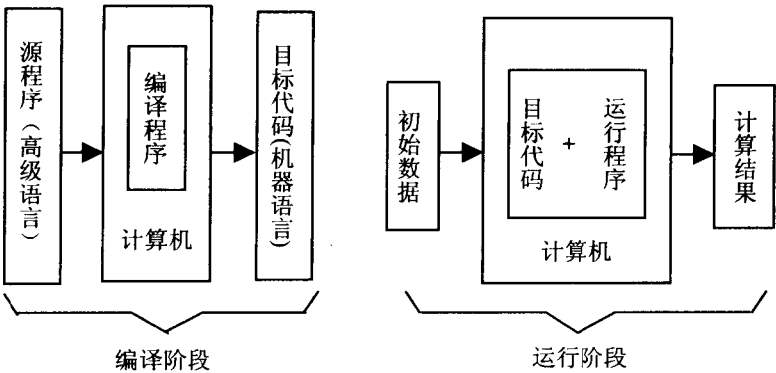


图 1.1 分成两个阶段的源程序的执行历程

如果编译程序生成的目标程序是汇编语言的程序，则源程序的执行分成 3 个阶段：编译阶段、汇编阶段和运行阶段，如图 1.2 所示。

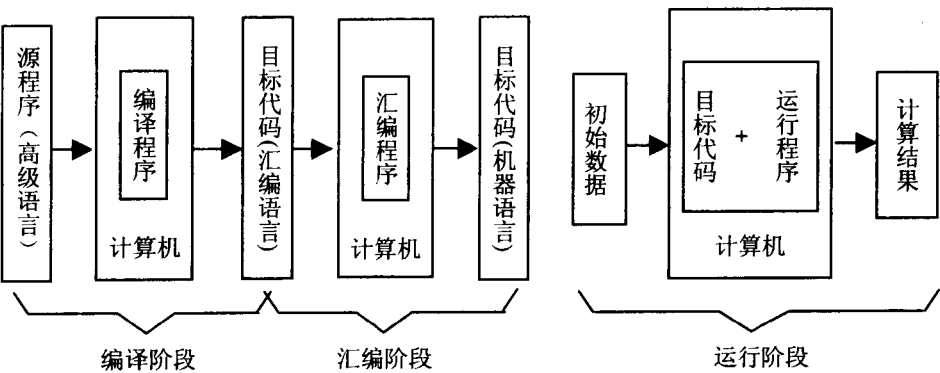


图 1.2 分成 3 个阶段的源程序的执行历程



(2) 在编译方式下, 对源程序进行翻译处理后会生成一种新的程序——目标代码。



注意 编译程序生成的目标程序不一定是机器语言的程序, 也有可能是汇编语言程序;

(2) 编译程序与具体的机器和语言有关, 即任何一个具体的编译程序都是某一特定类型的机器上关于某一特定语言的编译程序;

(3) 对编译程序而言, 源程序是被加工的对象, 目标程序是加工后的结果。即, 源程序是输入数据, 目标程序是输出结果。

(4) 所谓运行程序是指运行目标代码程序时必须配置的各种子程序的全体, 通常以库子程序的形式存在, 如一些连接装配程序及一些连接库等。

1.1.4 解释方式与解释程序

在解释方式下, 源程序的执行只有一个阶段——解释执行阶段。具体地说, 完成解释工作的解释程序将按源程序中语句的动态顺序, 逐句地进行分析解释, 并立即予以执行。在解释方式下, 源程序的执行历程, 如图 1.3 所示。

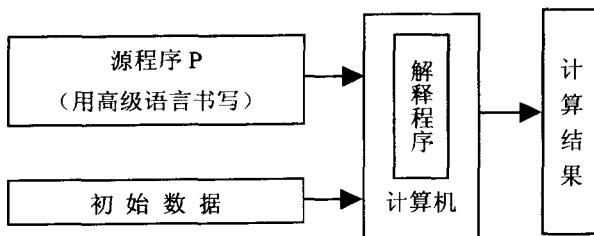


图 1.3 源程序解释执行的历程



注意 解释程序和编译程序对源程序都进行了翻译, 但它们对源程序的翻译方式是不同的。虽然有的解释程序也生成中间形式, 但在解释方式下, 是直接执行源程序本身或与源程序等价的中间程序, 并不生成目标代码, 而编译方式是一定要生成目标代码的, 这是编译方式与解释方式的根本区别。

在解释方式下执行源程序, 易于查错, 在程序执行中可以修改程序。但和编译方式相比, 执行效率太低。二者相结合, 可得到快速、灵活的集成开发环境。

1.1.5 编译程序的工作过程

编译程序的主要功能是将源程序翻译成与之等价的目标程序, 这个翻译过程可以分成词法分析、语法分析、语义分析和中间代码生成、代码优化和目标代码生成等阶段。

1. 词法分析

在词法分析阶段, 完成词法分析工作的词法分析程序将依据语言的词法规则, 分析由字符组成的源程序, 把它识别为一个一个具有独立意义的`最小语法单位`, 即“单词”, 并识别出与其相关的属性(如该单词是标识符, 是界限符, 还是数, 等等), 再转换成长度上统一的标准形式, 以供其他阶段使用。

2. 语法分析

在语法分析阶段,完成语法分析工作的语法分析程序将依据语言的语法规则,逐一分析词法分析时得到的单词,把单词串分解成各类语法单位,即确定这些单词是怎样组成说明和语句,以及说明和语句又是怎样组成程序的。语法分析程序在分析时如发现源程序中有不合语法规则的地方,便将出错的位置及出错性质打印报告给程序员;如源程序无语法错误,语法分析程序则用另一种中间形式给出源程序正确的语法结构,供下一阶段分析使用。

3. 语义分析和中间代码生成

在语义分析阶段,语义分析程序将依据语言的语义规则对语法分析得到的语法结构进行静态语义检查(确定类型、类型和运算合法性检查、识别含义与相应的语义处理及其他一些静态语义检查),并用另一种内部形式表示出来,即生成源程序的中间代码形式,或者直接用目标语言表示出来。

4. 代码优化

在代码优化阶段,代码优化程序将依据程序的等价变换规则,尽量压缩目标程序运行所需的时间和所占的存储空间,以提高目标程序的质量。

5. 代码生成

如果语义分析时把源程序表示成中间代码形式而不是表示成目标指令,则在代码生成阶段完成从中间形式到目标指令的转换;如果语义分析时,已直接生成目标指令,则无需做代码生成工作。目标指令可能是绝对指令代码,或可重新定位的指令代码或汇编指令代码。该阶段的工作有赖于硬件系统结构和机器指令含义。

6. 表格与表格管理

编译程序在工作过程中需要使用一些表格,用以登记源程序中出现的每个名字以及名字的各种属性和编译过程中各阶段的进展情况。这些名字的属性需要在各个阶段才能填入。

7. 出错处理

编译程序不仅应能对正确的源程序进行翻译,而且还应能对源程序中的错误进行处理。编译过程的每一阶段都可能检测出错误,其中,绝大多数错误可以在编译的前3个阶段检测出来。源程序中的错误一般分成为语法错误和语义错误两大类。

语法错误是指源程序中不符合语法(或词法)规则的错误,它们可在词法分析或语法分析时检测出来。

语义错误是指源程序中不符合语义规则的错误,一般的语义错误能够在语义分析时检测出来,称为静态语义错误,还有一些语义错误要在运行时才能检测出来称为动态语义错误。语义错误通常包括:说明错误、作用域错误、类型不一致、等等。



注意 编译程序要完成的任务分别由词法分析程序、语法分析程序、语义分析程序、代码生成程序、代码优化程序来实现。在这些任务中,词法分析、语法分析、语义分析、代码生成是必须的;而代码优化是为了提高目标程序质量进行的,中间代码生成是为了便于优化而进行的,所以它们不是必需的。

1.1.6 编译程序的逻辑结构

上述介绍的编译过程的各个阶段是从逻辑功能上进行划分的,在具体实现一个编译程序

时,还要根据源语言的特点、编译程序功能上的要求、编译程序的使用对象、编译程序的运行环境等诸多因素,将编译程序组织成若干遍。

1. 遍(趟,趟程)

所谓一趟或一遍是指一个编译程序在编译时刻把源程序或源程序的等价物(中间程序)从头到尾扫描一遍并转换成另一紧邻的等价物的全过程。可以将编译程序的几个不同阶段组织成为一遍,也可以把一个阶段的工作分成若干遍完成。

2. 单遍扫描与多遍扫描

根据编译程序在完成翻译任务的过程中需要对源程序或其中间等价物扫描的遍数,可以把编译程序分为单遍扫描的编译程序(只需扫描一遍)和多遍扫描的编译程序(需扫描多遍)。

在单遍扫描的编译程序中,语法分析程序处于核心地位,词法分析程序往往作为语法分析的一个子程序,当语法分析需要一个新单词时,便调用词法分析程序从源程序中获取单词,一旦语法分析程序识别出一个语法成分,它就调用中间代码生成程序,完成语义分析并生成相应的中间代码。

3. 典型的编译程序结构

如前所述,一个典型的编译程序包括词法分析、语法分析、语义分析和中间代码生成、代码优化和目标代码生成等阶段。相应地,编译程序的结构也可分成5个程序模块,即词法分析程序、语法分析程序、语义分析和中间代码生成程序、代码优化程序和目标代码生成程序。此外,编译程序中还包括表格处理程序和出错处理程序。编译程序典型的结构图,如图1.4所示。

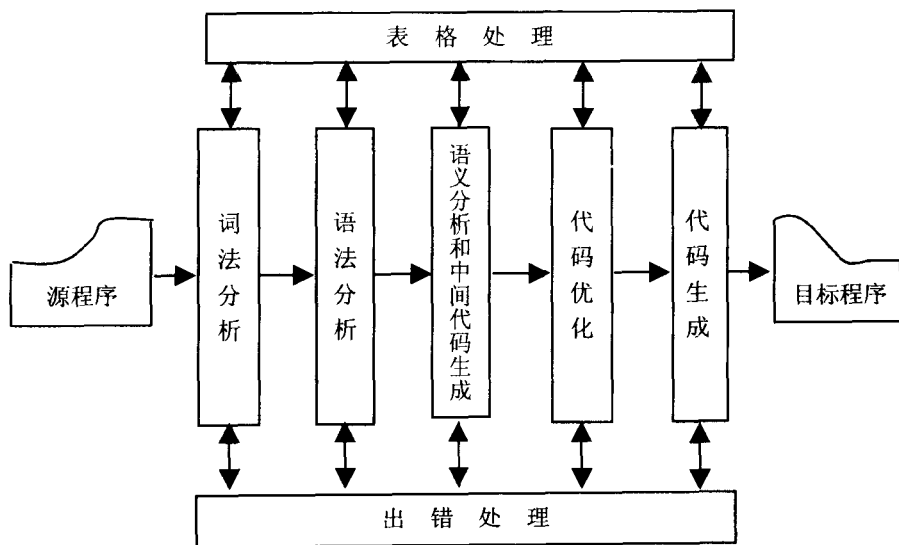


图 1.4 典型的编译程序结构



不同的编译程序的结构不一定相同,在图 1.4 中给出的只是编译程序各部分的逻辑结构图,它并不代表着时间上的执行顺序。有些编译程序可能恰好按图中的顺序执行这些逻辑过程,而有一些编译程序可能按平行、互锁方式执行它们。

1.1.7 编译程序的构造

构造编译程序可以使用机器语言、汇编语言和高级语言。

构造编译程序的方法通常有移植和自编译两种。

• 移植——这是一种利用 A 机器上的高级语言 L 编写能在 B 机器上运行的高级语言 L 的编译程序的方法。其过程如图 1.5 所示。

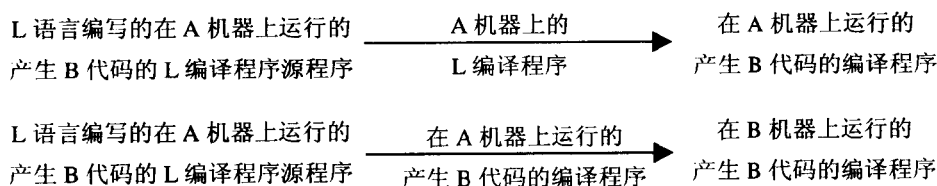


图 1.5 用移植方法构造编译程序

• 自编译——这是一种通过一系列自展途径而形成编译程序的过程。

按照自编译方式构造编译程序，需要先对语言的核心部分构造一个小小编译程序（可用低级语言实现），再以它为工具构造一个能够编译更多语言成分的较大编译程序，如此扩展下去，越滚越大，最后形成所期望的整个编译程序。

1.2 例题解析

例 1.1 ★试问编译程序从逻辑结构上讲，一般由哪几个部分组成？

【相关知识】 编译程序的组成。

【例题分析】 “逻辑结构”所需要的解答仍然是编译程序的组成。作为编译程序的各个组成部分一直是从逻辑功能上予以描述的，因此在此处只要给出编译程序的典型结构即可，可用图的方式给出，也可用文字一一列举。

【例题答案】 从逻辑结构上讲，一个典型的编译程序由词法分析程序、语法分析程序、语义分析程序、中间代码生成程序、代码优化程序和目标代码生成程序组成。此外，编译程序中还包括表格处理程序和出错处理程序。

（本题也可直接给出图 1.4 作为解答。）

例 1.2 ★汇编程序是将_____翻译成_____；编译程序是将_____翻译成_____。

【相关知识】 翻译程序的功能。

【例题分析】 编译方式下需要翻译程序将源程序翻译成相应的目标代码。如果翻译程序是将用高级语言书写的源程序翻译成与之等价的某计算机的汇编语言或机器语言的程序，则翻译程序称为编译程序。如果翻译程序是将汇编语言的程序翻译成某台计算机的机器语言，则翻译程序称为汇编程序。

在这里要特别注意的是第 4 个空，不要盲目地认为编译程序生成的就只是机器语言的程序，它也可能生成汇编语言的程序，这种情况下再由汇编程序将汇编语言的程序进一步翻译成机器语言的目标程序，但作为编译程序来说，生成的是汇编语言程序。

【例题答案】 汇编程序是将 汇编语言程序 翻译成 机器语言程序；编译程序是将 高级语言程序 翻译成 汇编语言或机器语言程序。

1.3 自测题及参考答案

1. 试问“解释方式与编译方式的区别在于解释程序对源程序并没有真正进行翻译。”这种说法对吗？

解：本题考查两种程序翻译方式的特点。

编译方式和解释方式对源程序实际上都进行了翻译，只是前者要产生目标代码，然后执行目标代码得到运行结果，相当于日常生活中的笔译；而后者是在源程序或与源程序等价的中间代码上直接翻译，相当于日常生活中的口译。编译方式与解释方式的根本区别就在于是否生成目标代码，而不在于是否进行翻译。

所以这种说法不对。

2. ★在编译过程中，语法分析器的任务是_____。

- ① 分析单词是怎样构成的
- ② 分析单词串是如何构成语句和说明的
- ③ 分析语句和说明是如何构成程序的
- ④ 分析程序的结构

解：本题考查编译过程中各个阶段的任务。对于这一部分内容需要在理解的基础上熟记。语法分析阶段主要是在词法分析得到的单词串的基础上完成对源程序语法结构的分析。

②、③所完成的工作共同构成④。所以本题的选择应为②、③、④。

3. ★编译程序的工作过程一般可以划分为_____等几个基本阶段，同时还会伴有_____和_____。

解：本题考查编译程序的工作过程。

编译程序的工作过程一般可以划分为 词法分析、语法分析、语义分析、代码生成、代码优化 等几个基本阶段，同时还会伴有 表格处理 和 出错处理。

4. ★编写一个计算机高级语言的源程序后，到正式上机运行之前，一般要经过_____这几步。

解：本题考查编译方式的特点。

在编译方式下，高级语言的源程序要被执行，首先要被翻译成与之等价的用机器语言书写的程序，然后对翻译出来的程序进行运行计算。前一阶段的翻译工作由翻译程序完成，后一阶段的运行计算需要有运行程序来配合完成。

所谓运行程序是指运行目标代码程序时必须配置的各种子程序的全体，如一些连接装配程序及一些连接库等。用户程序一般由一个主程序和若干个子程序组成，它们可以分别编译或汇编。要执行这样的用户程序，必须把主程序和子程序“连接装配”在一起，使之成为一个能执行的程序。

所以，高级语言程序在编写好后，要经过 编辑、编译、连接 等步骤才能正式运行。

第2章 文法和语言的形式定义

基本知识点：上下文无关文法，推导与归约，语言，文法分类，短语、简单短语与句柄，语法树和二义性。

重点：文法的构造，文法描述的语言，短语与句柄，语法树。

难点：文法的构造，短语、简单短语与句柄。

2.1 考点精要

2.1.1 上下文无关文法

为使源程序能被正确地翻译，产生等价的目标程序，源语言的使用者和实现者都应遵循关于源语言的共同约定。因此，每种程序设计语言都有自己的程序构成规则——语法规则。使用者可依据这些规则，以确定所书写程序的正确形式与结构；实现者则依据这些规则，以确定翻译程序可以接收什么样的程序及怎样翻译该程序。程序设计语言的大多数语法规则可用上下文无关文法进行描述。

一个上下文无关文法 $G[S]$ 一般定义成一个四元组： (V_T, V_N, S, P) ，其中

V_N ——有穷非空的符号集合。其元素代表程序中的语法成分或语法单位，称为非终结符号。

V_T ——有穷非空的符号集合。其元素只在产生式的右部出现，称为终结符号。 $V_T \cap V_N = \emptyset$ 。

S ——称为识别符号（开始符号），它至少要在一条产生式中作为左部出现。

P ——是形如 $U \rightarrow u$ （或 $U ::= u$ ）的产生式的有穷非空的集合。 $U \in V_N, u \in (V_T \cup V_N)^+$ 。

2.1.2 文法描述的语言

从一个程序设计语言文法的开始符号出发，可以通过推导可得到该文法所描述的语言。

1. 直接推导/直接归约

如果 $U \rightarrow u$ 是 G 中的一条产生式，而 x, y 是 V^* 中的任一符号串，则将 $U \rightarrow u$ 作用于符号串 $v = xUy$ 上得到的符号串 $w = xuy$ ，记为

$$xUy \Rightarrow xuy \quad (v \Rightarrow w)$$

这时称符号串 $w = xuy$ 是符号串 $v = xUy$ 的直接推导，或叫符号串 v 直接产生了符号串 w （也称 w 直接归约到 v ）。

2. 推导/归约

设 $u_0, u_1 \cdots u_n (n > 0)$ 均为 V^* 中的符号串，且有

$$v = u_0 \Rightarrow u_1 \Rightarrow u_2 \Rightarrow \cdots \Rightarrow u_{n-1} \Rightarrow u_n = w$$

则称以上序列是长度为 n 的推导，也称 v 产生 w （或称 w 归约到 v ），记为：