



普通高等教育“十一五”国家级规划教材
(高职高专教育)

程序设计技术 (C语言)

李勤 主编
张荣新 副主编



高等教育出版社
Higher Education Press

普通高等教育“十一五”国家级规划教材

(高职高专教育)

程序设计技术(C语言)

Chengxu Sheji Jishu (C Yuyan)

李勤 主编

张荣新 副主编



高等教育出版社·北京
HIGHER EDUCATION PRESS BEIJING

内容提要

本书是普通高等教育“十一五”国家级规划教材。本书按“任务驱动”教学法的要求编写,精选编程案例,介绍了C语言编程的基本方法:首先将实际问题抽象为变量和算法,然后给出完整的C程序,最后对编程要点进行总结,并辅以针对性的拓展习题,使读者在学习C语言的同时,能够掌握常用的程序设计方法。

全书共分9章,第1章介绍程序设计基础知识,包括编译预处理和Turbo C下C程序的运行和调试方法;第2章介绍C语言基础语法和C程序顺序结构,包括数据类型、运算符与表达式;第3章介绍选择结构编程特点;第4章重点介绍循环结构;第5~8章采用任务为中心的方式,介绍常用的程序设计方法;第9章介绍文件的基本操作。本书重点介绍了数组与循环结合的编程特点,字符串处理方法,使用函数实现模块化编程的技术,应用结构体、共用体、枚举等自定义类型描述实际问题的方法,线性链表的基本操作等,包括文本文件和二进制文件的读写方法。

本书可作为培养应用性、技能型人才的计算机相关专业的教学用书,也可作为计算机培训教材以及计算机从业人员和爱好者的自学教材。

图书在版编目(CIP)数据

程序设计技术:C语言/李勤主编. —北京:高等教育出版社,2010.4

ISBN 978-7-04-029057-8

I. ①程… II. ①李… III. ①C语言-程序设计-高等学校-教材 IV. ①TP312

中国版本图书馆CIP数据核字(2010)第028468号

出版发行	高等教育出版社	购书热线	010-58581118
社 址	北京市西城区德外大街4号	咨询电话	400-810-0598
邮政编码	100120	网 址	http://www.hep.edu.cn
总 机	010-58581000		http://www.hep.com.cn
经 销	蓝色畅想图书发行有限公司	网上订购	http://www.landaco.com
印 刷	北京丰源印刷厂		http://www.landaco.com.cn
		畅想教育	http://www.widedu.com
开 本	787×1092 1/16	版 次	2010年4月第1版
印 张	14.5	印 次	2010年4月第1次印刷
字 数	350 000	定 价	21.60元

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换。

版权所有 侵权必究

物料号 29057-00

前 言

在程序设计语言中,C语言一直是许多高校程序设计课程的首选语言,但是C语言的教学效果并不理想,尤其是在高职高专学校中,有些学生只会考试而不会编写程序。为了适应高等职业院校的教学改革,我们总结了C语言教学中的经验和常见问题,决定从高职学生的认知特点出发,采用“任务驱动”的编写模式来编写适合职业教育特点的教材。

本书通过精选案例,采用提出问题、分析问题、选择数据类型和程序算法、给出源程序和总结的方式,围绕如何解决问题来讲解C语言知识,旨在培养学生的语言应用和程序设计能力。

全书共分9章,第1章介绍C语言特点和程序结构、编译预处理命令和Turbo C下C程序的运行和调试方法。之所以将编译预处理命令放到第1章而不是在后面单独介绍,是因为在以后各章可以运用编译预处理命令,能够通过多次重复使学生熟练掌握,并培养良好的编程习惯。

第2章介绍C语言语法和顺序结构的程序设计。主要介绍数据类型、运算符与表达式,要求能够用C语言表达式描述简单的问题。

第3章介绍选择结构编程特点。通过“个人所得税计算程序”介绍了运用条件语句实现选择程序的方法,通过“奖金计算程序”介绍了用switch语句实现多分支结构的方法,同时分析了if语句和switch语句的应用特点。

第4章介绍循环程序设计方法,通过案例介绍了实现循环结构的三种语句,并介绍了“枚举法”、“递推法”等常用编程方法。

第5章介绍数组与循环结构结合的程序设计方法,重点介绍一维数组和二维数组的编程应用。通过“学生成绩统计”介绍了二维数组与三维数组的应用,并介绍使用数组处理字符串的方法。

第6章重点介绍自定义函数的编写、声明和调用方法,并介绍递归函数设计方法。通过大量的案例使读者能够运用函数来简化程序设计,并初步掌握递归程序的设计方法。

第7章介绍指针的概念和编程方法,重点介绍运用指针处理数组元素和字符串的方法,以及函数指针等内容。

第8章介绍结构体、共用体、枚举等自定义类型,通过“班级通讯录程序”案例使学生了解复杂程序的开发过程,包括从建立合理的数据结构开始到最终代码实现的全过程,学习用自定义类型描述实际问题的方法。本章还介绍了指针与结构体结合实现线性链表结构的方法。

第9章介绍文件的基本操作,包括文本文件和二进制文件的读/写方法。用文件知识重新编写了“学生成绩统计程序”,使学生初步掌握一个实用程序的编写方法。

全书内容丰富,深入浅出,系统性和应用性强,融入了编者多年的教学和实践经验。书中配备了大量的案例,叙述详细,通俗易懂,便于自学,并在每个综合案例后安排了具有相似内容的“练一练”。为了便于读者检验自己的学习效果,每章都附有大量的习题。通过本书的学习,读者不仅能掌握C语言的语法,同时也能掌握程序设计的方法和技巧。

II 前言

本书第2章由临汾职业技术学院李彩玲编写,第3章由天津职业大学张荣新编写,其余各章由天津职业大学李勤编写并负责统稿。

在本书的编写过程中,得到了同事陈洁和谢莉莉的帮助,他们对书后习题提出宝贵意见,在此一并表示衷心感谢。

书中的程序都经过了调试和运行,但是由于时间仓促和作者水平有限,难免存在一些错误和不足。希望广大读者予以指正。

作者

2010年1月

liqin66@126.com

郑重声明

高等教育出版社依法对本书享有专有出版权。任何未经许可的复制、销售行为均违反《中华人民共和国著作权法》，其行为人将承担相应的民事责任和行政责任，构成犯罪的，将被依法追究刑事责任。为了维护市场秩序，保护读者的合法权益，避免读者误用盗版书造成不良后果，我社将配合行政执法部门和司法机关对违法犯罪的单位和个人给予严厉打击。社会各界人士如发现上述侵权行为，希望及时举报，本社将奖励举报有功人员。

反盗版举报电话：(010)58581897/58581896/58581879

反盗版举报传真：(010)82086060

E-mail: dd@hep.com.cn

通信地址：北京市西城区德外大街4号

高等教育出版社打击盗版办公室

邮编：100120

购书请拨打电话：(010)58581118

策划编辑	杜冰
责任编辑	俞丽莎
封面设计	张雨微
版式设计	王莹
责任校对	俞声佳
责任印制	张泽业

目 录

第 1 章 C 语言编程基础

1.1 程序与高级语言	1	1.3.2 修改语法错误	8
1.1.1 程序与高级语言概念	1	1.3.3 排除链接错误	11
1.1.2 C 语言的特点	2	1.3.4 程序的调试	11
1.1.3 C 程序的结构	2	1.3.5 终止程序执行	14
1.2 算法设计与描述	5	1.4 编译预处理	14
1.2.1 算法设计	5	1.4.1 宏定义命令#define	15
1.2.2 算法描述	6	1.4.2 文件包含命令#include	16
1.3 C 程序的运行与调试	7	习题一	18
1.3.1 C 程序的运行过程	8		

第 2 章 顺序结构程序设计

2.1 数据类型	19	2.5.2 赋值运算	36
2.1.1 基本数据类型	20	2.5.3 关系运算符	38
2.1.2 类型修饰符	20	2.5.4 逻辑运算符	39
2.2 定义变量	22	2.5.5 位运算	40
2.3 定义常量	22	2.5.6 条件运算符	41
2.4 数据的输入输出	24	2.5.7 逗号操作符	42
2.4.1 格式化输出函数	24	2.5.8 sizeof()运算符	42
2.4.2 格式化输入函数	29	2.6 表达式中的类型转换	42
2.4.3 字符输入输出	31	2.6.1 自动类型转换	42
2.4.4 字符串输入输出	32	2.6.2 强制类型转换	43
2.5 运算符与表达式	34	习题二	44
2.5.1 算术运算符	34		

第 3 章 分支程序设计

3.1 程序的三种基本结构	47	3.3.1 奖金计算程序	53
3.2 使用条件语句编程	48	3.3.2 switch 语句要点	55
3.2.1 成绩判定程序	48	3.4 分支结构编程实例	56
3.2.2 个人所得税计算程序	49	习题三	58
3.3 多分支结构	52		

第4章 循环程序设计

4.1 应用 for 语句编程	61	4.6 循环程序设计实例	70
4.2 应用 while 语句编程	64	4.6.1 枚举法	70
4.3 应用 do-while 语句编程	65	4.6.2 递推法	73
4.4 跳转语句	67	4.6.3 求最大公约数	77
4.5 循环嵌套	69	习题四	78

第5章 数组与循环

5.1 一维数组及应用	82	5.3.1 多维数组定义	88
5.1.1 使用一维数组排序	82	5.3.2 三维数组编程实例	88
5.1.2 一维数组总结	84	5.4 数组编程示例	90
5.2 二维数组	85	5.4.1 筛选法求素数	90
5.2.1 二维数组定义	85	5.4.2 使用数组处理字符串	92
5.2.2 二维数组的初始化	85	5.4.3 矩阵运算	96
5.2.3 用二维数组编程	86	习题五	98
5.3 多维数组	88		

第6章 函 数

6.1 自定义函数示例	103	6.4.1 统计字符出现频率	115
6.2 自定义函数详解	105	6.4.2 班级成绩统计程序	117
6.2.1 函数定义	105	6.5 递归函数	120
6.2.2 函数调用	106	6.5.1 汉诺塔问题	120
6.2.3 函数声明	108	6.5.2 八皇后问题	122
6.2.4 参数声明形式	109	6.5.3 递归函数总结	125
6.3 变量的作用域和生存期	110	6.6 运行多文件程序	126
6.3.1 变量的作用域	110	习题六	128
6.3.2 变量的存储类别	112		
6.4 使用函数编程	115		

第7章 指 针

7.1 指针的定义及操作	135	7.3.2 用指针数组对字符串排序	145
7.1.1 指针的含义	136	7.3.3 用指针返回函数处理结果	148
7.1.2 指针变量的定义	136	7.4 main 函数中的参数	149
7.1.3 指针的基本操作	137	7.5 指向函数的指针	151
7.2 用指针处理数组元素	140	7.5.1 函数指针定义	151
7.2.1 用指针处理一维数组	140	7.5.2 函数指针的赋值	152
7.2.2 用指针处理二维数组	141	7.6 多级指针	154
7.3 用指针编程	142	习题七	156
7.3.1 用指针处理字符串	142		

第 8 章 自定义类型

8.1 自定义类型简述	162	8.3 共用体	184
8.2 结构体及应用	163	8.4 枚举类型	191
8.2.1 定义结构体	163	8.4.1 枚举定义	192
8.2.2 结构体变量的操作	166	8.4.2 枚举元素的值	192
8.2.3 使用结构数组编程	168	8.4.3 枚举变量的操作	193
8.2.4 指向结构的指针	175	习题八	194
8.2.5 位域及应用	181		

第 9 章 文件操作

9.1 文件系统操作基础	198	9.2.3 字符串形式的输入输出	205
9.1.1 文件的概念	198	9.2.4 读/写二进制文件	207
9.1.2 文件类型指针	199	9.2.5 文件的格式化输入输出	209
9.1.3 文件系统函数	199	9.3 文件的定位与检测	209
9.2 文件操作	200	9.3.1 文件定位	209
9.2.1 打开与关闭文件	200	9.3.2 文件的出错检测	211
9.2.2 文件的字符输入输出	203	习题九	212

附录 1 常用字符 ASCII 码表	215
--------------------------	-----

附录 2 C 语言常用库函数	216
----------------------	-----

参考文献	221
------------	-----

第 1 章

C 语言编程基础



学习目标

会用 N-S 图描述简单算法

能够在 Turbo C 下运行、调试给定的 C 程序



知识点

- 了解源程序和目标程序的概念
- 了解 C 程序的一般结构
- 理解编译预处理命令的作用
- 理解语法错误与逻辑错误



技能点

- 会用动态跟踪方法调试程序
- 会识别语法错误和链接错误
- 会使用#include 和#define 命令
- 熟练运行给定的 C 程序



重点和难点

- #include 和#define 命令
- 源程序的编译、运行和调试

1.1 程序与高级语言

1.1.1 程序与高级语言概念

在日常生活、工作中如果要完成一项任务,通常都需要通过一系列步骤来实现,例如,一项会议的日程安排、体育比赛的赛程等,这些步骤可以用人类自然语言描述。如世界杯足球比赛的赛程为:

- (1) 各大洲进行预选赛。
- (2) 抽签分组。

- (3) 按小组进行预赛,取前2名。
- (4) 1/8 决赛。
- (5) 1/4 决赛。
- (6) 半决赛。
- (7) 决赛。

这些按一定顺序安排的步骤序列就是广义的程序概念。

计算机程序是用程序设计语言来描述的解决问题的步骤与算法,通过特定的开发环境计算机才能够执行。

程序设计语言作为一种语言工具,是人们为了便于记忆和表达算法而设计的,可分为低级语言和高级语言两大类。低级语言指与机器硬件相关的语言,如汇编语言;而高级语言是与机器硬件无关的语言,它更符合人的思维习惯和英语语法,能够描述复杂算法。根据目标程序的执行特点,高级语言又分为“解释型”和“编译型”两种。所谓解释型就像同声翻译,对于程序中的语句边翻译边执行,直到遇到错误才停止;而编译型是将程序的所有语句全部翻译成计算机可以识别的语言,能够形成可脱离源程序环境的可执行程序。

1.1.2 C语言的特点

C语言是一种应用广泛的编译型高级程序设计语言,具有丰富的数据类型和流程控制语句,可以编写复杂的程序,如UNIX操作系统的大部分代码都是用C语言编写的。C语言具有灵活、高效的特点,被称为面向程序员的语言。与一般高级语言相比,C语言具有以下特点:

① C语言既有高级语言的特性,又具有汇编语言的灵活性,非常适合编写系统程序。例如C语言提供指针类型和位运算,可以直接对位、字节和内存地址单元进行操作,运用C语言能够编写简洁、高效的程序。

② C语言是“弱类型”语言,不像其他高级语言如Pascal那样严格进行类型匹配,允许除指针外大部分基本数据类型的转换,例如char型、int型、float型等数据可以混合出现在同一表达式中,并自动进行类型的转换。

③ C语言编译程序不对数组的边界进行检查。在使用数组时程序员必须自己考虑数组下标的越界问题。因此C语言被认为是面向程序员(即专业编程者)的语言,语法虽然更宽松,但比较容易出错。

④ C语言中定义变量时不仅需要声明变量的数据类型,还要指定变量的“存储类别”来限定变量的存储位置,以实现某些特殊需要。

⑤ C语言中没有定义输入输出语句,而是采用库函数来实现,因此便于在不同系统间移植。

以上特点读者在后面各章的学习中可以慢慢体会。

1.1.3 C程序的结构

首先了解以下C程序的结构。C程序由函数组成,一个完整的C程序必须包括一个主函数main(),并可以有多个子函数。

下面通过两个例子来了解C程序的组成和结构。

程序1-1:最简单的C程序。

```

1  /***** 程序 1-1 最简单的 C 程序 *****/
2  main()
3  {
4      printf("this is a simple C program !");
5  }

```

各程序行的作用为：

第 1 行为注释行。由“/*”和“*/”括起的部分称为注释,用于对程序功能做一些说明,以增加程序的可读性,使程序便于维护。注释在编译过程中将被忽略,不会生成任何目标代码。

第 2~5 行为主函数,也称为主程序。其中第 2 行的 main()为关键字,它表示程序开始执行的位置,这一行是必需的;第 3~5 行为函数体,花括号{}标示了函数的开始和结束,也是必需的。

程序 1-2: 只有主函数的 C 程序。

```

1  /***** 程序 1-2 只有主函数的 C 程序 *****/
2  /* 程序名称: cx1-2.c */
3  /* 功能: 统计键盘输入字符的个数 */
4  # define maxlen 255 /* 可统计的字符串最大长度 */
5  # include <stdio.h>
6  main()
7  {
8      int num=0;
9      char str[maxlen+1];
10     printf("\n please input a string(<255)\n");
11     gets(str);
12     while (str[num++]) ;
13     printf("number of char=%d",num-1);
14 }

```

程序 1-2 中只有一个 main()函数,本书为了便于叙述,为各行加上行号。但行号在实际源程序中是不存在的。

第 1~3 行为注释行。

第 4~5 行由“#”开始的行称为编译预处理命令,根据程序需要决定,在语法上也不是必需的。

第 6~14 行为主函数,功能是读入一个字符串并统计其字符个数。其中各行具体作用为:第 6 行的 main()为关键字,表示程序开始执行的位置,这一行是必需的;第 8、9 行为变量定义语句,任务是使系统知道程序所需的存储空间;第 7、14 行的花括号标示了函数的开始和结束,也是必需的;第 10 行是系统输出函数,用于在屏幕上显示一行字符“please input a string(<255)”；第 11 行是系统输入函数,用于获取用户输入的字符串;第 12 行是一条循环语句,以统计字符个数;第 13 行输出统计结果。

程序 1-1 和程序 1-2 是比较简单的 C 程序,在书写上采用每条语句占一行的方式。实际上 C 程序的书写是十分自由的,允许一行书写多条语句,也允许一条语句跨多行。C 语言用“;”号标志语句的结束,标识符之间用空格分隔,且一个空格和多空格的作用是相同的。

一个简单的 C 程序只有主函数,甚至可以没有任何语句。

例如:main(){}在语法上也是一个正确的 C 程序,只是没有任何意义。

下面是一个比较复杂的 C 程序,从中可了解 C 程序的最一般结构。

程序 1-3: 具有两个子函数的 C 程序。

```
/* ***** 程序 1-3 具有两个子函数的 C 程序 ***** */
1/* 程序名称 : cx1-3.c */
2/* 功能 : 将输入的 n 个整数排序并输出 */
3#define maxnum 100 /* 定义排序的最大个数 */
4 void sort(int a[],int n); /* 声明子函数 sort */
5/* 函数 sort 对数组 a 中的 n 个整数 a[0] ~ a[n-1] 按升序排序 */
6 void prt_data(int a[],int n); /* 声明子函数 prt_data */
7/* 输出数组 a 中的 n 个整数 a[0] ~ a[n-1] */
8 main() /* 主函数 */
9 {
10 int num,a[ maxnum ];
11 printf(" \nplease input number of integer(<=100):");
12 scanf("%d",&num); /* 读入待排序的整数个数 */
13 if (num>100)
14 printf("input error!,the number is >100");
15 else
16 {
17 printf(" \nplease input %d integer;",num);
18 /* 读入 num 个整数,并存放到数组 a 中 */
19 for (i=0; i<num; i++)
20 scanf("%d",&a[i]);
21 sort(a,num); /* 调用函数 sort 进行排序 */
22 printf("the result is: \n"); /* 输出排序结果 */
23 prt_data(a,num);
24 } /* if */
25 }
26 /* 定义输出函数,每行最多显示 10 个数据 */
27 void prt_data(int a[],int n)
28 { int i;
29 for (i=0; i<num; i++)
```

```

30  { printf("%d\t",a[i]);
31  if ((i+1) % 10 == 0) printf("\n");/* 换行 */
32  }
33 }
34 /* 定义排序函数,将数组 a 中的 n 个整数按升序排序 */
35 void sort(int a[],int n)
36 {
37     int i,j,tmp;
38     for (i=0; i<n; i++)
39         for (j=i+1; j<n-1; j++)
40             if (a[i]<a[j])          /* 交换数据 */
41                 { tmp=a[i];
42                   a[i]=a[j];
43                   a[j]=tmp;
44                 }
45 }

```

程序 1-3 组成与结构分析:

第 1~2 行:程序注释。

第 3 行:编译预处理命令行。

第 4~7 行:声明两个子函数,并使用注释对函数的功能进行说明。

第 8~25 行:定义主函数。

第 26~32 行:定义 prt_data() 函数。

第 34~45 行:定义 sort() 函数。

在程序 1-3 中,将 main() 函数放在了所有子函数定义的前面,这是为了方便阅读 main() 函数,尽快理解程序的功能。实际上主函数的位置是任意的,可以放在程序的任何位置,但是必须保证主函数中所调用的函数已经在前面作过定义或声明,因此本程序在主函数前增加了函数声明语句。

1.2 算法设计与描述

计算机程序设计的要点是对实际问题进行抽象:首先用数据描述问题,其次用算法描述处理过程,然后用具体的程序设计语言进行表达就成为程序。

1.2.1 算法设计

为了解决一个实际问题而采用的方法和步骤就称为算法。在编写程序前,首先要有一个算法设计的概念,有了正确的逻辑思路,然后再考虑用程序语言实现它,而不是直接写语句。

任务1-1: 一个班有40个学生,期末考试有3门课程,要求从中找出平均成绩最高的前3个学生名单。

算法思路1:

- ① 首先通过循环计算每个学生的平均成绩。
- ② 然后按平均成绩从高到低对学生进行排序。
- ③ 输出前三个学生名单。

算法思路2:

- ① 首先通过循环计算每个学生的平均成绩。
- ② 找出全班平均成绩最高的学生并输出。
- ③ 找出全班平均成绩第2名的学生并输出。
- ④ 找出全班平均成绩第3名的学生并输出。

但是,从以上用自然语言描述的算法到计算机程序,还有很大的距离,这是因为缺少对问题的抽象——用程序设计语言所提供的数据结构来描述问题。实际上,用合适的数据(变量)表达实际问题中的每个要素,再用程序语句来描述上面的算法,也就是程序设计的过程。

1.2.2 算法描述

描述算法的工具有很多,可以用自然语言、伪代码;也可以用流程图等方法。一方面要使算法便于理解,另一方面容易转化为程序,不要有二义性。自然语言虽然简单、易懂,但是有时语义不明确,如“或”的含义有时是“可兼或”,有时是“排斥或”。因此常用流程图或N-S图来描述算法。

1. 用流程图描述算法

算法思路1——求班级平均成绩前三名:

设三门课的成绩用A,B,C表示,平均成绩为G,则第*i*个学生的成绩为 A_i, B_i, C_i ;该算法描述的流程图如图1-1所示。

在图1-1所示的流程图中,“对平均成绩G由高到低排序”这个步骤还比较抽象,需要进一步细化。这也是程序设计中的常用方法“自顶向下,逐步求精”,可以将复杂的问题分而置之,降低难度。

用流程图表示算法灵活、直观、使用范围广,它允许用流程线使流程任意转移,但这却是程序设计的一个隐患,如果程序中的转换过多,会使程序难以维护。因此需要采用结构化程序来限制这种无规律的任意转换,要求用顺序、选择、循环三种基本结构做为构成程序的基本单位,这样就规范了流程线的使用。

同步练习:画出“对平均成绩G由高到低排序”的流程图。

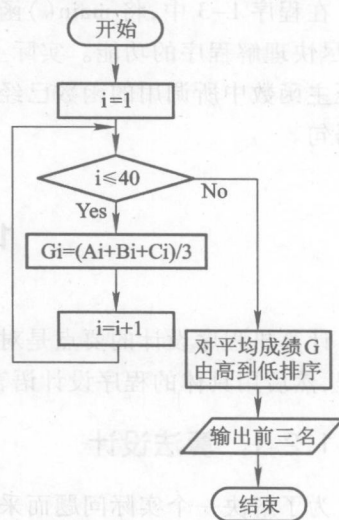


图1-1 算法流程图1

2. 用 N-S 图描述算法

针对流程图存在的问题,两位美国学者提出了一种无流程线的图法,只用三种基本结构描述算法,称为 N-S 图。如图 1-2 中的 A、B 是操作块,可以是输入、输出、赋值等基本操作,也可以是三种基本结构。

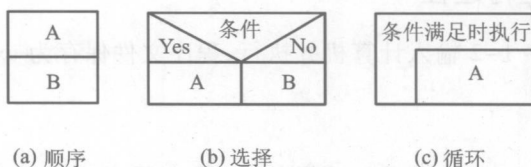


图 1-2 N-S 图的三种基本结构

N-S 图完全取消了带箭头的流程线,全部算法描述在一个大矩形框中,在这个大矩形框中又包含许多小的隶属矩形框。

任务 1-2: 用 N-S 图描述“对平均成绩 G 由高到低进行排序”的算法。

算法思路:

第 1 次是从 40 人中找到最高分,使之成为第 1 名。方法是用第 2 ~ 40 人的成绩依次与第 1 人比较,如果高于目标位置(第 1)就与之交换。

第 2 次是从剩下的 39 人中找到最高分放到第 2 个位置,方法是用第 3 ~ 40 人的成绩依次与第 2 人比较,如果高于目标位置(第 2)就与之交换……

不失一般性,第 i 次是用第 $i+1$ ~ 40 人的成绩依次与第 i 人比较,如果高于目标位置就与之交换,因此共需要 39 次选择,剩下的一个为成绩最低的。

以上算法用 N-S 图描述如图 1-3 所示。

思考:如何将任务 1-1 的算法思路用 N-S 图描述?

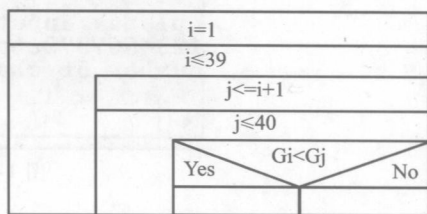


图 1-3 用 N-S 图描述的排序算法

1.3 C 程序的运行与调试

C 语言是一种“编译型”高级语言,源程序的执行需要经过“编辑”—“编译”—“链接”—“执行”的过程。“编辑”即建立源程序的过程,可以使用任何文本编辑程序完成。“编译”是将源程序翻译为目标程序的过程,需要通过特定的编译程序。“链接”则是将目标程序与系统库函数等装配为可执行程序的过程。

Turbo C 2.0(简称为 TC)不仅是优秀的 C 编译程序,而且是一个集成开发环境,在 Turbo C 2.0 下可以完成 C 程序由“编辑”到“执行”的全过程。

启动 TC 的方法是,找到可执行程序 TC,用鼠标双击该文件,或从右键菜单选择“打开”。

启动菜单命令的方法是用“Alt+菜单首字母键”激活菜单栏,例如用 Alt+F 键可激活“File”菜单,用 Alt+R 键可激活“Run”菜单,使用光标键→、←可移动到不同的菜单栏,再使用光标键↑或↓选择所要的菜单项,然后按回车键即可。

Turbo C 的菜单命令比较容易理解,下面结合实例予以介绍。

1.3.1 C 程序的运行过程

任务 1-3: 将示例程序 1-2 输入计算机并执行,程序文件保存为 cx1-1.c。

操作的关键步骤如下:

第一步:启动 TC。

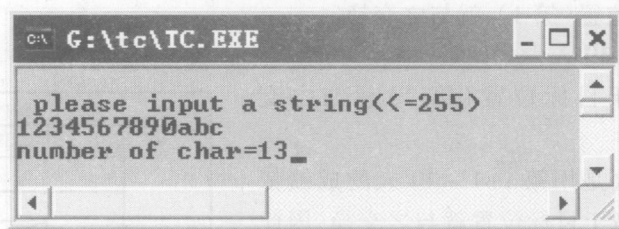
第二步:通过菜单“file”——“new”进入编辑状态,输入程序 1-2 中的代码后,执行菜单命令“File”→“Save”或按 F2 键保存,在弹出的保存对话框中输入“cx1-1”。文件将默认保存在 Turbo C 目录下,并自动为源文件添加扩展名.c,这是 C 语言源程序默认的扩展名。

第三步:按 Ctrl+F9 键,系统将自动启动程序 cx1-1 的编译、链接及执行过程。

编译程序的工作步骤是:先对 cx1-1 进行编译,如果没有语法错误,则生成目标文件 cx1-1.obj,然后继续进行链接过程,如果链接成功则生成可执行程序 cx1-1.exe,并自动执行该程序。

如果程序中需要输入数据,则等待用户输入,执行完毕返回编辑界面。

第四步:按 Alt+F5 键查看程序执行结果(如图 1-4 所示)。按任意键返回编辑界面。



```

G:\tc\TC.EXE
please input a string(<=255)
1234567890abc
number of char=13_
  
```

图 1-4 程序 1-2 执行结果



练一练: 在 Turbo C 下输入并执行程序 1-3。

以上是执行一个源程序时的理想情况。但是,即使是经验丰富的程序员,也难免会发生错误,例如源程序输入时产生的拼写错误常常难以避免,语法错误在编译阶段就可发现。那么如何改正源程序中的错误呢?下面仍将通过程序 1-2 来说明。

1.3.2 修改语法错误

如果源程序书写不符合 C 语言的语法规则,编译程序就不能将其翻译成目标指令,因此产生的错误称为语法错误。编译过程产生错误信息是很正常的,这也是编译程序的作用之一,可以帮助程序员发现程序中的错误,并快速定位到引起错误的语句行,以节省修改错误的时间。

如果仅编译(暂时不运行)程序,可通过“Compile”菜单来实现,如图 1-5 所示。其中:“Compile to OBJ”命令用于编译源程序,并生成目标文件。“Make EXE file”命令用于编译源程序并链接生成可执行文件。对该程序执行“Make EXE file”命令后结果如图 1-6 所示。