

青少年 信息学奥林匹克竞赛 初级规范教程

藤 伟 主 编

实践操作性强

本书编著者全部来自于青少年信息学奥林匹克竞赛教育和管理一线，经验丰富，成绩卓著。本书的出版对于从事信息学竞赛活动的选手和教师都有极大的参考价值。

重视建立编程思想

信息学奥赛的核心内容是程序设计，而编程思想的建立是程序设计的核心，青少年选手必须在打基础的阶段初步建立编程思想。

强化编程习惯培养

优秀的程序设计者应具备沉静、严谨、高效等诸多良好的编程习惯，本教程把习惯的培养贯彻于教学的全过程。



清华大学出版社

青少年信息学奥林匹克竞赛 初级规范教程

滕 伟 主编

清华大学出版社

北 京

内 容 简 介

本书主要包括 Pascal 语言语法知识、程序设计规范及基础的信息学奥林匹克竞赛编程训练三部分内容。

本书共分 11 章。第 1 章介绍 Pascal 语言的基础知识；第 2 章介绍 Pascal 语言语句；第 3 章介绍 Pascal 语言子程序；第 4 章介绍 Pascal 语言用户自定义类型；第 5 章介绍 Pascal 类型关系和集合类型；第 6 章介绍 Pascal 数组类型；第 7 章介绍 Pascal 记录类型；第 8 章介绍 Pascal 文件类型；第 9 章介绍 Pascal 指针类型；第 10 章为 Pascal 编程训练；第 11 章为历届分区联赛初赛试题题解。本书在讲解 Pascal 语言语法的同时讲述了程序设计规范和如何建立程序设计思想，讲解过程中始终贯彻知识讲解结合上机操作。本书内容丰富并具有体系性，讲解详细清晰，程序风格严谨规范。

本书适用于初学程序设计或初学信息学奥林匹克竞赛的中小学生自学，也可作为信息学奥林匹克竞赛的基础培训教材。

版权所有，翻印必究。举报电话：010-62782989 13501256678 13801310933

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。

本书防伪标签采用特殊防伪技术，用户可通过在图案表面涂抹清水，图案消失，水干后图案复现；或将表面膜揭下，放在白纸上用彩笔涂抹，图案在白纸上再现的方法识别真伪。

图书在版编目(CIP)数据

青少年信息学奥林匹克竞赛初级规范教程/滕伟主编. —北京：清华大学出版社，2006.4

ISBN 7-302-12499-X

I. 青… II. 滕… III. 计算机课—中小学—教学参考资料 IV. G634.673

中国版本图书馆 CIP 数据核字(2006)第 007351 号

出 版 者：清华大学出版社 地 址：北京清华大学学研大厦
http://www.tup.com.cn 邮 编：100084
社 总 机：010-62770175 客户服务：010-62776969

组稿编辑：王景先

文稿编辑：张彦青

排 版 者：房利萍

印 装 者：北京鑫海金澳胶印有限公司

发 行 者：新华书店总店北京发行所

开 本：185×260 印张：22.25 字数：528 千字

版 次：2006 年 4 月第 1 版 2006 年 4 月第 1 次印刷

书 号：ISBN 7-302-12499-X/TP·8015

印 数：1~4000

定 价：32.00 元

信息学奥林匹克竞赛学习心得(代序)

信息学奥林匹克竞赛(NOI)与其他奥林匹克学科竞赛一样,都是智力和逻辑思维的竞赛。信息学还有其独特的一点,就是对于计算机应用能力和水平的考查。编程序解决问题,就像做数学题一样,看似乏味单调,但如果深入其中,则会体验到思维的乐趣和成功的快乐。同时对算法和数据结构的研究,也能使选手从更深的层次看待和使用计算机。真正将计算机看成为我所用的工具,而不是作为休闲娱乐或工作的电器设备。

通过许多年学习和参加信息学奥林匹克竞赛,以及最近两年的大学学习,具体来说我觉得在以下几个方面学习信息学对我有很大的帮助。一是培养了编写程序的能力,懂得了如何同计算机交流。编程能力对于学习计算机科学的人来说非常重要。高效正确地编写出高质量的程序能够对将来的学习、研究或工作有很大的帮助。对于不从事计算机相关专业的人来说,掌握了编程方法就多了一种使用计算机的途径——当遇到需要解决的问题时,往往不是去寻找什么软件能够解决这个问题,而是考虑自己编写程序解决问题或在其他软件的基础上改编以适应自己的要求。二是学习了程序设计语言、常用算法、数据结构、图论、组合数学等知识,这些知识都是计算机系本科培养课程中比较重要的科目。在大学学习这些知识时,由于时间短往往只能讲解理论而缺乏实践,如果能够在参加信息学竞赛的过程中自学这些知识,在解答题目中使用和实践这些理论,无疑能够加深对这些知识的理解和掌握。三是培养了学习的能力,参加信息学竞赛所需要掌握的知识很广泛,没有哪一本书能够覆盖全部的知识点,同时新算法新知识也层出不穷,这就要求选手有较强的自学能力。从获取学习资料,到看懂理解,再到和别人讨论交流,最后到能够使用,这种学习能力的锻炼对于参赛选手来说是终生受益的。

对于决定参加信息学奥林匹克竞赛的初学者来说,我有几点个人建议。首先要熟练掌握一门编程语言,这是最基本的要求。Pascal 是非常好的入门学习的语言,它的程序简洁漂亮,语法严格,非常适合初学者。不要因为 Pascal 比较过时、使用不多,就认为它不好。其实编程语言都是触类旁通的,掌握一门新语言是很简单的,最难的是掌握编程的思想。学习 Pascal 是从不会编程到会编程的非常好的方法。这一点,要求初学者学会基本语法的使用,能够编写简单的程序,学会与竞赛相关的高级编程方法(数组清零和计时中断等),另外熟练使用集成开发环境(IDE)编写、调试程序也是非常重要的。其次要打好基础——这是非常重要的,打基础就是要熟练掌握基本的算法和基本的数据结构,能够很快地编写出代码。这一步做好了,就能够通过编程解决很多的竞赛题目了,同时也为学习更复杂的算法和数据结构做好了准备。对于初学者,最后的建议就是不要过分看重竞赛成绩,能够掌握知识、培养能力并且每天都在进步是最关键的,我身边有很多同学竞赛并没有取得多好的成绩,但参与信息学竞赛使得他们无论是学习能力还是对专业知识的理解都比其他同学有很大的优势。

最后,热烈祝贺这本书的出版,希望这本书能够成为广大青少年开启信息学大门的钥匙,也祝愿你们能够在未来的竞赛之路中取得好的成绩,培养锻炼出很强的逻辑思维能力和学习能力。

前言

计算机的深入普及和信息技术的迅速发展，对人类社会产生了难以估量的深远影响，政治、经济、文化、国防和科技研发无不紧紧地与信息技术连接在一起，信息社会将成为新世纪的一个标志。

“计算机的普及要从娃娃抓起”是科教兴国的一项重要内容。程序设计是信息技术领域的根本，而软件业的发展水平则是信息技术发展的国际公认标尺。从基础教育抓起，**从青少年阶段开始培养具有规范严谨的程序设计习惯**，是培养大量高级信息科技人才的必由之径，也是振兴中国信息产业的关键。

信息学奥林匹克竞赛是为信息学的普及推广和优秀信息人才的选拔而设立的。但由于信息学奥林匹克竞赛教育培训涵盖面广，并且尚未形成体系，教育培训师资又相对匮乏，致使信息学编程教育培训的普及推广还需要我们付出长期的努力。

这套教程是在中国飞扬奥林匹克竞赛信息学奥林匹克竞赛教育培训教材的基础上改编而成的，是长期从事中小学信息技术教学培训达 15 年，并取得优异成绩的全国金牌教练滕伟老师的呕心之作。

这套教程清晰地提炼出信息学奥林匹克竞赛青少年培训的**三大原则**：

- 化繁为简，搭建简练实用的知识结构，使基础培训有章可循。
- 严格执行规范，保证教育培训的质量与高效。
- 以兴趣为引导，以鼓励为动力，坚持不懈地培养青少年沉静、严谨、乐观、自信的心理素质和意志品质。

这套教程适于青少年自学研读，更适合一线培训教师授课参考，其特点如下：

- 从学习者的认知规律出发，安排简练实用的知识结构，力求使受训者轻松入门。
- 注重基础能力的培养，循序渐进地培养良好严谨的编程素质。
- 统筹计划教育培训的时间，明确每章节知识容量与课时安排。
- 突出实际操作和编程能力的训练，并配有大量的基础试题和编程试题。

为配合读者更好地使用这套教程，中国飞扬奥林匹克竞赛在其网站（www.fnoi.net）上设专栏解答培训中出现的疑难问题，讨论最新课题，并向广大师生提供信息学奥林匹克竞赛教育培训在线支持。

本书由滕伟主编，参加编写的人员还有张玲、王慧英、王育龄、耿昱、张艳津、王勇、王锋、杨欣、杨郡、庞铮、范镇、王燕、刘晓东、喻梅、李平和李娜等。

本书在编写过程中，得到了北京大学信息科学技术学院李文新教授和清华大学出版社第三事业部王景先老师的大力支持和帮助，在此表示衷心的感谢。

作者长期从事青少年信息学奥林匹克竞赛教育 10 多年，根据自己的教案整理完成本书内容，由于时间仓促，难免会有不足之处，欢迎大家批评指正。

编者

目 录

第 1 章 Pascal 语言基础	1	1.8.1 标准函数	27
1.1 了解程序结构	1	1.8.2 算术运算函数	27
1.1.1 Pascal 程序结构	1	1.8.3 转换函数	29
1.1.2 Pascal 程序编写格式	2	1.8.4 顺序函数	30
练习题	4	1.8.5 判断函数	30
1.2 上机操作一	5	1.8.6 随机函数	30
1.2.1 安装 Pascal	5	练习题	31
1.2.2 上机操作	5	1.9 赋值语句	32
1.3 语法图和词法单位	7	1.9.1 赋值语句语法图	32
1.3.1 语法图图形符号	7	1.9.2 赋值语句执行过程	32
1.3.2 程序结构描述	7	1.9.3 使用注意事项	32
1.3.3 词法单位	9	练习题	33
练习题	10	1.10 输出调用语句	34
1.4 上机操作二	11	1.10.1 输出调用语句	
1.4.1 熟悉键盘	11	语法图	34
1.4.2 学习程序的编辑操作	11	1.10.2 输出调用语句	
1.4.3 块操作	12	执行过程	34
1.5 标准类型	14	1.10.3 输出格式	34
1.5.1 数据类型	14	1.10.4 用户自定义输出格式	36
1.5.2 标准类型	15	练习题	37
练习题	19	1.11 输入调用语句	37
1.6 常量、变量的说明	19	1.11.1 输入调用语句	
1.6.1 常量说明	20	语法图	38
1.6.2 变量说明	21	1.11.2 输入调用语句	
1.6.3 说明语句的注意事项	21	执行过程	38
练习题	22	1.11.3 数据输入	38
1.7 Pascal 语言的算符	22	练习题	40
1.7.1 算符介绍	23	第 2 章 Pascal 语句	42
1.7.2 算术运算符	23	2.1 复合语句	42
1.7.3 关系运算符	24	2.1.1 复合语句语法图	42
1.7.4 布尔运算符	24	2.1.2 复合语句的执行过程	42
1.7.5 算符的优先级	24	练习题	43
练习题	26	2.2 如果语句	43
1.8 标准函数	27	2.2.1 如果语句语法图	44

2.2.2 如果语句的执行过程.....	44	2.8 编程练习题.....	80
2.2.3 如果语句嵌套原则.....	47	2.8.1 理解循环.....	82
练习题.....	49	2.8.2 模拟编程.....	83
2.3 情况语句.....	51	2.8.3 累加、累乘两小时.....	83
2.3.1 情况语句语法图.....	51	2.8.4 求最值.....	84
2.3.2 情况语句的执行过程.....	52	2.8.5 统计问题.....	84
2.3.3 情况语句使用说明.....	53	2.8.6 归纳分析.....	84
2.3.4 情况语句嵌套.....	54	2.8.7 枚举问题.....	84
练习题.....	55	第3章 子程序.....	87
2.4 循环语句.....	56	3.1 过程与函数简介.....	87
2.4.1 重复语句语法图.....	56	3.1.1 子程序的概念.....	87
2.4.2 Repeat 语句的 执行过程.....	57	3.1.2 子程序的作用.....	87
2.4.3 当语句语法图.....	58	3.1.3 子程序的分类.....	88
2.4.4 当语句的执行过程.....	58	练习题.....	89
2.4.5 两种循环语句的比较.....	60	3.2 函数的定义与调用.....	89
2.4.6 For 语句语法图.....	62	3.2.1 函数的定义.....	89
2.4.7 For 语句的执行过程.....	62	3.2.2 函数的调用.....	91
2.4.8 For 语句与 Repeat 语句 和 While 语句的比较.....	63	练习题.....	92
2.4.9 累加、累乘的使用.....	64	3.3 过程的定义与调用.....	93
2.4.10 控制循环流程的 过程语句.....	65	3.3.1 过程的说明.....	93
练习题.....	66	3.3.2 过程的调用.....	95
2.5 多重循环.....	67	练习题.....	96
2.5.1 打印图形.....	67	3.4 参数传递.....	97
2.5.2 人口增长问题.....	69	3.4.1 值参数.....	97
练习题.....	69	3.4.2 变量参数.....	97
2.6 程序设计初步.....	70	3.4.3 选择参数的原则.....	100
2.6.1 程序设计方法.....	70	练习题.....	100
2.6.2 程序质量.....	71	3.5 标识符的作用域.....	101
2.6.3 程序优化.....	72	3.5.1 全程变量和局部变量.....	101
练习题.....	73	3.5.2 标识符的作用域.....	103
2.7 编程风格.....	74	3.5.3 将子程序值传 递给主程序.....	104
2.7.1 程序简要说明.....	74	练习题.....	106
2.7.2 程序版式.....	74	3.6 子程序的嵌套、递归和递推.....	106
2.7.3 命名规则.....	77	3.6.1 子程序嵌套.....	106
2.7.4 基本语句.....	78	3.6.2 子程序递归.....	109
练习题.....	80	3.6.3 递推.....	114
		练习题.....	115

3.7 子程序的使用	121	6.1.3 数组变量说明	146
3.7.1 子程序的结构化使用	121	6.1.4 数组类型常量的定义	147
3.7.2 递归的深入理解	122	6.1.5 数组类型的操作	147
练习题	123	练习题	147
第4章 Pascal 语言用户		6.2 一维数组应用	148
自定义类型	125	6.2.1 数据交换	148
4.1 枚举类型	125	6.2.2 排序算法	149
4.1.1 用户自定义类型	125	6.2.3 打印杨辉三角形	152
4.1.2 枚举类型定义	126	练习题	154
4.1.3 变量说明	126	6.3 二维数组应用	154
4.1.4 枚举类型数据的操作	127	6.3.1 观察行列关系	154
练习题	129	6.3.2 方阵转置	155
4.2 子界类型	129	6.3.3 打印杨辉三角形	156
4.2.1 子界类型的定义	130	6.3.4 成绩统计	156
4.2.2 变量说明	130	6.3.5 矩阵相乘	157
4.2.3 子界类型的操作	131	练习题	158
练习题	132	6.4 二维数组练习	160
第5章 Pascal 类型关系和		6.5 布尔数组	170
集合类型	133	6.5.1 布尔数组和集合类型	
5.1 类型的关系	133	的关系	170
5.1.1 类型的一致性	133	6.5.2 应用举例	170
5.1.2 类型的相容性	134	练习题	171
5.1.3 赋值相容	135	6.6 字符串类型	171
5.1.4 过程和函数参数传递的		6.6.1 字符串类型定义	171
相容性	136	6.6.2 字符串的变量说明	172
练习题	136	6.6.3 字符串类型数据	
5.2 集合类型	137	的操作	172
5.2.1 集合的概念	137	6.6.4 字符串使用	174
5.2.2 集合变量说明	138	练习题	176
5.2.3 集合的操作	138	第7章 Pascal 记录类型	177
5.2.4 集合的使用	140	7.1 记录类型	177
练习题	143	7.1.1 记录类型定义	177
第6章 Pascal 数组类型	144	7.1.2 记录变量说明	178
6.1 数组类型	144	7.1.3 类型常量说明	179
6.1.1 数组的引入	144	7.1.4 记录类型操作	179
6.1.2 数组类型的定义	145	7.1.5 记录类型定义的嵌套	181
		7.1.6 开域语句	182
		练习题	184

7.2 记录类型练习	184	第 10 章 编程训练	235
7.2.1 简单练习	185	10.1 进制转换	235
7.2.2 记录数组的练习	188	10.1.1 进制介绍	235
练习题	190	10.1.2 十进制二进制 数值转换	235
第 8 章 Pascal 文件类型	191	10.1.3 二进制、十进制、十六进制 数值转换	237
8.1 文件类型	191	练习题	238
8.1.1 文件的概述	191	10.2 计算机编码	239
8.1.2 文件类型定义	192	10.2.1 原码	239
8.1.3 变量说明	192	10.2.2 反码	240
8.1.4 处理文件的过程 和函数	193	10.2.3 补码	240
8.1.5 文件的分类	195	10.3 高精度训练	241
练习题	196	10.3.1 高精度计算的引入	241
8.2 文本文件的使用	196	10.3.2 数据输入	243
8.2.1 文本文件的特点	196	10.3.3 估算结果位数	244
8.2.2 文本文件的操作	196	10.3.4 计算和进位问题	245
8.2.3 文本文件练习	197	10.3.5 输出问题	247
8.2.4 标准文件	198	10.4 高精度练习一	247
练习题	207	练习题	252
8.3 类型文件的使用	209	10.5 编程完成进制转换	253
8.3.1 类型文件的特点	209	练习题	257
8.3.2 类型文件的练习	209	10.6 高精度练习二	258
练习题	211	10.7 循环枚举	263
第 9 章 Pascal 指针类型	215	第 11 章 历届分区联赛初赛 试题解析	274
9.1 指针类型	215	第七届 全国青少年信息学奥林匹克 联赛普及组初赛试题和解答	274
9.1.1 指针类型概念	215	第七届 全国青少年信息学奥林匹克 联赛提高组初赛试题和解答	281
9.1.2 指针的类型定义	216	第八届 全国青少年信息学奥林匹克 联赛普及组初赛试题和解答	289
9.1.3 指针变量说明	216	第八届 全国青少年信息学奥林匹克 联赛提高组初赛试题和解答	297
9.1.4 指针的变量操作	216	第九届 全国青少年信息学奥林匹克 联赛普及组初赛试题和解答	304
练习题	218	第九届 全国青少年信息学奥林匹克	
9.2 指针类型应用	221		
9.2.1 链表结点定义	222		
9.2.2 链表的建立	222		
9.2.3 链表的操作	224		
9.2.4 环形链表结构	230		
9.2.5 双向链表结构	232		
练习题	232		

联赛提高组初赛试题和解答.....309	附录 A Pascal 编译错误对照表..... 331
第十届 全国青少年信息学奥林匹克	附录 B Pascal 运行错误对照表..... 338
联赛普及组初赛试题和解答.....316	参考文献..... 341
第十届 全国青少年信息学奥林匹克	
联赛提高组初赛试题和解答.....322	

第 1 章 Pascal 语言基础

教学目的：掌握 Pascal 语言的基础知识，认识编程过程，能够上机进行简单的程序设计。

教学内容：Pascal 语言的程序结构、词法单位、标准类型、常量、变量、算符、标准函数、表达式、赋值语句、输入语句、输出语句等基础知识和上机操作。

教学重点：掌握 Pascal 语言的基础知识。

教学难点：认识编程过程，理解编程思想。

1.1 了解程序结构

- 掌握 Pascal 程序结构：程序首部、描述部分、执行部分
- 掌握 Pascal 程序简单语法规则：词法分隔、语句分隔
- 掌握 Pascal 程序格式规范：缩排、加注释
- 掌握 Pascal 程序编程原则：先说明后使用
- 课时：1 小时

1.1.1 Pascal 程序结构

无论是什么版本的 Pascal 语言，也无论程序多么复杂，程序的结构形式是固定的。我们对照着下面的程序来说明。

【例 1-1】观察如下程序结构：

```
Program Square;  
Var  
  a, l, s: Integer; {a 表示正方形边长, l 表示周长, s 表示面积}  
Begin  
  a := 5;  
  l := 4 * a;  
  s := a * a;  
  WriteLn('l=', l, 's=', s);  
End.
```

本程序用来计算边长为 a 的正方形的周长和面积。

我们这里暂且不必考虑各种语句的具体含义，只看程序的结构。一个 Pascal 语言的程序分为三部分：**程序首部**、**分程序**、**注释**。

(1) **程序首部** 就是程序的开头部分。它包括以下几部分：

Pascal 语言程序的标志：必须用“Program”来表示 Pascal 程序，每一个 Pascal 程序都以此开头。

程序名称：本程序的名字为“Square”。

程序首部最后加一个分号“;”。

(2) **分程序** 也称为程序体，它是程序的主体。它分为两部分：说明部分和执行部分。

① **说明部分** 用来描述程序中用到的数据的属性。它是由说明语句组成的。凡 Pascal 程序中用到的如标号、符号常量、类型、变量、过程和函数等数据，都要在说明部分进行定义，即不允许使用未定义的数据。

② **执行部分** 以 Begin 开始，以 End. 结束的这一段程序，就是执行部分。它是由一系列语句组成，一条语句执行一定的功能，所有语句完成程序设计的任务。

注：语句之间用分号“;”分隔。本程序中用到了三个赋值语句、一个输出语句。

(3) **注释** 就是写在程序中起注释作用的文字，计算机不执行此内容。注释内容必须用 { } 括起来。在程序的任何位置都可以插入一个或多个注释。

 **注意：** 说明部分是描述要用到的数据的属性，执行部分是对描述的数据进行操作。它们体现了 Pascal 语言最重要原则之一：先说明后引用。

1.1.2 Pascal 程序编写格式

1. 编写规则

词之间要有间隔，用空格或回车分隔。

Pascal 程序的书写方法比较灵活，允许一行中写多条语句(但要用分号“;”分隔)，每行的字符数不得超过 128 个。

2. 编写规范

采用缩排的方式，即采用阶梯格式突出层次，便于阅读和调试。

如下面的书写格式，Begin 和 End. 之间的每行语句空两个空格，并列意义的语句对齐，从属关系的语句缩排两个空格。

```
Begin
  a := 5;
  l := 4 * a;
  s := a * a;
  WriteLn('l=', l, 's=', s);
End.
```

加入适当的注释，观察下面的程序。

```
Program Example;
Var
  i, j, n: Integer;
Begin
  n := 4;
  For i:=1 To n Do Begin
    {第 i 行输出 n-i+1 个空格}
    For j:=1 To n-i+1 Do Write(' ');
    {第 i 行输出 2*i-1 个星号}
```

```
For j:=1 To 2*i-1 Do Write('*');  
{换行准备输出下一行}  
WriteLn;  
End;  
End.
```

【课上练习】

熟悉程序结构和编程规范。

【例 1-2】编程计算长方形的面积。已知长方形的长和宽(a, b 为整数), 计算长方形的周长和面积, $l=2*(a+b)$, $s=a*b$ 。

【源程序】

```
Program Rectangle;  
Var  
  a, b, l, s: Integer;  
Begin  
  a := 10; b := 20;  
  l := 2 * (a + b);  
  s := a * b;  
  WriteLn('l=', l);  
  WriteLn('s=', s);  
End.
```

【例 1-3】编程计算圆的面积。已知圆的半径(r 为整数), 计算圆的周长和面积, $l=2*\pi*r$, $s=\pi*r*r$ 。 π 可以直接写为 3.14。

【源程序】

```
Program Circle;  
Var  
  r: Integer, l, s: Real;  
Begin  
  r := 10;  
  l := 2 * 3.14 * r;  
  s := 3.14 * r * r;  
  WriteLn('l=', l);  
  WriteLn('s=', s);  
End.
```

【例 1-4】编程计算两个数的加、减、乘。已知两个数(a, b 为整数), 计算 $a+b$, $a-b$, $a*b$ 。

【源程序】

```
Program Calculate;  
Var  
  a, b, x1, x2, x3: Integer;  
Begin  
  a := 12; b := 100;  
  x1 := a + b; x2 := a - b; x3 := a * b;  
  WriteLn('a+b=', x1);  
  WriteLn('a-b=', x2);
```

```
WriteLn('a*b=', x3);  
End.
```

练 习 题

1. Pascal 语言程序结构包括_____、_____、_____。
2. 程序首部以_____开始, 以_____结束。
3. _____部分是描述要用到的数据的属性, _____部分是对描述的数据进行操作。它们体现了 Pascal 语言最重要原则之一: _____。
4. 执行部分以_____开始, 以_____结束。
5. 语句之间用_____分隔。
6. Pascal 程序中词与词用_____或_____分隔。
7. 划分出下面程序的程序首部、程序名、说明部分、执行部分。

【源程序】

```
Program Example7;  
Var  
  n, a, b, c: Integer;  
Begin  
  n := 287;  
  a := n Mod 10;  
  b := n Div 10 Mod 10;  
  c := n Div 100;  
  WriteLn('a=', a);  
  WriteLn('b=', b);  
  WriteLn('c=', c);  
End.
```

8. 观察下面程序, 调出程序中的编写错误。

【源程序】

```
Program Example8;  
Var  
  n, m, r: Integer;  
Begin  
  n := 100  
  m := 200  
  r := n  
  n := m  
  m := r  
  WriteLn(n:5, m:5);  
End;
```

1.2 上机操作一

- 熟悉 Pascal 集成环境
- 掌握 Pascal 输入、编译、运行、存盘
- 强化 Pascal 程序格式规范
- 课时: 1.5 小时

1.2.1 安装 Pascal

1. Pascal 系统主要文件名

Turbo.exe	-----	Pascal 语言执行程序, 最重要的文件。
Turbo.tpl	-----	Pascal 语言的标准单元文件。
Turbo.tph	-----	Pascal 语言的帮助文件。

2. 建立目录

在用户的硬盘上建立目录, 目录名为 TP7, 存放 Pascal 系统文件。

3. 复制

将上述三个文件复制到目录 TP7 下。

4. 建立快捷方式

在用户的桌面上创建指向目录 TP7 下的 Turbo.exe 文件。

1.2.2 上机操作

1. 启动 Turbo

双击图标



即可。

2. Turbo 环境介绍

熟悉菜单栏, 下拉菜单选择 Alt+首字母, 用←、↑、→、↓移动, 空格、回车执行。

3. 建立新文件

操作: Alt+F, 光标移至 New 回车。

编辑窗口: 文件名为 noname00.pas、光标在第一行第一列。

4. 编辑程序

输入计算长为 10, 宽为 2 的长方形的面积的程序。

5. 程序存盘

将输入好的程序存盘。

操作: Alt+F、光标移至 Save 回车。给程序起一个外部文件名即可。

6. 运行程序

让计算机运行此程序。

操作: Ctrl+F9。

7. 观察结果

如果运行成功, 观察运行结果。

操作: Alt+F5。

8. 再次存盘

将运行无误的程序保存好。

操作: Alt+F、Save 或者直接按 F2 功能键。

9. 退出

操作: Alt+F、Exit 或者直接按 Alt+X。

10. 打开文件

将已编写过的程序打开修改。

操作: Alt+F、Open 或者直接按 F3 键。

选择文件: 各项之间用 Tab 键切换。Files 区域内用箭头键选择。

11. 窗口切换

在一个编辑窗口中, 只能编辑一个 Pascal 程序。Turbo Pascal 集成环境中允许同时打开多个编辑窗口, 窗口之间的切换操作作用 F6 键。关闭当前窗口的操作作用 Alt+F3。

【课上练习】

【例 1-5】输入第 1.1 节的练习题编译、运行、存盘。

【例 1-6】输入如下程序运行, 熟悉编辑操作和运行程序。

【源程序】

```
Program Example1_6;
Var
  n, i, j: Integer;
Begin
  n := 10;
  For i:=1 To n Do Begin
    For j:=1 To n-i Do Write(' ');
    For j:=1 To 2*i-1 Do Write('*');
    WriteLn;
  End;
End.
```

1.3 语法图和词法单位

- 掌握语法图各个图形符号所表示的含义
- 掌握标识符的起名规则
- 了解保留字、标准标识符、用户自定义标识符的关系
- 课时：1 小时

1.3.1 语法图图形符号

用自然语言描述语法规则，往往语言冗长，且不严格。所以常常采用语法图来表示语法规则。下面我们就介绍几种语法图的符号，如图 1-1 所示。

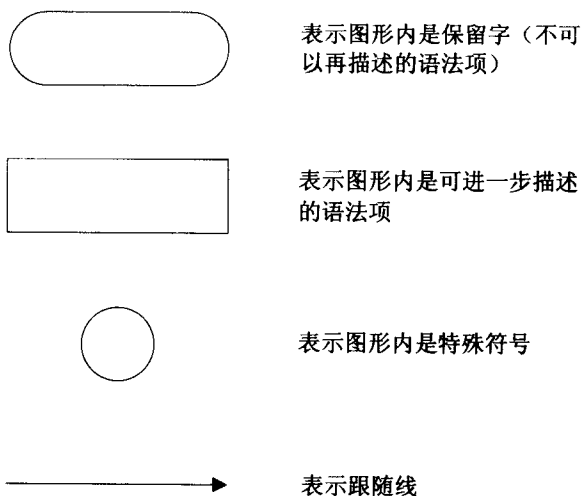


图 1-1 语法图图形符号

1.3.2 程序结构描述

我们下面就用语法图来全面描述程序结构。

(1) Pascal 程序结构，如图 1-2 所示。

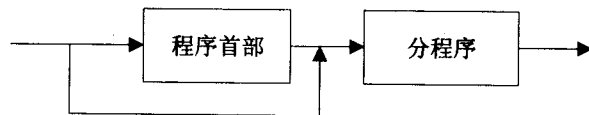


图 1-2 程序结构语法图

(2) 程序首部，如图 1-3 所示。