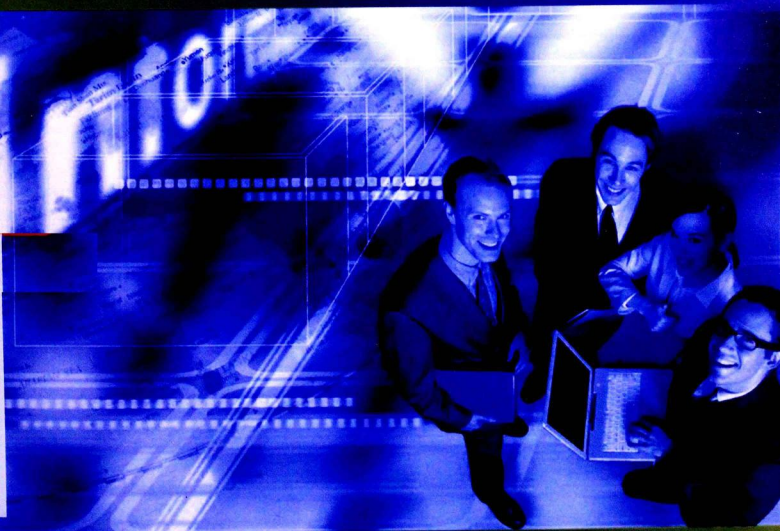


全国青少年信息学奥林匹克竞赛指导丛书

Free Pascal 程序设计

FREE PASCAL CHENGXUSHEJI 林盛华◎编著



江南大学出版社
JIANGNAN UNIVERSITY PRESS

Free Pascal 程序设计

林盛华◎编著



暨南大学出版社
JINAN UNIVERSITY PRESS

中国·广州

图书在版编目 (CIP) 数据

Free Pascal 程序设计/林盛华编著. —广州: 暨南大学出版社, 2013. 7

(全国青少年信息学奥林匹克竞赛指导丛书)

ISBN 978 - 7 - 5668 - 0653 - 6

I. ①F… II. ①林… III. ①Pascal 语言—程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字 (2013) 第 150110 号

出版发行: 暨南大学出版社

地 址: 中国广州暨南大学

电 话: 总编室 (8620) 85221601

营销部 (8620) 85225284 85228291 85228292 (邮购)

传 真: (8620) 85221583 (办公室) 85223774 (营销部)

邮 编: 510630

网 址: <http://www.jnupress.com> <http://press.jnu.edu.cn>

排 版: 广州市天河星辰文化发展部照排中心

印 刷: 广东省农垦总局印刷厂

开 本: 787mm × 1092mm 1/16

印 张: 10.75

字 数: 261 千

版 次: 2013 年 7 月第 1 版

印 次: 2013 年 7 月第 1 次

定 价: 32.00 元

(暨大版图书如有印装质量问题, 请与出版社总编室联系调换)

前 言

全国青少年信息学（计算机）奥林匹克竞赛（早期称为青少年计算机程序设计竞赛）是旨在广大青少年中普及计算机教育，推广计算机应用的一项学科性竞赛活动。从1984年开始举办全国性竞赛。而自从1989年我国参加第一届国际信息学奥林匹克竞赛（International Olympiad in Informatics，简称IOI）以来，全国青少年计算机程序设计也更名为全国青少年信息学（计算机）奥林匹克竞赛（National Olympiad in Informatics，简称NOI）。

实践证明，该竞赛对于选拔计算机后备人才起到了立竿见影的效果，经由该比赛选拔出来的人才，如今已在国内IT行业甚至国际上渐领风骚，每年毕业的信息学学生也是各大IT巨头青睐的人才资源。不无夸张地说，信息学竞赛是一项有效的人才培养、选拔机制，它的推广和普及或可谓功在当代，利在千秋。

作为信息学竞赛的语言之一，Pascal语言语法严谨，层次分明，程序易写，具有很强的可读性，与其他编程语言如C、C++来说，Pascal语言更适合初学者学习，其严格的结构化形式、丰富完备的数据类型、运行效率高、查错能力强等特点，都有益于培养良好的程序设计风格和习惯。

一本好的入门书，使人如沐春风，豁然开朗，进而渐入佳境。一本生搬硬套的书籍，则不但浪费时间，更是扼杀兴趣。所幸本书编者具有十几年的信息学竞赛辅导经验，所辅导学生有113人次获得信息学联赛（NOIP）一等奖，4人获得全国信息学（NOI）一等奖，2人入选国家集训队。编者深谙初学者的学习心理和认知特点，不仅在内容结构的设计上循序渐进，而且不拘泥于语言的学习，重在通过思路点拨来培养和启发学习者的思维模式和程序设计能力。

本书内容共分十章，分别讲解Free Pascal语言基础、顺序结构、分支结构、循环结构、数组、函数与过程、文件、高精度计算算法、指针等内容。每章均配备相关例题与练习，帮助读者熟练掌握编写程序的方法和技巧，同时侧重培养学生的数学思维与算法设计能力。当然，对于初学者来说，阅读本书时，不仅要思考，同时还要勤于上机练习，只有通过设计程序、编写程序、调试程序，才能更快速有效地提升编程水平。

本书按照全国青少年信息学奥林匹克竞赛的大纲要求编写，结构严谨，理论联系实际，特别适合作为中小学开展信息学奥林匹克竞赛的培训教材。

由于编者水平有限，书中若有疏漏之处，恳请各位读者不吝指正。

编者
2013年6月

目 录

前 言	1
第一章 Pascal 语言的基本知识	1
第一节 Pascal 程序的基本组成	1
第二节 Pascal 语言系统的使用	2
第三节 Free Pascal 的菜单和窗口	6
第二章 Pascal 语言的基础知识	9
第一节 Pascal 基本字符集、标识符与保留字	9
第二节 Pascal 的标准数据类型	11
第三节 标准函数	14
第四节 Pascal 语言的运算符	15
第三章 顺序结构程序设计	20
第一节 常量和变量	20
第二节 输入输出语句	22
第四章 分支结构程序设计	30
第一节 Pascal 中的布尔(逻辑)类型	30
第二节 关系表达式与布尔表达式	31
第三节 简单的 If 语句	32
第四节 复合语句	34
第五节 If 语句的嵌套	35
第六节 Case 语句	35
第五章 循环结构和数组	40
第一节 For 循环语句	40
第二节 While 循环语句	42
第三节 直到型循环(Repeat - Until 语句)	44
第四节 多重循环	46
第五节 控制循环流程或程序的语句	49



第六节 一维数组	50
第七节 二维数组	54
第六章 字符串和进制的转换	59
第一节 字符串	59
第二节 计算机中数的表示	64
第三节 不同进制数之间的转换	65
第七章 模块化程序设计	73
第一节 自定义函数	73
第二节 过程的定义	79
第三节 变量参数	81
第四节 全程变量和局部变量	83
第五节 函数和过程的区别	84
第八章 文件操作、枚举、子界、集合和记录类型	88
第一节 文件的概念	88
第二节 文件的函数和过程	89
第三节 枚举类型	90
第四节 子界类型	94
第五节 集 合	96
第六节 记录类型	102
第九章 高精度计算	112
第一节 高精度加法	113
第二节 高精度减法	114
第三节 高精度与单精度的乘法	117
第四节 高精度与高精度的乘法	119
第五节 高精度与单精度的除法	121
第六节 高精度与高精度的除法	123
第十章 指针变量及线性链表	137
第一节 指针变量	137
第二节 线性链表	142
附录一	155
附录二	160
附录三	162

第一章 Pascal 语言的基本知识

【知识要点】

1. 程序的基本结构
2. 语言系统
3. 程序菜单

第一节 Pascal 程序的基本组成

我们首先来了解 Pascal 程序的基本结构。

例 1-1 就是用 Pascal 语言编写的一个小程序,在计算机上运行它,就会在屏幕上显示出 Google 公司两位创始人的名字。

【例 1-1】观察下面的程序结构:

```
program google;  
begin  
    writeln('Sergey Brin');| google 公司的 CEO,创始人之一|  
    writeln('Larry Page');|//google 公司的另一创始人|  
end.
```

上述程序第一行称为程序首部。其中用括号(注释可以用| |或//来表示)括起来的内容是注释。注释除了给人看,增加程序的可读性外,对程序编译和运行不起作用。一个程序可以包含多个出现在不同处的注释,亦可无注释。程序从 begin 到 end 都是执行(语句)部分。

1. 程序首部

上例中的第一行称为程序首部。program 是保留字,接着是程序名(可依据“标示符”规则自行定义),最后以分号表示程序首部结束,下面是程序主体的开始。程序首部在一个 Pascal 程序中并非必须出现,它是可选的。写上它仅起到文档作用。

2. 程序体

(1) 说明部分。

说明部分用于定义和说明程序中用到的数据,由单元说明、标号说明、常量说明、类型说明、变量说明、函数说明或过程说明组成,并且这些数据的说明次序必须按照以上次序。但



是一个简单的程序也可以不包含说明部分,也就是说说明部分是可选的。

(2) 执行部分。

执行部分描述了程序要执行的操作。它必须以 Pascal 保留字 begin 开始,以保留字 end 后跟句点结束,其间是一些执行具体操作的语句,并且以分号作为语句之间的分隔符。begin 和 end 必须成对出现,这是一个 Pascal 程序所必须有的。紧跟 end 之后的句号表示执行部分的结束,也表示整个程序的结束。

(3) 注释部分。

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!!或//括起来。在程序的任何位置都可以有注释。

(4) 一个完全的 Pascal 程序结构。

program 程序名;

uses

 已知单元说明;

label

 标号说明;

const

 常量说明;

type

 类型说明;

var

 变量说明;

function

 函数说明;

procedure

 过程说明;

begin

 语句;

 语句;

 语句;

end.

第二节 Pascal 语言系统的使用

目前,常用的 Pascal 语言系统有 Free Pascal,下面我们就来学习 Free Pascal 系统的使用。

1. Pascal 的启动

双击桌面上 Free Pascal 对应的图标,或单击“开始”->“所有程序”找到 Free Pascal 图标,就能启动 Free Pascal,启动后如图 1-1 所示。



图 1-1 Free Pascal 的系统集成环境

2. 新建程序窗口

按 F10 运行主菜单,选择 File 菜单,执行其中的 New 命令,就可建立一个新的程序窗口(默认文件名为 Noname01.pas)。

3. 程序的编辑

在当前程序窗口中,一行一行地输入例 1-1 程序(如图 1-2 所示)。事实上,程序窗口是一个全屏编辑器。



图 1-2 在编辑窗口中编辑程序

4. 程序的保存

保存程序可以通过按 F2 键(或执行 File 菜单中的 Save 命令)来保存程序。第一次保存文件时屏幕上会出现一个对话框要求输入文件名(默认扩展名为 .pas)。

5. 程序的编译

当程序输入完毕之后,要先按 Alt + F9 (或执行 Compile 菜单中的 Compile 命令)对程序进行编译。如果程序有语法错误,则会在程序窗口的第一行处显示第一个红色错误信息;若无语法错误,则窗口正中央会出现一个对话框,提示编译成功(如图 1-3 所示)。

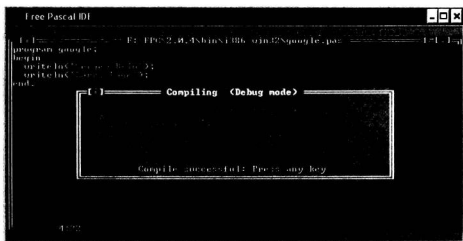


图 1-3 程序编译成功

6. 运行程序

程序的运行可以通过 Run 菜单中的 Run 命令,或直接按快捷键 Ctrl + F9,就可以在用户窗口中输出运行结果。通常在程序运行结束后系统回到 Pascal 系统的集成环境,因此要查看运行结果,要按 Alt + F5 将屏幕切换到用户屏幕(如图 1-4 所示)。

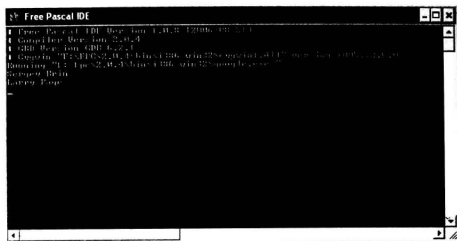


图 1-4 例 1-1 程序的运行结果

【例 1-2】输入程序,按要求操作,观察编译、运行的结果。

编程计算长方形的面积。已知长方形的长和宽(a, b 都为整数),计算长方形的周长 l 和面积 s 。

【源程序】

```
program count;
var
  a, b, l, s: integer;
begin
  a := 100;
  b := 50;
  l := 2 * (a + b);
  s := a * b;
  writeln('l = ', l);
  writeln('s = ', s);
end.
```

要求:按下表的操作顺序操作,观察结果。

序号	操作要求	结果
1	把程序保存在 D 盘根目录,文件名存为:count.pas	
2	编译程序	
3	删去 program 所在的一行,然后编译程序	
4	删去 begin,然后编译程序(编译后加回)	
5	删去第一个 writeln 后的“;”号,编译程序(编译后加回)	
6	删去第二个 writeln 后的“;”号,编译程序	
7	把最后一行的 end 后的“.”号删除,编译程序(编译后加回)	
8	运行程序,观察程序运行结果	



强化训练

一、填空题

1. 程序的首部以 _____ 开始,以 _____ 结束。
2. 执行部分以 _____ 开始,以 _____ 结束,语句之间用 _____ 分隔。

二、改错题(观察下面的程序,指出程序中错误的地方并修改)

```
program example;  
var  
    x,a,b,c:integer;  
begin  
    n:=128;  
    c:=n mod 10;  
    b:=(n div 10) mod 10;  
    a:=n div 100;  
    writeln('a=',a);  
    writeln('b=',b)  
    writeln('c=',c)  
end;
```

第三节 Free Pascal 的菜单和窗口

* 供自学使用

选取菜单条的方法有多种:用 F10 键可以激活菜单条,或者用 Alt 键与菜单名称的第一个字母的组合键来选取该菜单。如 Alt + F 即选中 File 菜单。也可以用鼠标单击进行选择。

Free Pascal 的主菜单包括:

File Edit Search Run Compile Debug Tools Options Window Help

每个菜单名称下面有各自的条目。下面选择常用的操作来介绍。

File(文件)菜单包括:

New		建立新文件
Open...	F3	打开一个文件
Save	F2	保存一个文件
Save as...		文件另存为
Change dir...		更换工作目录
Exit	Alt + X	退出 Free Pascal

名称后面加…的表示命令执行时将弹出一个对话框供选择,有些命令后面有热键说明,请在平常的练习中,观察它所执行的功能。

Edit(编辑)菜单包括:

Undo	Alt + Bksp	撤销操作
Redo		恢复操作
Cut	Shift + Del	剪切
Copy	Ctrl + Ins	拷贝
Paste	Shift + Ins	粘贴

如果命令为灰色,则表示不适于当前执行,即使选中执行也没有效果。

Search(搜索)菜单包括:

Find...	搜索关键字
Replace...	替换

Run(运行)菜单包括:

Run	Ctrl + F9	运行
Step over	F8	执行下一句
Trace into	F7	单步运行
Parameters...		设置程序参数

这里 Run 命令在执行前如果没有进行 Compile 与 Make 的操作,则首先执行这两个命令,再运行程序。Parameters 用于给程序设置参数。

Compile(编译)菜单包括:

Compile	Alt + F9	编译
Make	F9	连接
Build		建立
Target...		选择目标文件的形式

Debug(调试)菜单包括:

Output		结果输出窗口
User screen	Alt + F5	用户(结果)输出窗口
Add watch	Ctrl + F7	加入观察变量或表达式
Watches		打开 Watch 窗口
Breakpoint	Ctrl + F8	设置断点
Call stack	Ctrl + F3	调用栈窗口,观察栈的调用情况

User screen 用于观察程序运行结果,Watches 主要用在调试程序中,观察指定变量或表达式的值。

Tools(工具)菜单包括:

Messages	F11	打开信息窗口
----------	-----	--------



Grep	Shift + F2	用户安装工具
Calculator		调用计算器
Options(设置) 菜单包括:		
Compiler...		设置缺省编译指示
Memory sizes...		设置缺省栈大小
Directories...		设置路径
Environment▶	▶	指定环境设置
Editor...		编辑器设置
Mouse...		鼠标设置
Start up...		启动设置
Window(窗口中) 菜单包括:		
Close all		关闭所有工作面窗口
Next	F6	使下一窗口为活动窗口
Previous	Shift + F6	使上一窗口为活动窗口
Close	Alt + F3	关闭活动窗口
Help(帮助) 菜单包括:		
Index	Shift + F1	帮助索引
Topic search	Ctrl + F1	主题搜索
Using help		使用帮助

第二章 Pascal 语言的基础知识

【知识要点】

1. Pascal 语言中的符号
2. 简单数据类型
3. 常用的标准函数
4. 运算符与优先级

第一节 Pascal 基本字符集、标识符与保留字

1. 基本字符集

Pascal 集的基本字符是 ASCII 码的一个子集。Pascal 基本字符包括：

- (1) 大小写英文字母：从 A 到 Z 以及从 a 到 z。
- (2) 数字：从 0 到 9。
- (3) 空格：ASCII 码序号为 32。
- (4) 其他符号如：

<	>	< =	> =	< >	+	-	*	/	()	[]
!	!	@	#	\$	^	&	;	:	.	'

2. 保留字

作为一种高级语言，Pascal 给一些英文单词赋予了特定的含义，这些特定符号叫做保留字或关键字。标准 Pascal 语言中的保留字一共有 35 个，Turbo Pascal 语言中一共有 51 个。下面是标准 Pascal 语言的保留字：

And	Array	Begin	Case	Const
Div	Do	Downto	Else	End
File	For	Function	Goto	If
In	Label	Mod	Nil	Not
Of	Or	Packed	Procedure	Program
Record	Repeat	Set	Then	To



Type	Until	Var	While	With
Free Pascal 7.0 还扩充了 16 个保留字, 分别是:				
Asm	Constructor	Destructor	Exports	Implementation
Inherited	Inline	Interface	Library	Object
Shr	String	Shl	Uses	Unit
Xor				

3. 标识符

标识符用来标记程序、类型、常数、变量、过程、函数、指令等的名字。

标识符就是以字母或下划线开头, 后面是英文字母、数字、下划线的任意组合。有效长度为 63 个字符, 并且大小写等效。

应当注意区分合法与非法的标识符。下面是一些合法的标识符:

A turbo Sit_in_the_sum _xyz R2d3_4

标识符可以分为预定义的标准标识符和自定义标识符两类。

(1) 在标准的 Pascal 中预定义了 40 个标准标识符:

3 个标准常量:

False True Maxint

5 个标准类型:

Integer Real Boolean Char Text

13 个标准过程:

Read Readln Write Writeln New Get Put
Pack Unpack Page Reset Rewrite Dispose

17 个标准函数:

Exp Ln Sqr Sqrt Pred Succ Abs
Sin Cos Arctan Round Trunc Odd Ord
Eof Eoln Chr

2 个标准文件:

Input Output

(2) 自定义标识符, 是根据用户自己的需要来命名。除了要符合上述的标识符使命名规则外, 通常还要注意如下问题:

① 自定义标识符不能与保留字同名, 否则编译出错。

② 自定义标识符必须在程序的说明部分被合理说明之后, 才能在程序的语句部分出现, 否则编译出错。

③ 不提倡将标准标识符用作自定义标识符。

比如: 假设我们定义 `exp = 7;`

则语句中出现的所有 $\exp$ 都被认为是常数 7, 这样指数函数 $\exp()$ 就无法使用了。

在 Pascal 环境中写程序代码时, 遇到与保留字匹配的字符串, 编辑器会自动将整个字符串标为白色, 提醒用户这是系统的保留字。

第二节 Pascal 的标准数据类型

Free Pascal 提供了丰富的数据类型, 这些数据类型可以分为三大类: 简单类型、构造类型和指针类型。其中简单类型可以分为标准类型(整型、实型、字符型和布尔型)和自定义类型(枚举型和子界型); 构造类型可以分为数组类型、集合类型、记录类型和文件类型。在这些数据类型中除了指针类型是动态数据类型外, 其他的都是静态数据类型。在这些数据类型中简单类型都是有序类型, 除了实型以外的简单类型都是顺序类型, 所谓顺序类型就是他们的值不仅是有序的而且是有顺序号的。

在程序的说明部分进行变量说明的时候, 不但要指明作为变量名的标识符, 而且要指出这个变量的数据类型。

在这里主要介绍实型、整型、布尔型和字符型四种常用的数据类型。

1. 实型(Real)

实型就是实数类型, 是一种最常见的数据类型。

在 Pascal 中, 实数有两种表示方法: 小数表示法和指数表示法(也称科学表示法)。

下面是小数表示法与科学表示法的对照实例:

小数表示法	科学表示法
3.47	3.47E0
123.6	1.236E2
89000.0	8.9E4
-0.00000023	-2.3E-7

在科学表示法中, “E”后面的数代表 10 的幂, 前面的数一般在 1 到 10 之间取值。实际上, Pascal 有五种实数类型(见表 2-1)。

表 2-1 Pascal 的 5 种实数类型

类型	取值范围		占字节数	有效位数
Real	$2.9e-39$	$1.7e38$	6	11 12
Single	$1.5e-45$	$3.4e38$	4	7 8
Double	$5.0e-324$	$1.7e308$	8	15 16
Extended	$3.4e-4932$	$1.1e4932$	10	19 20
comp	$-2^{63} + 1$	$2^{63} - 1$	8	19 20