

新世纪 理工科研究生入学考试指导丛书

典型题解析与实战模拟

编译原理

◇刘春林 王挺 等编著◇



国防科技大学出版社

2002

新世纪
理工科研究生入学考试指导丛书

编 译 原 理

典型题解析与实战模拟

刘春林 王挺 周会平 钟求喜 编著

国防科技大学出版社
·长沙·

内 容 简 介

本书根据国家教育部制定的计算机专业编译原理课程教学大纲和硕士研究生入学考试要求而编写。全书分为两部分。第一部分为解析篇，内容涉及高级语言及编译程序概述，词法分析，语法分析，语法制导翻译和中间代码产生，运行时存储空间组织，代码优化与目标代码生成等。共分为七章，每一章包括复习提要、考试重点、典型题解和习题精选。大部分例题和习题选自近年来全国十几所在计算机领域著名大学和研究所的研究生入学考试试题。所有习题都给出了详细的参考答案。第二部分为实战篇，包括考研模拟试卷及其参考答案（2份）、考研典型试卷（10份）。

本书适合作为研究生入学考试的复习参考书，也宜作为本科生、自学考试考生和其他人员学习编译原理的辅助用书。

图书在版编目（CIP）数据

编译原理典型题解析与实战模拟/刘春林，王挺等编著. —长沙：国防科技大学出版社，2001.6

（新世纪理工科研究生入学考试指导丛书）

ISBN 7-81024-745-X

I. 编… II. ①刘…②王… III. 编译程序—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（2001）第 029122 号

国防科技大学出版社出版发行

电话：(0731) 4572640 邮政编码：410073

E-mail: gfkdcbs@public.cs.hn.cn

责任编辑：张静 责任校对：罗青

新华书店总店北京发行所经销

国防科技大学印刷厂印装

*

787×1092 1/16 印张：15.75 字数：364 千

2002 年 4 月第 1 版第 2 次印刷 印数：4001—9000 册

*

定价：26.00 元

新世纪理工科研究生入学考试指导丛书

编审委员会

主任委员：

陈火旺（中国工程院院士、全国高校计算机专业教学指导委员会主任委员、国防科技大学教授）

副主任委员：

麦中凡（教育部工科计算机基础课程教学指导委员会副主任委员、北京航空航天大学教授）

侯文永（全国高校计算机专业教学指导委员会委员，上海交通大学电子信息学院副院长、教授）

委 员：

屈婉玲（北京大学计算机系副主任、教授）

王广芳（国防科技大学教授）

陈松乔（中南大学教授）

宁 洪（全国高校计算机专业教学指导委员会委员兼秘书，国防科技大学计算机学院计算机科学与技术系主任、教授）

邹逢兴（国防科技大学教授）

策 划：

潘 生 张 静 石少平

新世纪来临，挑战和机遇共存。作为当代大学生和有志青年，当务之急是积累知识，培养能力，以备将来为祖国为人民服务，实现自身的理想和价值。因而，近年来高校“考研热”不断升温，引人关注。

为满足广大学生考研复习之需，更为了适应培养高素质高水平人才的形势，不少出版社出版了辅导学生深入学习课程的参考书，但多是关于数学、外语、政治等公共基础课的，针对各门专业课的指导书较少，精品更少。鉴于此，国防科技大学出版社经多方调研，全面规划，精心组织作者编写了这套旨在帮助学生学习各门专业课、提高考研应试能力的指导丛书。该套丛书具有以下几大特色：

（一）作者经验丰富，权威性强

本丛书的作者都是经悉心遴选，从事教学、科研、著书多年，某些是在全国有相当影响、所著的教材（或专著）在相应专业使用较广的资深专家教授。他们都是高校硕士或博士指导教师。他们在编写这套丛书时废寝忘食，躬行写作，将自己多年积累的经验、体会凝聚在字里行间，奉献给广大的读者，相信他们的辛勤劳动成果必然会对大家学习有关课程有极大帮助，这正是我们丛书编审委员会最感欣慰的。

（二）题目收集广泛，针对性强

这套丛书紧扣国家教育部制定的课程教学大纲和研究生入学考试要求，合理安排各书内容，条理清晰，详略分明，深入浅出，释疑去惑，并广泛搜集近年全国 20

余所重点高校或研究所考研试卷，加以分析、归纳、提高，使读者既能把握各门专业课程的全貌，又能抓住主脉络，领会其中的主要原理、方法，真正提高能力。

（三）突出实战模拟，操作性强

这套丛书中每本书分解析篇和实战篇。其中解析篇按章分提要、例题、习题、习题解答，分别讲清理论、分析各种解题技巧、提供练习和检验机会，使学生全面掌握课程的概念、原理、方法和技巧，学深、学透。实战篇，提供几份模拟题及其参考答案、多份重点高校近年考研试卷，供学生在课程考试或考研的前夕实景备战，以巩固复习成果，丰富考场经验，增强自信心。这样的结构安排极利于学生使用好本丛书。

国防科技大学出版社、丛书编审委员会和编写者共同努力，辛勤劳动，所有的书稿均经多次审定、修改，使这套丛书达到了较高的质量水平，相信本丛书必能为在书海中遨游的学子指点迷津，助他们踏上成功之路。

本丛书除了适合高校学生学习使用外，对广大的自学者、相关专业工程技术人员亦会有所裨益。

丛书编审委员会邀我为该书作序，谨寄数言，既是对这套丛书的郑重推荐，也是对该套丛书编写者的敬意。

陳大旺

2001 年 5 月

前 言

程序设计语言编译原理是计算机专业的一门核心课程，在计算机本科教学体系中占有十分重要的地位。它通常作为计算机专业，特别是计算机软件方向研究生入学考试科目。该课程具有很强的理论性与实践性，许多读者在学习时感到内容抽象、不易理解、难以掌握。为此，本书以如何正确理解概念和原理、掌握解题技巧、突出重点难点为原则，对那些具有普适性的典型问题，特别是针对研究生入学考试中的重点和疑难问题，详尽地予以阐述和讨论。

本书包括两部分，第一部分为解析篇，以国家教育部制定的计算机专业编译原理课程教学大纲为主线，分为七章：第一章介绍高级语言及编译程序概述，第二章介绍词法分析，第三章介绍语法分析——自上而下分析，第四章介绍语法分析——自下而上分析，第五章介绍语法制导翻译和中间代码产生，第六章介绍运行时存储空间组织，第七章介绍代码优化与目标代码生成。每一章包括复习提要（含考试重点）、典型题解和习题，所有习题均给出了详细的参考答案。复习提要着重理清各章的概念、原理、方法或算法，使读者掌握重点和难点；典型题解通过对精选的各类例题进行深入的分析、讨论与解答，使读者领会各章的精髓，提高解题技巧；习题给读者提供更多的练习，而详尽的习题解答可进一步检验、提高读者解决实际问题的能力。为了强调研究生入学考试的针对性，本书的大部分例题和习题选自近年来全国十几所著名大学和研究所的研究生入学考试试题。本书第二部分为实战篇，包括2份考研模拟试卷及其详细的参考答案，10份考研典型试卷。读者通过这部分的实战练习，定会更全面、透彻地理解、掌握编译原理这门课程，并能轻松地应付各种类型的考试。

本书适合作为研究生入学考试的复习参考书，也宜作为本科生、自学考试考生和其他人员学习编译原理的辅助用书。

本书第一部分的第一、二章由周会平编写，第三、四章由王挺编写，第六、七章由钟求喜编写。刘春林编写第五章和第二部分，并负责最后统稿。编者感谢为本书提供资料的各高校、研究所的研究生招生办和老师们。

由于编者水平有限，书中难免疏漏，恳请读者批评指正。

作者

2001年5月



录

解 析 篇

第一章 高级语言及编译程序概述

1.1 复习提要	(1)
1.1.1 概述	(1)
1.1.2 考试重点	(4)
1.2 典型题解	(4)
1.3 习题	(11)
1.4 习题解答	(13)

第二章 词法分析

2.1 复习提要	(14)
2.1.1 概述	(14)
2.1.2 考试重点	(18)
2.2 典型题解	(18)
2.3 习题	(26)
2.4 习题解答	(29)

第三章 语法分析——自上而下分析

3.1 复习提要	(33)
3.1.1 概述	(33)
3.1.2 考试重点	(35)
3.2 典型题解	(36)
3.3 习题	(56)
3.4 习题解答	(61)

第四章 语法分析——自下而上分析

4.1 复习提要	(71)
4.1.1 概述	(71)
4.1.2 考试重点	(74)
4.2 典型题解	(76)
4.3 习题	(99)
4.4 习题解答	(104)

第五章 语法制导翻译和中间代码产生

5.1 复习提要	(116)
5.1.1 概述	(116)
5.1.2 考试重点	(120)
5.2 典型题解	(120)
5.3 习题	(137)
5.4 习题解答	(142)

第六章 运行时存储空间组织

6.1 复习提要	(155)
6.1.1 概述	(155)
6.1.2 考试重点	(158)
6.2 典型题解	(159)
6.3 习题	(168)
6.4 习题解答	(171)

第七章 代码优化与目标代码生成

7.1 复习提要	(174)
7.1.1 概述	(174)
7.1.2 考试重点	(176)
7.2 典型题解	(178)
7.3 习题	(189)
7.4 习题解答	(193)

实 战 篇

第一部分 硕士研究生入学考试模拟试卷

1. 模拟试卷 (1)	(199)
-------------------	-------

2. 模拟试卷 (1) 参考答案	(201)
3. 模拟试卷 (2)	(206)
4. 模拟试卷 (2) 参考答案	(208)

第二部分 硕士研究生入学考试典型试卷

1. 国防科技大学 1998 年硕士研究生入学考试试卷	(215)
2. 国防科技大学 1999 年硕士研究生入学考试试卷	(216)
3. 国防科技大学 2000 年硕士研究生入学考试试卷	(217)
4. 国防科技大学 2001 年硕士研究生入学考试试卷	(218)
5. 上海交通大学 2000 年硕士研究生入学考试试卷	(220)
6. 北京航空航天大学 2000 年硕士研究生入学考试试 卷	(222)
7. 中国科学院软件研究所 2000 年硕士研究生入学考 试卷	(224)
8. 哈尔滨工业大学 2000 年硕士研究生入学考试试卷	(226)
9. 清华大学 2000 年硕士研究生入学考试试卷	(227)
10. 北京邮电大学 2000 年硕士研究生入学考试试卷	(230)
11. 北京邮电大学 2001 年硕士研究生入学考试试卷	(231)
12. 北京航空航天大学 2001 年硕士研究生入学考试试 卷	(232)
13. 浙江大学 2001 年硕士研究生入学考试试卷	(234)
14. 国防科技大学 2002 年硕士研究生入学考试试卷	(234)
参考文献	(238)

第一章

高级语言及编译程序概述

计算机语言是人机交流的工具，在当今，世界上现有的高级语言已超过千种，其中，在较大的范围内得到使用的语言也有几十种甚至上百种。一般来说，计算机无法直接执行高级语言，因此我们要为计算机构造编译程序。编译程序将高级语言程序翻译成计算机能够识别的机器语言，然后交给计算机去执行。要学习构造高级语言的编译程序，我们必须首先对高级程序语言有充分的了解。

本章的内容主要是有关编译程序和高级语言的一些基本概念。

1.1 复习提要

1.1.1 概述

通常所说的翻译程序是指这样的一个程序，它能够把某一种语言程序（称为源语言程序）转换成另一种语言程序（称为目标语言程序），而后者与前者在逻辑上是等价的。如果源语言是诸如 FORTRAN、Pascal、C、Ada、Smalltalk 或 Java 这样的“高级语言”，而目标语言是诸如汇编语言或机器语言之类的“低级语言”，这样的一个翻译程序就称为编译程序。

编译程序的工作，从输入源程序开始到输出目标程序为止的整个过程，是非常复杂的。通常，编译程序的工作过程可以划分为五个阶段：

第一阶段，词法分析。词法分析的任务是：输入源程序，对构成源程序的字符串进行扫描和分解，识别出一个个的单词（亦称单词符号或简称符号）。

第二阶段，语法分析。语法分析的任务是：在词法分析的基础上，根据语言的语法规则，把单词符号串分解成各类语法单位（语法范畴）。

第三阶段，语义分析与中间代码产生。这一阶段的任务是：对语法分析所识别出的各类语法范畴，分析其含义，并进行初步翻译（产生中间代码）。

第四阶段，优化。优化的任务在于对前段产生的中间代码进行加工变换，以期在最后阶段能产生出更为高效（省时间和空间）的目标代码。

第五阶段，目标代码生成。这一阶段的任务是：把中间代码（或经优化处理之后）变换成特定机器上的低级语言代码。

一个程序语言是一个记号系统。如同自然语言一样，程序语言主要是由语法和语义两方面定义的。有时，语言定义也包含语用信息，语用主要是有关程序设计技术和语言成分的使用方法，它使语言的基本概念与语言的外界（如数学概念或计算机的对象和操作）联系起来。任何语言程序都可看做是一定字符集（称为字母表）上的一个字符串。合乎语法的字符串才算是一个合法的程序。

语言的语法是用来形成一个合法程序的一组规则。这些规则的一部分称为词法规则，另一部分称为语法规则（或产生规则）。词法规则是指单词符号的形成规则。语法规规则则规定了如何从单词符号形成更大的结构（即语法单位），换言之，语法规规则是语法单位的形成规则。一般程序语言的语法单位有：表达式、语句、分程序、函数、过程和程序等等。词法规则和语法规规则定义了程序语言的形成规则，而程序的意义则由语言的语义规则来定义。

程序语言的语法规规则是用文法来描述的，构成文法的规则必须是准确而且易于理解的，并具有相当强的描述能力，足以描述各种不同的结构。现今的程序语言一般用上下文无关文法来描述。

根据其功能和构成规则，高级语言可以分为以下几类：

(1) 强制式语言（Imperative Language），也称过程式语言。如 FORTRAN、C、Pascal，Ada 等等。

(2) 应用式语言（Applicative Language），也称函数式语言，如 LISP、ML 等。

(3) 基于规则的语言（Rule-based Language），也称逻辑程序设计语言，如 Prolog 等。

(4) 面向对象语言（Object-oriented Language），如 Smalltalk 等。

程序的执行过程是对数据的操作，高级语言的数据类型可以分为三类：

(1) 初等数据类型，如数值数据、逻辑数据、字符数据、指针类型等。

(2) 数据结构，如数组、记录、字符串、表格、栈、队列等。

(3) 抽象数据类型，如 Ada 中的程序包、Java 中的类。

程序是由函数或过程构成的，程序的任务是通过函数或过程之间的协作来完成的，函数或过程的调用有以下四种方式：传地址（Call by reference）、得结果（Call by result）、传值（Call by value）和传名（Call by name）。以下面例子为例：

过程原型

SUBROUTINE SWAP (M, N)

.....

END

过程调用

CALL SWAP (I, J);

1) 传地址

今天的高级语言基本上都实现了传地址的参数传递方式。所谓传地址是指把实在参数的地址传递给相应的形式参数。在过程段中每个形式参数都有一对应的单元，称为形式单元。形式单元将用来存放相应的实在参数的地址。当调用一个过程时，调用段必须预先把实在参数的地址传递到一个被调用段可以拿得到的地方。当程序控制转入被调用段之后，被调用段首先把实参地址抄进自己相应的形式单元中，过程体对形式参数的任何引用或赋值都被处理成对形式单元的间接访问。当调用段工作完毕返回时，形式单元（它们都是指示器）所指的实在参数单元就持有所指望的值。对于传地址，上面函数调用的参数传递过程如图 1.1 所示。

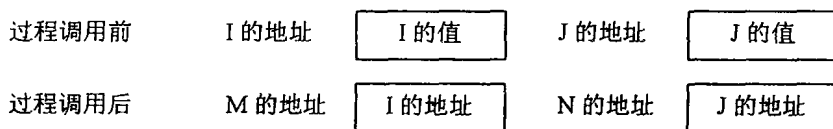


图 1.1 传地址的参数调用

图中参数调用后，形式参数 M 的值是 I 的地址，N 的值是 J 的地址，这样通过 M 的值可以间接访问到 I 的值，通过 N 的值可以间接访问到 J 的值。

2) 得结果

和“传地址”相似（但不等价）的另一种参数传递方法是所谓“得结果”。这种方法的实质是，每个形式参数对应有两个单元，第 1 个单元存放实参的地址，第 2 个单元存放实参的值。在过程体中对形参的任何引用或赋值都被看成是对它的第 2 个单元的直接访问。但在过程工作完毕返回前必须把第 2 个单元的内容存放到第 1 个单元所指的那个实参单元之中。对于得结果，上面函数调用的参数传递过程如图 1.2 所示。

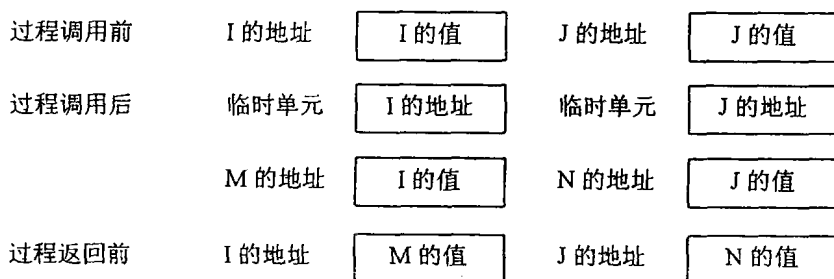


图 1.2 得结果的参数调用

图中参数调用后，形式参数 M 的值是 I 的值，N 的值是 J 的值，但在函数中无法访问到 I 和 J 所在的内存单元，只有在函数返回前将 M 和 N 的值分别抄送到 I 和 J 所在的内存单元中。

3) 传值

“传值”是一种简单的参数传递方法，也是今天高级语言中使用最多的参数传递方法之一，调用段把实在参数的值计算出来并存放在一个被调用段可以拿得到的地方。被

调用段开始工作时，首先把这些值抄入形式单元中，然后就好像使用局部名一样使用这些形式单元。如果实在参数不为指示器，那么，在被调用段中无法改变实参的值。对于传值，上面函数调用的参数传递过程如图 1.3 所示。

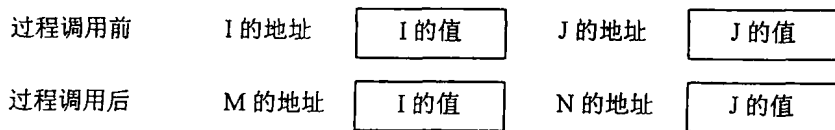


图 1.3 传值的参数调用

图中参数调用后，形式参数 M 的值是 I 的值，N 的值是 J 的值，但在函数中无法改变 I 和 J 的值。

4) 传名

传名是 ALGOL60 所定义的一种特殊的形—实参数结合方式。过程调用的作用相当于把被调用段的过程体抄到调用出现的地方，但把其中任一出现的形式参数都替换成相应的实在参数（文字替换）。它与采用“传地址”或“传值”的方式所产生的结果均不相同。对于传名上面函数调用的参数传递过程如图 1.4 所示。

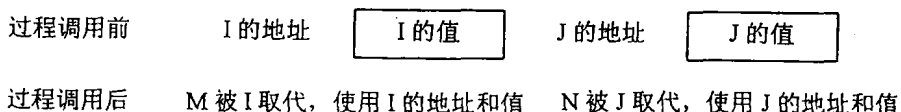


图 1.4 传名的参数调用

图中参数调用后，形式参数 M 被 I 所取代，N 被 J 所取代，函数中对 M 和 N 的引用都是直接对 I 和 J 的引用。

1.1.2 考试重点

1. 编译程序的结构。
2. 高级语言及其语法描述的基本概念。
3. 过程和函数调用时的参数传递方法。

1.2 典型题解

例题 1.1 是非题

1. 因名字都是用标识符表示的，故名字与标识符没有区别。(X)
2. 正规文法产生的语言都可以用上下文无关文法来描述。(X)
3. 上下文无关文法比正规文法具有更强的描述能力。(X)

分析

1. 标识符是高级语言中定义的字符串，一般是以英文字母（包括大小写字母）开头的，由数字、字母和一些特殊字符（如下划线等）组成的一定长度（不同的高级语言对标识符的长度的规定不同）的字符串，它只是一个标志，没有其它含义。名字是用标识符表示的，但名字不仅仅是一个字符串，它还具有属性和值，就像一个人的名字不仅是一个符号，对于认识这个人的人来说，它还代表着这个人，因此本题错。

2. 乔姆斯基把文法分成4种类型，从4种文法的形式化描述来看，它们的差别主要在对规则形式的约束上。0型文法的规则形式是：每个产生式 $\alpha \rightarrow \beta$ 都满足， $\alpha \in (V_N \cup V_T)^*$ ，且至少含有一个非终结符， $\beta \in (V_N \cup V_T)^*$ ，0型文法也叫短语文法。1型文法的规则形式是：每个产生式 $\alpha \rightarrow \beta$ 均满足 $|\alpha| \leq |\beta|$ ，仅仅 $S \rightarrow \epsilon$ 例外，且 S 不得出现在任何产生式的右部，1型文法也称上下文有关文法，从两种文法的规则形式来看，1型文法对规则形式的约束比0型文法强，1型文法能描述的语言0型文法也能描述，而0型文法能描述的语言1型文法不一定能够描述，1型文法是0型文法的特例。2型文法的规则形式是 $A \rightarrow \beta$ ，其中 $A \in V_N$ ， $\beta \in (V_N \cup V_T)^*$ ，2型文法也称上下文无关文法，分析2型文法的规则形式不难发现，2型文法是1型文法的特例，也是0型文法的特例。3型文法的规则形式是 $A \rightarrow \alpha B$ 或 $A \rightarrow \alpha$ ，其中 $\alpha \in V_T^*$ ， $A, B \in V_N$ ，3型文法也称正规文法，可以看出3型文法是前面3种文法的特例。从上面的分析可知，正规文法是上下文无关文法的特例，可以用正规文法描述的语言，其正规文法描述的形式也是上下文无关文法的描述形式，即可以用上下文无关文法描述，因此本题对。

3. 上下文无关文法是2型文法，正规文法是3型文法，从上题的分析可以看出，3型文法是2型文法的特例，3型文法可以描述的语言都可以用2型文法来描述，而2型文法可以描述的语言则不一定能用3型文法来描述。即2型文法比3型文法的描述能力强，因此本题对。

例题 1.2 填空题

1. 程序语言是由 (语法) 和 (语义) 两方面定义的。
2. 常用的参数传递方式有 (传地址)、(传值) 和 (传名)。
3. 文法 G 所产生的句子的全体是一个 (语言)，将它记为 ($L(G)$)。
4. 一个上下文无关文法包含四个组成部分是 (一个开始符号、一组非终结符、一组终结符、一组产生式)。

解答

1. 如同自然语言一样，程序语言主要由 (语法) 和 (语义) 两个方面定义，有时，语言定义也包含语用信息，语用主要是有关程序设计技术和语言成分的使用方法。

2. 参数传递主要有三种不同的途径，(传地址 (Call by reference))、(传值 (Call by value)) 和 (传名 (Call by name))，传名也常称为换名。

3. 假定 G 是一个文法， S 是它的开始符号。如果 $S \xRightarrow{*} \alpha$ ，则 α 称是一个句型。仅含终结符号的句型是一个句子。文法 G 所产生的句子的全体是一个 (语言)，将它记为

(L (G))。

4. 一个上下文无关文法 G 包括四个组成部分：(一组终结符号)，(一组非终结符号)，(一个开始符号)，(一组产生式)。

说明

高级语言参数传递的方式通常有 4 种，传地址、得结果、传值和传名。早期的 FORTRAN 语言通常采用传地址或得结果的方式来实现参数的传递，这两种方法的不同给 FORTRAN 的标准化增加了难度，相比来说，传地址是一种常用的参数传递方式，而得结果则不是。

例题 1.3 简答题

1. 什么叫自展？什么叫交叉编译？
 2. 什么是标识符，什么是名字，他们的区别是什么？
- (北京航空航天大学 1998 年硕士生入学考试试题)

解答

1. 采用“自编译方式”产生编译程序的方法叫自展。即先对语言的核心部分构造一个小小的编译程序(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，最后形成整个编译程序。一般称运行编译程序的计算机为宿主机，运行编译程序所产生目标代码的计算机称目标机。如果编译程序编译时产生不同于其宿主机的机器代码，则称它为交叉编译，即在 A 机上的编译程序对高级语言编译后所产生的目标代码是 B 机的机器代码，而且 A 机和 B 机使用不同的机器代码。

2. 标识符是由字母或数字以及某些特殊符号(因不同的高级语言而不同)组成的，但是以字母开头的一个字符串。当给某标识符以确切的涵义时，这个标识符就叫做名字。程序语言中各种名字都是用标识符表示的。名字和标识符具有相同的形式，名字使用标识符来描述，但标识符是没有意义的字符序列，而名字却有确切的意义和属性(即类型和作用域)。

例题 1.4

画出编译程序的总体结构图，简述个部分的主要功能。

(哈尔滨工业大学 2000 年硕士生入学考试试题)

解答

编译程序的总框图见图 1.5。其中词法分析器，又称扫描器，它接受输入的源程序，对源程序进行词法分析，识别出一个个的单词符号，其输出结果是单词符号。

语法分析器，对单词符号串进行语法分析(根据语法规则进行推导或归约)，识别出程序中的各类语法单位，最终判断输入串是否构成语法上正确的“程序”。