

2004

考研

计算机专业
研究生
入学考试

全真题解

——编译原理分册

◎ 前沿考试研究室 编著 ◎

- 重点内容归纳
- 考试要点提炼
- 复习方法总结
- 例题详尽解析
- 真题实战演练



人民邮电出版社
POSTS & TELECOM PRESS



考研

计算机专业
研究生
入学考试

全真题解

——编译原理分册

◎ 前沿考试研究室 编著 ◎

人民邮电出版社

图书在版编目(CIP)数据

计算机专业研究生入学考试全真题解. 编译原理分册 / 前沿考试研究室编著.

—北京: 人民邮电出版社, 2003.7

ISBN 7-115-11351-3

I. 计... II. 前... III. ①电子计算机—研究生—入学考试—解题②编译程序—程序设计—研究生—入学考试—解题 IV. TP3-44

中国版本图书馆 CIP 数据核字 (2003) 第 046435 号

计算机专业研究生入学考试全真题解 ——编译原理分册

◆ 编 著 前沿考试研究室

责任编辑 王文娟

◆ 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 010-67132692

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: 11.75

字数: 376 千字

2003 年 7 月第 4 版

印数: 22 001-28 000 册

2003 年 7 月北京第 7 次印刷

ISBN7-115-11351-3/TP · 3504

定价: 18.00 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

《计算机专业研究生入学考试全真题解》共分 5 册，分别是：

- 数据结构与程序设计分册。
- 离散数学分册。
- 操作系统分册。
- 编译原理分册。
- 硬件分册（包括数字逻辑、计算机组成原理、计算机系统结构）。

这 5 册内容基本覆盖了计算机专业研究生入学考试涉及的 7 大部分。书中对知识点和考点之间的关系进行了深入挖掘，对典型例题进行了深入剖析，以求达到举一反三的目的。

本书为编译原理分册，包括如下几部分内容：

- 试题分析和解题方法。这部分体现了全书的指导思想。
- 主要内容概述。
- 常考知识点及复习方法建议。
- 真题详细解析。这部分是本书重点，汇集了近年来全国 20 余所著名院校计算机专业研究生入学考试的试题，对其进行了细致、深入的分析、解答和扩展。
- 习题精选及部分答案。
- 部分学校最新考研试题汇编。

本书适合报考计算机专业研究生的考生有针对性地进行专业课的复习，也适合希望深入学习计算机专业知识的高校学生作为辅导书参考。同时，本书还可以作为习题集使用。

计算机专业研究生入学考试全真题解

编写委员会

主编：

温 谦

副主编：

谢廷宝 李国强 马骁骐

编写人员（排名不分先后）：

谢廷宝	李国强	翁 鸣	何 军	赵 宁	肖 宇
杨 勇	李 勇	温 谦	冯 军	马 捷	贺 劲
刘 洪	李 凡	王海洋	邓 刚	张桢睿	张金波
梁 泉	马骁骐	唐志虎	唐力军	解永良	吴少刚
钟 喻	张 健	贾 培	骆 文	罗 平	胡明昌

修订说明

从 2000 年《计算机专业研究生入学考试全真题解》丛书出版开始,今年已经出到第 4 版了。在这 4 年中,这套书经过了 3 次修订,其内容不断丰富,组织更加合理,受到广大读者的欢迎,我们感到十分欣慰。许多读者给我们写信,提出了宝贵的意见和建议,与我们探讨计算机专业考研方面的经验,我们也将部分反馈意见添加进了每次的修订,在此向他们表示衷心的感谢!

经过 4 年的努力,本书内容趋于稳定,题目讲解部分的改动不大。我们的目标就是希望通过这套书,使读者可以在最短的时间里付出最小的代价,获取最大的信息量,从而取得最好的复习效果。

今年我们主要在信息量方面做了一些工作,添加了少量新题;优化了一部分讲解的内容;收集了一些考生相对集中的院校最近一两年的考研试题,放在了附录中,作为仿真试题。同时,考虑到我们有很大一部分读者是学生,我们在增加内容的同时,通过紧缩版式,缩小字号,下调了这套丛书的价格,以减轻读者的负担。

欢迎广大读者继续和我们交流,我们的网站和信箱依然有效。

网站: <http://www.artech.com.cn/kaoyan>

作者电子信箱: books@artech.com.cn

责任编辑电子信箱: wangwenjuan@ptpress.com.cn

再次感谢过去 4 年中广大读者给予的热情支持,你们的热情,是我们工作的无尽动力。希望这次仍能得到你们一如既往的支持!

编者

2003.5

序

一年一度的研究生入学考试又结束了，这也意味着新一轮复习的开始。一份努力，一份收获，付出之后总有回报。在备考过程中，每一位参加者都在进行着智力、勇气和毅力的较量。我们作为一群参加过研究生入学考试，并如愿考取的学生，以“过来人”的经验编写了本套丛书。意在将我们在考研复习过程中的收获进行总结，传递给后来的考生，帮助你们少走弯路，在复习时达到融会贯通、举一反三的境界。衷心希望每一位读者都能成为竞赛的胜利者！

本套丛书的由来及改进

这套《计算机专业研究生入学考试全真题解》丛书最初诞生于2000年，当时这套丛书共有3册。2001年，我们给原书中没有提供答案的习题加上了答案，同时添加了一些新的题目。2002年对这套丛书进行修改时，我们认真总结了过去两年中读者的意见和建议，进行了较大的改进，主要体现在以下7个方面。

1. 对于每一章的结构，2001年版中按知识点罗列了各类考题，没有充分考虑题目与考点以及题目之间的内在联系，而这次我们对所有题目进行了深入的挖掘，寻找其内在联系，经过归纳、融合，使之成为有机的整体。这样读者在复习的时候，就可以找到循序渐进、深入浅出的感觉。

2. 对于每一道例题的讲解，我们都进行了精心设计，每道例题基本包括“分析”、“解答”、“扩展”3个部分，使读者能够明确整个题目的分析过程和需要注意的地方，而不仅仅是知道答案而已。这些都是我们经验的总结，相信能够帮助读者少走弯路，提高复习效率。

3. 针对每一科目，增加了第0章，提纲挈领地分析了该科目试题的特点并提供了解题方法指导。

4. 根据读者建议，增加了对学校和年份的试题索引，读者可以方便地找出某所学校、某一年的试题。有的学校的题目不足一张完整的试卷，是因为我们觉得有些题目过于雷同，就没有收入，凡是典型的题目我们都收录了。

5. 增加了近两年的最新题目及其答案。

6. 所有科目单独成册（硬件理论仍保持1册），由原来的一套3本扩充到5本，细分读者群的做法更体现了我们以读者为本的宗旨，也更方便了读者的选购。

7. 这次版式安排接受了读者的建议，尽量把内容安排紧凑，并删除了2001年版中关于招生信息的附录，将这部分内容放到我们的网站上（<http://www.artech.com.cn/kaoyan.htm>），尽量降低图书成本。

今年是第4次对本套丛书进行修订，题目讲解部分的改动不大，主要在信息量方面做了一些工作，添加了少量新题；优化了一部分讲解的内容；对题目容量进行了一定的扩充，收集了一些考生相对集中的院校最近一两年的考研试题，放在了附录中。

如何进行考研复习

谈到复习，根据我们的经验，复习必须注重“全面”与“重点”相结合。因为一门课程的内容非常多，考试只考其中的一小部分。其实所谓重点就是出题概率比较高，并且所占的分值也比较高的内容。对不同的内容，考试中考到的可能性（概率）也不同，有的内容考到的概率很高，有的则很低，如图1所示。

如果仅仅针对一些“重点”来进行复习，肯定得不了高分。图2表示了只抓重点的复习方式。图中矩形覆盖的部分表示复习到的内容，从某种意义上说，这种方式的效率是比较高的。对于本科学习期间的期末考试，为了通过，这种方式是有效的，抓一抓重点（加上老师可能会告诉你一些“真正”的重点），也许就可以轻松过关了。然而“考研”是选拔性考试，仅仅通过最低分数线没有任何意义，还要和所有考生一起比一比高低，这时就要看每个考生的真本事了。想要得高分，就必须真正对考试的内容有着深刻的理解，仅仅靠压题、抓重点是不行的。因此，从全面的角度出发，我们尽量多地收集了各知名院校近年来的“考研”试题。如果书里的所有题目你都会做，那么你的专业课成绩应该不会低。这不是因为我们帮你压准了哪道具体的题目，而是通过解这些题目，你已经更深刻地理解了这门学科，就好像现在让你去参加小学生

的考试，即便你完全不知道要考什么内容（即你完全不知道范围、重点在哪里），你会害怕吗？

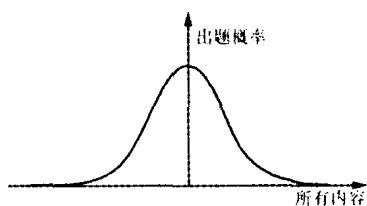


图1 对于不同的内容，出题的可能性不同

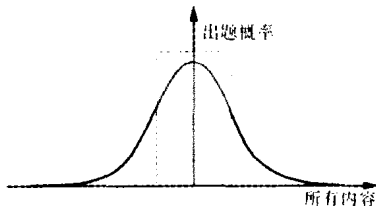


图2 只抓重点的复习方式

显然，在全面的基础上抓重点，是得高分的重要手段，也就是说，在出题概率大的部分要多花力气。客观地说，每个院校都有自己的考查重点，因此作为一个聪明的考生，必须了解所报考院校的试题风格，而这种风格具有很强的延续性。因此这套书中的所有试题都注明了院校名称。如果只强调全面，由于复习时间和人的精力是有限的，因此可能会把很多重点漏掉，那就很可惜了，如图3所示。

比较科学的复习方法，应该如图4所示，考生在各个部分花费的精力应该与出题概率尽量一致。考试是通过“以偏概全”方式评价考生：通过短短的几个小时、几道题目，就要判断一名考生的水平，确实具有一定的偶然性。但是大家千万不要有侥幸心理，侥幸心理只会害了自己。只有扎扎实实地准备才能获得令人满意的结果。

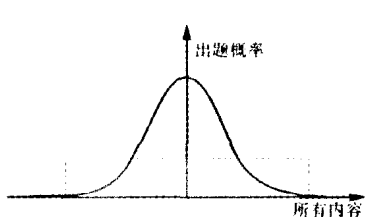


图3 片面强调全面的复习方式

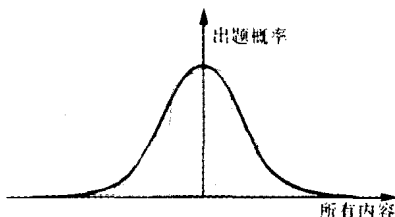


图4 比较科学的复习方式

如何使用本套丛书

这套书里含有大量的考研真题，对于考试复习是非常珍贵的，建议大家按照下面5个步骤，利用本书进行复习：

1. 第一遍，进行广泛复习，把本书中所有例题都做一遍，在做的同时要进行分类，并做好标记。建议分为3类：①完全掌握，没有看答案就做得非常正确。②会做，但没有做对，就是说对于题目体现的知识点基本上掌握了，但是还存在缺陷，所以没有做对。③看着题目，无处下手，就是离准确掌握同类题目的差距还比较大。我们在每道例题的后面附了3个小方格，以便读者做标记。

2. 第二遍，把第一遍中标记为第②类的题目，一个不少地认真再做一遍，把所有漏洞都补上。同时把相关类型的习题做一遍，作为补充。

3. 在上两遍的基础上，再来攻克第③类题目，有上两遍的基础，这时再来做这些难题，相信感觉一定会大不相同。当然建议大家针对自己所报考学校的要求，来决定这些题目需要掌握的程度。

4. 在以上3步中，都要养成记笔记的习惯，随时记录题目中存在的问题和做题过程中的心得体会。这样，在三遍完成之后，可以参照笔记，对存在问题的题目做最后的拾遗补漏。

5. 报考本书中收录题目较多的学校的读者，建议在考前，再根据试题索引，把该校的题目，一个不漏地过一遍。如果报考其他院校，也建议最后把往年试卷仔细过一遍。

几点建议

最后提醒读者注意几点：

1. 一定要重视基础题，尤其是那些自己会做，但是做错了的题目。紧盯住这些题目，是提高复习效率的“捷径”。不要急于做难题，基础扎实了，水平提高了，难题自然就不难了。

2. 一个题目, 只有自己做出来, 自己想出办法, 才是真正掌握了, 如果看了答案才做出来, 还不算真正掌握。此外, 如果没有完全做对, 即使只差一点点, 也要告诉自己, 这个题目还没有掌握。真题是十分有限的, 因此要珍惜。看了答案才做对的题, 就要放一段时间, 等印象不深了, 再做一遍, 完全做对才算通过。

3. 考试, “无他, 唯手熟尔!” 尽管人的智力存在差异, 但只要肯下功夫, 人人都可以考高分。但是方法是很重要的, 真正聪明的人是找到适合自己方法的人。找到合适的方法, 等于成功了一半。

4. 各个学校的考试范围不尽相同, 要尽早搞清楚自己所报考学校大体的考试范围, 复习的时候, 尽量复习得比这个范围略大一些, 略难一些, 以增加保险系数。

读者交流

本书是这套丛书的编译原理分册。

我们的网站和信箱如下。

网站: <http://www.artech.com.cn/kaoyan.htm>

作者电子信箱: books@artech.com.cn

责任编辑电子信箱: wangwenjuan@ptpress.com.cn

鸣谢

首先感谢人民邮电出版社对这套书的重视, 该社连续 4 年的不懈努力, 使这套书逐步完善。

感谢考研加油站 (<http://www.kaoyan.com>) 的站长林毅强以及其“考研论坛”计算机版的众多热心网友, 本版的编写得到了众多网友的大力支持。

感谢过去两年广大读者给予的热情支持, 你们的热情是我们工作的无尽动力。希望这次仍能得到你们一如既往的支持!

编者
2003.5

目 录

第0章 编译原理试题分析和解题方法	1
0.1 编译原理不好学	1
0.2 编译原理也不难学	2
0.3 精通编译原理还是不容易	3
0.4 复习要三步走	3
第1章 编译程序概述	4
1.1 主要内容概述	4
1.1.1 程序的翻译	4
1.1.2 编译的过程	5
1.1.3 编译程序的结构	5
1.2 常考知识点及复习方法建议	6
1.3 真题详细解析	6
第2章 高级语言及其语法描述	8
2.1 主要内容概述	8
2.1.1 高级语言概述	8
2.1.2 文法及语言	9
2.2 常考知识点及复习方法建议	13
2.3 真题详细解析	13
2.4 习题精选与部分解答	24
2.4.1 习题精选	24
2.4.2 部分习题参考答案与提示	25
第3章 词法分析	26
3.1 主要内容概述	26
3.2 常考知识点及复习方法建议	29
3.3 真题详细解析	30
3.4 习题精选及部分解答	42
3.4.1 习题精选	42
3.4.2 部分习题参考答案与提示	45
第4章 语法分析——自上而下分析	47
4.1 主要内容概述	47
4.2 常考知识点及复习方法建议	49
4.3 真题详细解析	49
4.4 习题精选与部分解答	56

4.4.1 习题精选	56
4.4.2 部分习题参考答案与提示	60
第5章 语法分析——自下而上分析	61
5.1 主要内容概述	61
5.1.1 自下而上分析概述	61
5.1.2 算符优先文法	62
5.1.3 LR 分析法	64
5.2 常考知识点及复习方法建议	67
5.3 真题详细解析	67
5.4 习题精选与部分解答	97
5.4.1 习题精选	97
5.4.2 部分习题参考答案与提示	102
第6章 中间代码生成	105
6.1 主要内容概述	105
6.2 常考知识点及复习方法建议	107
6.3 真题详细解析	107
6.4 习题精选与部分解答	121
6.4.1 习题精选	121
6.4.2 部分习题参考答案与提示	125
第7章 存储空间组织	127
7.1 主要内容概述	127
7.2 常考知识点及复习方法建议	130
7.3 真题详细解析	130
7.4 习题精选与部分解答	142
7.4.1 习题精选	142
7.4.2 部分习题参考答案与提示	144
第8章 代码优化和生成	145
8.1 主要内容概述	145
8.2 常考知识点及复习方法建议	147
8.3 真题详细解析	147
8.4 习题精选与部分解答	158
8.4.1 习题精选	158
8.4.2 部分习题参考答案与提示	160
附录 部分学校最新考研试题汇编	162
历年试题学校与年份分类索引	173

第0章 编译原理试题分析和解题方法

本书的读者很多都是计算机专业高年级（或者同等水平）的学生，对于编程已经不再陌生，甚至我们很多人都要把它作为一门安身立命的技术。这里先给大家出一个很小的题目，它不是考研试题，而是一个公司在一次招聘时出的考试题目，很有意思，题目是这样的：

以 DOS 为操作系统，使用 C 语言，编写一个函数，功能是取得当时的 BP 寄存器的值，只能使用标准的 C 语句（也就是不能使用扩充的宏等其他手段）。

如果大家在面试的时候遇到这道题目，头脑中能够马上反应出这到底是考的什么吗？这里实际上考的是编译程序的存储组织。

下面我们简单分析一下该题。我们知道 C 语言是栈式存储分配，局部变量的分配和参数传递都是通过栈来完成的，在调用函数的时候，首先把被调函数的参数从右向左依次压入栈，再把返回地址压入栈，然后转入被调函数内部，进入被调函数以后，首先执行固定的两条指令：

```
PUSH BP
```

```
MOVE BP, SP
```

这样，通过 BP 寄存器，加上或者减去一定的偏移量，就可以对传递来的参数和函数内新分配的局部变量进行寻址了。到这里，我们就找到答案了，这个题目所要求的函数不需要做任何实际的工作，只需要声明一个整数类型的变量，那么显然这个变量就在栈顶，而它的下面的两个字节保存的就是在调用这个函数之前 BP 寄存器的值（也就是上面说的指令 PUSH BP 所保存的内容），将这个值返回就可以了，也就是把这个整型变量的地址加 2 后的内容返回即可。

这里举这个小例子就是为了说明编译原理对于编程的重要性，它可以帮助我们更深刻地理解程序运行时内部的机制。

回顾一下我们学习编程的历程：开始是学习一门高级语言，比如 C 或者 Pascal，这样就掌握了编写一些简单程序的方法。然后又学习了数据结构，这是一门非常重要的课程，学习了数据结构，我们的头脑中建立了“算法”的概念，对编程有了更深的理解，遇到问题的时候，我们知道应该寻找相应的数据结构模型，设计适当的算法来解决问题。此外，我们通常还要学习汇编语言，尽管大多数人并不会在实际中用汇编语言来进行开发，但实际上，这门课程是真正深入了解计算机内部的第一门课程。通过学习汇编语言，知道了每一条指令是如何在计算机内部运行的。而此时，我们已经可以编写很多程序了，但是还不知道高级语言最终是如何成为汇编语言的，甚至如何成为机器语言的，编译原理课程就可以帮助我们了解高级语言变成汇编语言的过程。更进一步，要想知道高级语言是如何变为机器语言，以及如何在 CPU 内部运行的，就要通过学习计算机组成原理的课程了。学习后，我们就可以清楚地认识到编译原理在计算机学科中所处的重要位置了。同时，我们也可以感到，它的复杂性和学习的难度都是很大的。

0.1 编译原理不好学

编译原理课程的一个明显特点是它的理论性很强，在学习的时候需要很强的逻辑思维能力。这门课程初看起来内容很多，而且又非常抽象，如果和“数据结构”比起来，就会发现，编译程序所涉及的算法要复杂得多，而且要深入地理解这些算法也很困难。比如在学习数据结构的时候，会接触很多算法，经常会跟踪算法的执行过程，比如遍历二叉树等。而跟踪编译程序中的算法则要麻烦得多，从而要想深刻理解算法的思想就很困难。编译程序的构造方法确实非常精妙，就像一部走时精确的时钟，很多齿轮、部件协调地运转，以驱动指针准确地旋转；编译程序也是如此，一边扫描源程序，一边经过各个部件的运算，准确地输出为目标语言。

0.2 编译原理也不难学

上面说了编译原理不好学的原因,但如果弄清了这门课程的特点和规律,以及它作为一门考试科目的特点,就会发现攻克编译原理其实也不难。

1. 编译原理课程各个部分之间的独立性很强,实际上就包括了词法分析、语法分析、存储的组织与分配、中间语言、语法制导翻译、代码生成与优化这几大部分。而且前两个部分是其中的重点,也是难点,需要掌握比较复杂的算法逻辑,而其他部分相对来说知识性更强一些。另外,编译原理每个部分之间的方法也互相独立,在复习的时候,便于逐个击破。

2. 实际上编译原理考试考查的内容相对来说是很稳定的,而且绝大多数题目的解法都非常机械,如果熟练掌握,考试的时候沉着冷静,不要粗心,做对是非常容易的。

具体举例说明一下,比如词法分析这一部分,如果仔细研究之后,就会发现,最重要的就是要掌握如图 0.1 所示的 6 个转换关系。对于给出的一个正规式,要依次求出相应的 NFA、DFA、最小化的 DFA 等,这 6 种转换关系都有确定的方法,只要熟练掌握了相应的方法,无论怎么出题都应该能够套用。

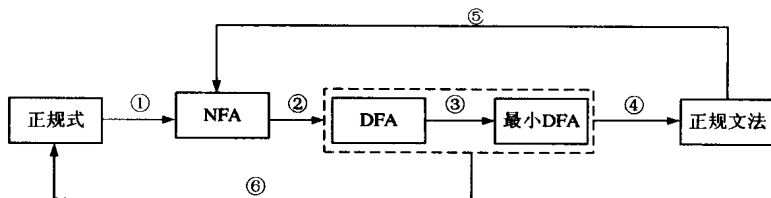


图 0.1 考点及其关系图

再比如语法分析部分,这部分是整个编译原理考试中比重最大、难度也最大的一个部分。但仔细分析之后,我们会发现,语法分析只有两大类:自顶向下和自底向上,每类中又有几种不同的方法,每种方法都对应于一类文法。因此在复习的时候,首先弄清楚每种方法适用的具体条件,具体解法套用相应的步骤就可以了。

下面以自底向上得 LR 分析法为例加以说明。LR 分析法通常包括了 4 种具体的分析方法:LR(0)、SLR(1)、LR(1)、LALR(1)分析法,这 4 种 LR 分析法的原理如图 0.2 所示。

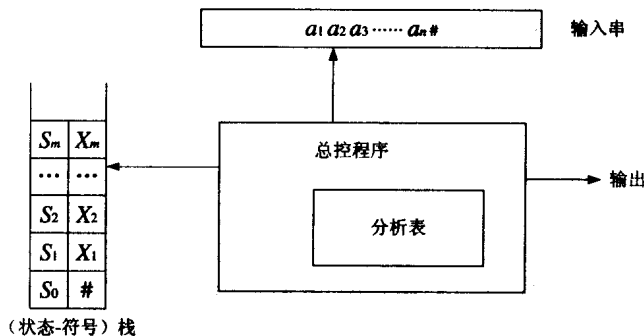


图 0.2 分析法示意图

这 4 种 LR 分析法的总控程序是相同的,只是分析表不同。因此复习的关键是掌握构造分析表的方法,以及使用分析表进行分析的方法。LR(0)是 LR 分析法的基础,首先要深刻地理解 LR(0)分析法的思想,熟练掌握 LR(0)文法的判断方法和 LR(0)分析表的构造方法,这里再次强调,其方法是非常机械的,只要多加练习,就可以掌握得非常牢固,只有这样,在考场上才能游刃有余。

在掌握了 LR(0)分析法后,就可以看出 LR(0)分析法要求构造出的识别可归前缀的有穷自动机的各个状态不能有冲突项目,而 SLR(1)分析可以部分地解决这个问题,其方法是向前多看一个输入符号,这样又需要掌握 SLR(1)分析法,同样要注意其适用的文法限制和分析表的构造方法。要注意 SLR(1)只是部分解决了 LR(0)的缺陷,当我们需要进一步加强分析法的能力时,就到达了 LR(1)分析法和 LALR(1)分析法,大体的复习思路都是相同的。

这 4 种分析法都掌握后,最好仔细总结一下其内在的关系,使用的文法限制等。总之,这部分要掌握

每种分析法适用的文法，并能构造出相应分析表，以及应该做到当给出一个该文法的句子，能够跟踪使用分析表进行分析。

其他各个章节也都有类似的特点，大家在复习的时候应该认真地进行总结，提纲挈领地抓出线索，熟练掌握具体的方法。一定要多做习题，千万不能觉得自己知道方法了，就不用亲手做了。尤其对于编译这样的学科，题目的规模很大，步骤繁多，而且前面的步骤一旦出错，后面都错。如果不熟练掌握的话，考试的时候必定慌乱，所以切忌眼高手低，一定要多加练习，才能到考场上沉着应对。

客观来说，如果这些大的方法（如 DFA 相关的转换、LR 分析法）都掌握了，即使对编译原理本身的理解并不十分深刻，也可以应对考试。因为在考试中绝大部分都是常规题目，抓住这些常规题是效率最高的复习方法。

0.3 精通编译原理还是不容易

看了上面的方法，似乎给大家一个印象，记住这些条条框框就可以了，不用认真掌握编译原理的内部机制了。其实并非如此，有人说，编译原理以其独特的风格和韵味，带给大家无穷的乐趣。确实如此，如果仔细研究，深入分析，可以从中发现极具艺术性的美感。但是要想真正深入地精通它，绝不是仅仅掌握一些固定甚至是支离破碎的算法就可以达到的。切实掌握这门课程离不开实践，实际上亲自动手完成一个小型的编译器，对于理解这门学科可能胜于做一千道习题。因此如果用一句话来概括编译原理的学习过程，可以说是入门难，提高易，精通难。

我们之所以在上面反复强调题目解法固定、机械的特点，更多的是从本书的特点出发，从考试的角度分析的。如果在复习前比较认真地上过课，相对来说，编译原理是比较容易拿分的，因为它的考点相对集中，如果和“数据结构”比较一下就会发现，数据结构的考点非常散，试题的花样远远多于编译原理的考试，即使是最基本的算法，也可以考得非常难。

实际上，学习了编译原理，对提高算法的分析和设计能力是非常有帮助的。比如在学习编译原理之前，数据结构中遇到的编写表达式求值的算法，都是比较难的题目，但是在学习了编译原理之后，就变得非常简单了。另外，学习编译原理，对递归的分析算法的掌握也有极大的帮助。

0.4 复习要三步走

上面讲了学习编译原理的特点是入门难、提高易、精通难，相应地，我们的复习也有三步。

● 第一步是尽可能地掌握编译原理的思想，不要一开始就紧紧盯在一些具体的算法上，要站得高一点，尽可能理解算法的内部机制，而不是去机械地背下那些固定的算法，比如各种语法分析的方法。应该尽力理解为什么要这样做，逐渐在头脑中建立起编译器的整体概念，而不是零零散散的一些算法。

● 第二步就是要扎扎实实地把重要算法记牢，同时配合大量的习题进行练习，达到拿到题目就可以动手做的地步。而不是拿到一个题目后，还要犹豫半天，再查书，看看笔记，这样是不行的，说明练得还不够，要继续练。等题目做多了，脑子里自然就会形成印象，一看题目就知道是怎么回事。进行这一步同时也会加深对编译原理进一步的理解。

● 对前面的复习做认真的总结，关键是找差异，各个方面的差异都要关注，如果有调件，亲自动手编一些简单的编译程序或者编译程序的一部分，都是很有益的，这样能更灵活地掌握编译程序构造的精髓。

第 1 章 编译程序概述

本章中考试涉及的主要知识点包括编译原理中的基本概念、编译程序工作的基本过程及各阶段的基本任务、编译程序总的框架等。

1.1 主要内容概述

1.1.1 程序的翻译

1. 程序设计语言与程序的翻译

程序设计语言有多种：不需翻译就可直接执行的机器语言，机器指令助记符形式的汇编语言，以及接近自然语言的高级程序设计语言（如 Pascal、C、Java 等）。

除机器语言程序外，用其他语言书写的程序都必须经过翻译才能被计算机识别，这一过程由翻译程序来完成。所谓翻译程序是这样一种程序，它能把甲语言编写的程序翻译成与之等价的乙语言程序。甲语言称为该翻译程序的源语言，乙语言称为该翻译程序目标语言。用源语言书写的程序称为源语言，与源语言等价的目标语言程序称为目标程序。

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。

2. 编译方式和编译程序

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机器语言程序，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程序或机器语言程序，则这种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，则这种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程序或简称为代码程序，有时又称为目标代码或结果代码。

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。

3. 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言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，并不生成目标代码，而是直接执行源程序本身。这是编译方式与解释方式的根本区别。

语言的语法规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。

4. 自展

自展是先对语言的核心部分构造一个小的编译程序，再以它为工具构造一个能够编译更多语言成分的较大的编译程序。如此扩展下去，直至最后形成人们所期望的整个编译程序。这是在编译程序构造中经常使用的方法。

5. 交叉编译

交叉编译是在计算机系统 A 上编译的能够在与系统 A 不同的计算机系统 B 上运行的程序，在 B 系统中编译 A 系统中的程序。

1.1.2 编译的过程

编译程序的功能是把一个由高级语言（如 C 语言）编写的源程序进行翻译，转化成低级语言（如汇编语言）或者机器语言的目标程序，使机器可以链接执行这个程序。不同种类的语言需要不同的编译器，所以产生了 C 的编译器、Pascal 的编译器等。

一般把源程序翻译成目标程序，需要以下几个步骤。

（1）词法分析

首先对源程序进行扫描，把一段字符或者字符串的序列分解为一个一个单词，忽略掉不必要的空格，转化为内部表示。如 C 语言程序 `main{ float i; i=10;}`，首先分解为 `main、{、float、i、;、=、10、}` 这些单词序列。

（2）语法分析

在前面的基础上将单词序列分解为可以用语法单位衡量的语法短语，检查语法上是否正确，能否表示成语法树。如根据 C 的赋值语句，可以判断“`i=10;`”的语法成分。

（3）语义分析

判断程序中是否有语义错误。语义指各语法成分所代表的含义。如检查 `i=10` 时，发现前后类型不同，则根据不同的语言规范进行不同处理，如把 10 扩展为 10.0 或者直接报错。

（4）优化及代码生成

先将源程序转化为一种内部表示形式，进而可以进行程序的优化处理以提高程序执行效率。最后生成目标程序的代码。

1.1.3 编译程序的结构

1. 编译程序总框

编译程序的结构可以按照编译过程的 5 个阶段的任务分模块进行设计。图 1.1 给出了编译程序总框图。

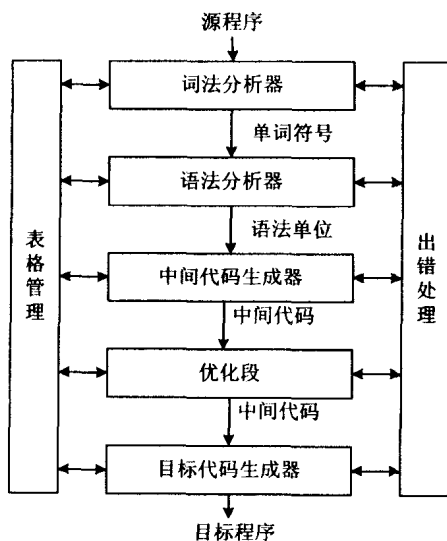


图 1.1 编译程序框架

词法分析器，又称扫描器，输入源程序，进行词法分析，输出单词符号。

语法分析器，简称分析器，语义分析与中间代码产生器，按照语义规则对语法分析器规约出（或推导出）的语法单位进行语义分析并把它们翻译成一定形式的中间代码。

优化器，对中间代码进行优化处理。

目标代码生成器，把中间代码翻译成目标代码。

2. 表格与表格管理

编译程序在工作过程中需要保持一系列的表格，以登记源程序的各类信息和编译各阶段的进展状况。

其中最重要的是符号表，它用来登记源程序中出现的每个名字以及名字的各种属性。

3. 出错处理

编译程序应该包括一组出错处理程序，用于发现错误，把有关错误的信息报告给用户。源程序中的错误通常分为语法错误和语义错误两类。

4. 遍（趟）

所谓一趟或一遍是一个编译程序在编译时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。

根据编译程序在完成翻译任务过程中需要对源程序或其中间等价物扫描的遍数，可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，可调用词法分析程序。

1.2 常考知识点及复习方法建议

本章内容较少，内容简单，出题几率不大。考点主要包括编译程序的基本概念及其执行过程等。对于编译程序的执行过程，要知道几个大的步骤，以及每个步骤所完成的功能。特别是对于编译程序的框架图，一定要掌握。

1.3 真题详细解析

【例题 1.1】（北京航空航天大学 1998 年试题）

□□□

画出编译程序的组成框图。

【解答】编译程序主要由词法分析器、语法分析器、中间代码生成器、优化段和目标代码生成器顺次完成编译任务。每一阶段的输出作为下一阶段的输入，第一阶段的输入是源程序，最后一阶段的输出是目标代码程序。每阶段的工作都和“表格管理”和“出错处理”这两个部分功能模块相关。

编译程序框架如图 1.1 所示。

【例题 1.2】（中国科学技术大学 1997 年试题）

□□□

在 Pascal 语言中，BEGIN、VAR 和 TYPE 等是保留字，而 integer、true 等是标准标识符。保留字有固定含义，而标准标识符可由用户重定义。编译器在处理这两类符号时有何区别？

【解答】编译器在进行词法分析时，词法分析器输入源程序，输出单词符号。Pascal 编译程序在词法分析阶段识别出 BEGIN、VAR 和 TYPE 等保留字，将它们和标识符、常数、运算符、界限符等其他单词相区别，而对于 integer、true 等标准标识符则并不与其他标识符相区别，统一处理。

【例题 1.3】（中国科学技术大学 1999 年试题）

□□□

填空：

程序设计语言的语法分析方法可分为两大类，即（ ）和（ ），其中前者采用了（ ）分析方法，后者采取了（ ）法。

参数传递方式有（ ）、（ ）和（ ）等几种。

LEX 是用于（ ）的工具，而 YACC 是用来（ ）的工具。

【解答】程序设计语言的语法分析方法可分为两大类，即（自下而上分析）和（自上而下分析），其中前者采用了（算符优先）分析方法，后者采取了（递归下降）法。

参数传递方式有（传名）、（传值）和（传址）等几种。

LEX 是用于（词法分析）的工具，而 YACC 是用来（分析二义文法）的工具。

【例题 1.4】（武汉大学 1999 年试题）

□□□

计算机执行用高级语言编写的程序有哪些途径？他们之间的主要区别是什么？

【解答】计算机执行用高级语言编写的程序主要途径有两种，即解释与编译。

像 BASIC 之类的语言，属于解释型的高级语言。它们的特点是计算机并不事先对高级语言进行全盘翻