



普通高等教育“十三五”规划教材

田兆福 矫庆军 陈志双 主编

C语言程序设计 基础教程

C YUYAN CHENGXU SHEJI
JICHU JIAOCHENG



国防工业出版社

National Defense Industry Press

普通高等教育“十三五”规划教材

C 语言程序设计基础教程

田兆福 矫庆军 陈志双 主编

国防工业出版社

· 北京 ·

内 容 简 介

本书针对非计算机专业学生的专业知识要求和学生知识结构的构成,主要包括计算机程序设计概述、C 语言基础、输入输出与顺序结构程序设计、选择结构程序设计、循环结构程序设计、数组、指针、函数、结构体与共用体、文件操作等。书中涵盖了计算机二级考试的全部内容和题型。

本书内容循序渐进,注重启发,可作为本科院校非计算机专业学生的 C 语言教材,也可供 C 语言初学者参考。

图书在版编目(CIP)数据

C 语言程序设计基础教程 / 田兆福, 矫庆军, 陈志双
主编. —北京: 国防工业出版社, 2016. 9
普通高等教育“十三五”规划教材
ISBN 978 - 7 - 118 - 11036 - 4

I. ①C… II. ①田… ②矫… ③陈… III. ①C 语言 -
程序设计 - 高等学校 - 教材 IV. ①TP312.8

中国版本图书馆 CIP 数据核字(2016)第 251105 号

※

国防工业出版社 出版发行

(北京市海淀区紫竹院南路 23 号 邮政编码 100048)

三河市德鑫印刷有限公司印刷

新华书店经售

*

开本 787 × 1092 1/16 印张 10 字数 222 千字

2016 年 9 月第 1 版第 1 次印刷 印数 1—3000 册 定价 29.00 元

(本书如有印装错误,我社负责调换)

国防书店:(010)88540777

发行邮购:(010)88540776

发行传真:(010)88540755

发行业务:(010)88540717

C 语言以其使用灵活方便、功能强大被计算机程序设计者所喜爱,也适合计算机初学者掌握程序设计的基本方法和技巧。本教材针对非计算机专业学生的专业知识要求和学生知识结构的构成,内容编排上尽量符合初学者的认知能力,注重学生分析问题和解决问题能力的提高,教材内容难易合适,配上难易合适的习题,全面提升学生程序设计技巧和能力,力求为学生今后掌握计算机知识,利用计算机解决问题打下良好的基础。

我们编写教材力求克服教材难度偏大、习题量偏少且习题难度较大、不能有效辅助学生对知识的掌握和消化的问题,力争做到以全面提高学生的计算机设计能力为宗旨,以培养学生的创新精神和实践能力为重点,强调基础知识和基本技能的掌握,实例丰富,例题举一反三,注重分析和设计思路,提高学生程序设计的技巧和质量。重点题型提供了启发性的思考题,每章有适合学生掌握和练习的习题,使学生能掌握程序设计语言的基本语法和提高程序设计能力,让教材适合非计算机专业学生的学习。

本书由田兆福、矫庆军、陈志双担任主编,参加本书编写的还有孟凡波、彭武良、徐皓、吴巍巍、任彦华、侯献军、周启微。本书在编写过程中参考了部分同类图书,在此向书中引用文献的作者表示感谢。

由于作者水平有限,加之时间仓促,书中不妥之处在所难免,敬请广大读者批评指正。

2016 年 3 月

编者

第1章 计算机程序设计概述

1.1 计算机程序语言	1
1.2 C语言的特点	1
1.3 简单的C程序介绍	2
1.4 C程序的结构	4
1.5 C程序运行	4
1.6 Turbo C 2.0 集成开发环境的使用	4
1.6.1 Turbo C 2.0 简介和启动	4
1.6.2 Turbo C 2.0 的操作步骤	5
1.7 程序设计的算法简介	8
1.8 C语言的学习方法	10
习题1	10

第2章 C语言基础

2.1 标识符	12
2.2 数据类型	12
2.3 变量与常量	13
2.3.1 整型变量	13
2.3.2 整型常量	13
2.3.3 实型变量	14
2.3.4 实型常量	14
2.3.5 字符型变量与常量	14
2.3.6 字符串常量	15
2.3.7 符号常量	15
2.4 变量的赋值	15
2.5 运算符与表达式	16
2.5.1 算术运算符和算术表达式	16
2.5.2 强制类型转换运算符	16
2.5.3 自增、自减运算符	17
2.5.4 复合赋值运算符	17
2.5.5 逗号表达式	18

2.5.6 求字节运算符:sizeof	18
2.5.7 位运算符	18
习题2	21

第3章 输入输出与顺序结构程序设计

3.1 数据的输出	23
3.1.1 字符输出函数 putchar()	24
3.1.2 printf()函数	24
3.2 数据的输入	27
3.2.1 字符输入函数 getchar()	27
3.2.2 scanf()函数	27
习题3	29

第4章 选择结构程序设计

4.1 关系运算和逻辑运算	31
4.1.1 关系运算符和关系表达式	31
4.1.2 逻辑运算符与逻辑表达式	32
4.2 if 语句	33
4.2.1 单分支 if 语句	33
4.2.2 双分支 if 语句	35
4.2.3 多分支嵌套 if 语句	36
4.3 条件运算符和条件表达式	39
4.4 switch 语句	39
4.5 选择结构应用实例	41
习题4	45

第5章 循环结构程序设计

5.1 while 语句	49
5.2 do-while 语句	51
5.3 for 语句	52
5.4 循环嵌套——多重循环结构	54
5.5 break 语句和 continue 语句	56
5.5.1 break 语句	56
5.5.2 continue 语句	57
5.6 循环结构应用实例	57
习题5	63

第6章 数组

6.1 一维数组	69
----------------	----

6.1.1	一维数组的定义	69
6.1.2	一维数组的初始化	71
6.1.3	一维数组应用举例	72
6.2	二维数组	75
6.2.1	二维数组的定义	75
6.2.2	二维数组的初始化	76
6.3	字符数组	77
6.3.1	字符数组的定义	77
6.3.2	将字符串赋值给字符数组	77
6.3.3	字符串函数	78
6.4	字符串处理函数	80
6.4.1	字符串连接函数 strcat()	80
6.4.2	字符串复制函数 strcpy()	80
6.4.3	字符串比较函数 strcmp()	80
习题 6		81

第7章 指针

7.1	指针变量的运算符	85
7.2	指针的类型	86
7.3	指针与数组	87
7.3.1	一维数组与指针	87
7.3.2	二维数组与指针	88
7.4	字符型指针变量与字符串	89
7.4.1	字符型指针变量的定义和赋值	89
7.4.2	字符指针变量与字符数组的区别	90
7.5	主函数参数	91
7.6	指针编程实例	92
习题 7		94

第8章 函数

8.1	函数定义	99
8.2	形参与实参	100
8.3	函数调用	101
8.4	一维数组与函数参数	104
8.4.1	数组名作实参	104
8.4.2	数组元素作实参	105
8.5	变量的作用域和存储类别	106
8.5.1	变量的作用域	106
8.5.2	变量的存储类别	108

习题8	114
-----------	-----

第9章 结构体和共用体

9.1 结构体类型和结构体变量	119
9.1.1 引用结构体变量	121
9.1.2 结构体变量的初始化	122
9.2 结构体数组	123
9.2.1 定义结构体数组	123
9.2.2 结构体数组的初始化	124
9.3 结构体指针	124
9.3.1 指向结构体变量的指针变量	124
9.3.2 指向结构体数组的指针	125
9.4 共用体类型	126
9.4.1 共用体类型	126
9.4.2 共用体变量的引用方式	127
9.4.3 共用体类型数据的特点	127
习题9	127

第10章 文件操作

10.1 C语言文件系统概述	130
10.2 文件类型指针	131
10.3 文件的打开与关闭	132
10.3.1 文件的打开	132
10.3.2 文件的关闭	134
10.4 文件的读写	134
10.4.1 fputc()函数和 fgetc()函数	134
10.4.2 fread()函数和 fwrite()函数	136
10.4.3 fprintf()函数和 fscanf()函数	136
10.4.4 判断文件结束函数 feof()	137
10.4.5 fgets()函数和 fputs()函数	137
10.5 文件定位	137
10.5.1 fseek()函数	138
10.5.2 ftell()函数	138
10.5.3 rewind()函数	138
10.6 出错检测	138
10.6.1 ferror()函数	139
10.6.2 clearerr()函数	139
习题10	142

附录 A 常用字符与 ASCII 码对照表 144

附录 B C 常用库函数 145

附录 C 运算符与优先级 148

参考文献 150

第 1 章

计算机程序设计概述

本章介绍 C 语言的基本结构和算法,主要内容有:函数结构、算法描述、C 程序上机调试与运行。

1.1 计算机程序语言

什么是计算机程序呢?就是遵循一定规则并能完成指定工作的一系列指令和数据的集合。采用计算机语言对程序进行编写,以使计算机解决问题的整个处理过程就称为程序设计。这段指令必须按着一定的规则书写,否则计算机是不能执行的,提供这种规则和运行环境的软件也叫作计算机程序设计语言。针对不同的需要有不同的计算机程序设计语言,有针对网络的语言,有针对数据库的语言。C 语言是众多语言中应用极为广泛的一种计算机程序设计语言,在 20 世纪 70 年代初问世,美国国家标准协会于 1983 年发表了一个 C 语言标准,通常称为 ANSI C。

早期的 C 语言主要是用于 UNIX 系统,由于 C 语言的强大功能和各方面的优点逐渐为人们认识,它成为当代最优秀的程序设计语言之一。使用 C 语言不仅可以编写应用软件,也可以编写系统软件。C 语言的主要应用除了设计操作系统,还有计算机通信的控制,计算机病毒和木马程序的编写,计算机杀毒软件和防火墙设计等领域,是黑客和反黑客的必杀技。

许多软件公司推出了自己的 C 语言,目前最流行的 C 语言有以下几种:

- Microsoft C 或称 MS C
- Borland Turbo C 或称 Turbo C

这些 C 语言版本不仅实现了 ANSI C 标准,而且在此基础上各自做了一些扩充。本教材主要讲解 ANSI C 的基本规则。

1.2 C 语言的特点

(1) C 语言使用方便、灵活。

- (2) 运算符丰富。共有 34 种。可以实现其他高级语言难以实现的运算。
- (3) 语法限制不太严格,程序设计自由度大。
- (4) C 语言提供了对位、字节和地址的操作,使得程序可以直接对内存进行访问,可以直接对内存进行操作。
- (5) 生成目标代码质量高,程序执行效率高。

1.3 简单的 C 程序介绍

为了说明 C 语言源程序结构的特点,先看以下几个程序,可从这些例子中了解一下 C 语言的基本规则、基本部分和书写格式。

【例 1.1】

```
#include <stdio.h>
main()
{
    int a,b,c;
    scanf("%d,%d",&a,&b);
    c=20+a*b;
    printf("%d\n",c);
}
```

说明:

- ① 这样有一定格式要求的一段程序,称作函数。函数体由花括号{}括起来。
- ② main 是函数名,表示这是一个主函数。
- ③ 每一个 C 源程序都必须有,且只能有一个主函数。
- ④ int 是说明 a,b,c 三个变量都是整型的,应该放整型数。
- ⑤ scanf 函数的功能是键盘输入两个整数给 a 和 b,%d 表示输入的内容按整型输入。
- ⑥ printf 函数的功能是把要输出的内容送到显示器去显示。%d 表示输出的内容按整型输出,\n 表示换行。

⑦ #include <stdio.h> 是说明 scanf 和 printf 两个标准函数是在 stdio.h 文件中,不用编写,直接使用就行。include 称为文件包含命令,被包含的文件通常是由系统提供的,其扩展名为.h,因此也称为头文件。C 语言的头文件中包括了各个标准函数,凡是在程序中调用一个库函数时,都必须包含该函数所在的头文件。C 语言规定,对 scanf 和 printf 这两个函数可以省去对其头文件的包含命令。

【例 1.2】

```
#include <stdio.h>
float ti(float r);
main()
{
    float r,s;
    printf("input a number:\n");
    scanf("%f",&r);
```

```

    ti(r);
}

float ti(float r)
{
    float s;
    s=3.14*r*r;
    printf("area is %f \n",s);
}

```

说明:

- ① float r,s 是说明 r,s 这两个变量是实型的,应该存放实数。
- ② C 语言规定,源程序中所有用到的变量都必须先说明类型,后使用,否则将会出错。

- ③ 一个 C 程序可以由几个函数构成。每个函数完成一个特定的功能。

【例 1.3】用户输入两个整数,程序执行后输出其中较大的数。

```

#include <stdio.h>

void main( )
{
    int x,y;
    scanf("%d,%d",&x,&y);
    if(x > y)
        printf("最大值为:%d",x);
    else
        printf("最大值为:%d",y);
}

```

【例 1.4】用户输入两个整数,程序执行后输出其中较大的数,通过函数实现。

```

#include <stdio.h>

int max(int a,int b);

main()
{
    int x,y,z;
    printf("input two numbers:\n");
    scanf("%d,%d",&x,&y);
    z=max(x,y);
    printf("maxmum=%d",z);
}

int max(int a,int b)
{
    if(a>b) return a;else return b;
}

```

1.4 C 程序的结构

从上面的程序可以看出以下几点:

- ① C 程序以“;”作为语句结束标志。每一个说明,每一个语句都必须以分号结尾。
- ② 函数由函数首部和函数体组成。函数首部由函数类型、函数名和参数组成。函数体以花括号“{}”作为标志。

函数的一般形式:

函数返回值类型 函数名(函数形参表)

{ 声明部分 /* 说明变量部分 */

 执行部分

}

- ③ C 程序由函数组成,有且仅有一个主函数 main(),且程序从 main() 函数处开始执行,在 main() 函数中结束。

- ④ C 程序书写格式自由,可以多条语句写在一行,也可以每条语句单独写一行。

- ⑤ C 语言的注释符有两种注释方法,“/* */”为块注释,“//”为行注释(标准 C 语言不支持行注释),程序编译时,不对注释作任何处理。注释可出现在程序中的任何位置。注释用来向用户提示或解释程序的意义。

1.5 C 程序运行

C 语言源程序必须转换成计算机可执行的二进制文件(.exe 或者 .com)才能运行,C 语言采用编译方式。编译方式是将高级语言源程序翻译成目标程序后,再链接成机器可直接运行的可执行文件。执行方式如图 1.1 所示。链接的原因是由于在目标程序中要调用一些标准函数库中的标准函数,需通过链接组合成一个完整的“可执行程序”。由于产生的可执行程序可以脱离编译程序和源程序独立存在并反复使用,故编译方式执行速度快,但每次修改源程序后,必须重新编译。

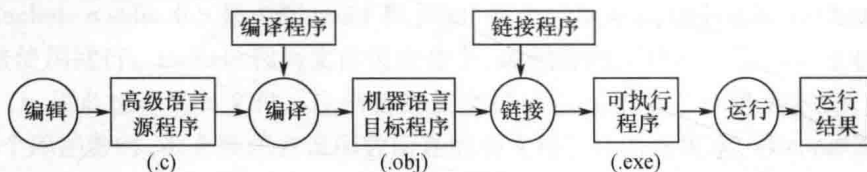


图 1.1 运行过程

1.6 Turbo C 2.0 集成开发环境的使用

1.6.1 Turbo C 2.0 简介和启动

我们上机实习和考试通常使用美国 Borland Turbo C 2.0 这个版本。该系统是 DOS 操作系统支持下的软件,在 Windows 环境下,可以在 DOS 窗口下运行。

可在 D 盘根目录下建立一个 TC 子目录安装 Turbo C 2.0 系统。TC 下还建立了两个子目录 LIB 和 INCLUDE, LIB 子目录中存放库文件, INCLUDE 子目录中存放所有头文件。

在 DOS 环境下或在 Windows 的 DOS 窗口下运行 Turbo C 2.0 时,只要在 TC 子目录下键入 TC 并回车即可进入 Turbo C 2.0 集成开发环境。

在 Windows 环境下,也可以选择运行菜单,然后键入 d:\tc\tc 进入 Turbo C 2.0 集成开发环境;还可以在 TC 文件夹找到 tc.exe 文件,然后用鼠标双击该文件名进入 Turbo C 2.0 集成开发环境。

1.6.2 Turbo C 2.0 的操作步骤

1. 进入 Turbo C 的工作环境

(1) Turbo C 的启动运行。

找到 Turbo C 所在的目录或单击桌面上的 TC 快捷方式调用 Turbo C,屏幕上显示 Turbo C 工作界面,见图 1.2。



图 1.2 Turbo C 的进入界面

(2) 熟悉 Turbo C 窗口主菜单(File、Edit、Run、Compile、Project……)。这些菜单可用键盘上的 F10 键击活,再用上、下、左、右键进行选择,用回车键激活相应子菜单。下面通过做一个简单程序来熟悉一下 Turbo C 的编辑运行过程。

2. Turbo C 的具体操作步骤

(1) 新建 C 程序:每次进入后按 F10 键,选择 File 下的 New,输入程序(系统默认文件名 noname.c)。

(2) 保存:按 F10 键或(Alt + F)组合键选择 File,出现两个保存子菜单:save(F2)和 Write to。

① Save 以当前文件名保存;Write to 以键入文件名保存,如输入 D:\41205\1,这时文件名由 * noname.c 变为 D:1.c(即证明程序已以指定的文件名保存在指定的文件夹内)如图 1.3 所示。

② 文件名后不需加尾缀,系统默认为.c(如加尾缀,必须是.c)。



图 1.3 Turbo C 的存储界面

③ 修改输出路径:运行每个 C 程序时系统自动将生成的 .obj 和 .exe 文件放在 TC 子目录中,为避免 TC 目录中文件越来越多,程序无法运行,必须修改 TC 输出路径,具体操作:按 Alt + O 组合键选择 Options - Directories \ Output directory 后在出现的对话框中键入输出路径(即存储 C 程序的其他文件夹),如图 1.4 所示。

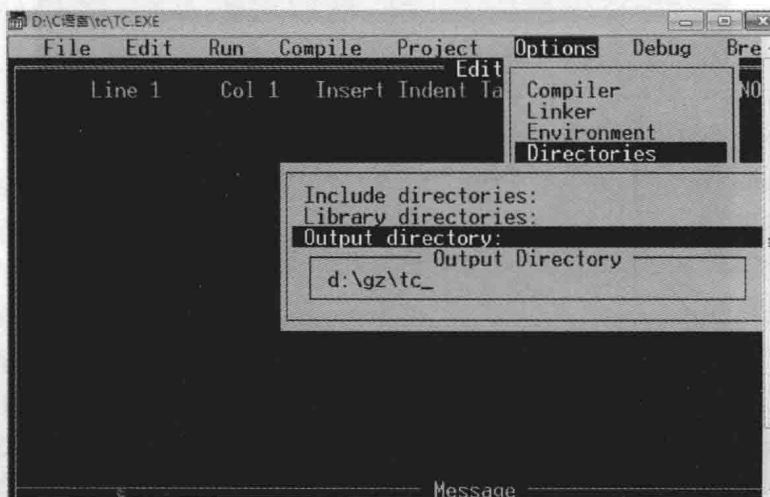


图 1.4 修改 Turbo C 的包含文件、库文件和输出目录的界面

(3) 运行:按 Alt + R 组合键选择 Run(或按 Ctrl + F9 组合键),有错误则在信息栏 Message 中显示出错信息,同时光标以红字方式停留在错误行(语句漏掉分号行除外)。如有错误则将光标移到错误处进行修改,然后再重复第(2)和(3)步操作。

注意:① 如果是漏打分号“;”,光标停在下一行。

② 如光标停在 Message 栏中,按 Alt + E 组合键选择 Edit,按任意键鼠标回到原文件,进行修改。完成后再存盘运行(可用 F2 键)。如无错误信息,则 Message 变为 Watch,这时用户看不到运行结果(因为这步只是运行程序)。

(4) 看结果(即切换到用户屏):按 Alt + R 选择 User screen (Alt + F5)。这时程序文件的结果就会显示在用户屏上。看完后按任意键,可返回 TC 窗口。建立新文件再重复

步骤(1)。

(5) 找一个已经保存的文件。

① 按 Alt + F 组合键选择 Load, 在出现的对话框中输入带有路径的文件名, 如图 1.5 所示。

② 在 TC 窗口未关闭的前提下, 可按 Alt + F 组合键选择 File 下的 Pick 子菜单, 调用本轮 TC 使用或编辑过的文件。

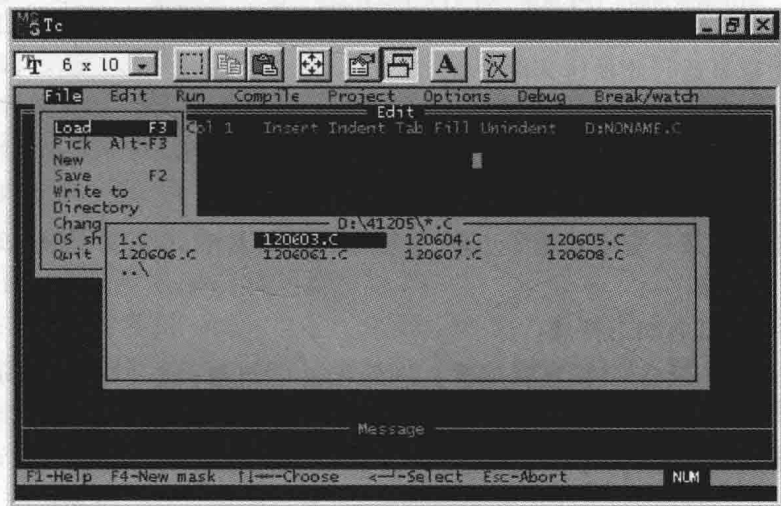


图 1.5 调入文件界面

注意: 如 Turbo C 不在 C 盘上, 还应按 Alt + O 组合键选 Directories 后, 在出现的对话框中输入库文件、包含文件、要存储的文件夹和 Turbo C 组合键执行文件夹。

(6) 运行多个文件构成的程序。

按 Alt + P 组合键可进入 Project 菜单, 该菜单包括以下内容, 如图 1.6 所示。

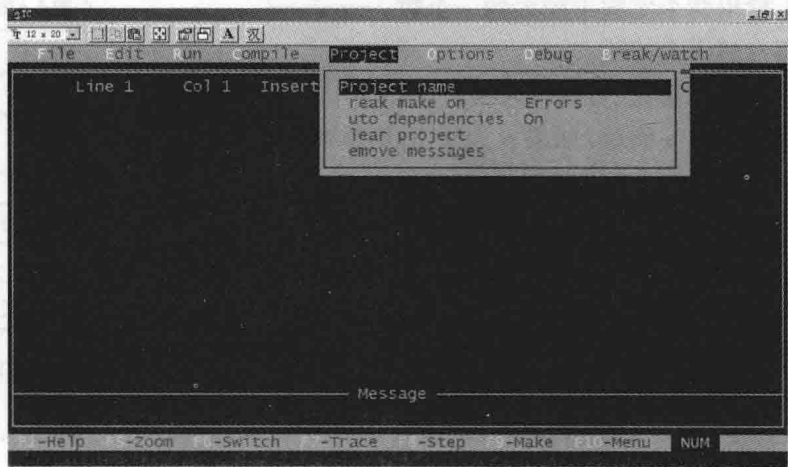


图 1.6 项目文件处理界面

项目名具有 .prj 的扩展名, 其中包括将要编译、连接的文件名。

① 先输入多个文件并分别保存(如 File1.c, File2.c, File3.c)。

② 再建立一个项目文件。后缀 .c 可有可无, 这几个文件的顺序无所谓, 其中的文件

也可以是库文件,但必须写上扩展名 . LIB。

如:d:\46105\ File1. c

d:\46105\ File2. c

d:\46105\ File3. c

③ 并以“prj”为扩展名(文件名可由用户命名)存盘。按 Alt + P 组合键,在弹出的 Project name 对话框中输入用户命名的文件名 . prj。

④ 清空项目文件名:切记每次运行项目文件,验证无误后,一定要清空 Project name 中的名字,避免重复运行此项目文件。

1.7 程序设计的算法简介

算法,就是解决某一应用问题的步骤,是程序设计的基础,采用不同的算法会带来程序的不同质量和效率。

一个算法应该具有如下特点:

(1) 易理解性。算法能够直观地描述问题的处理过程,通过算法能够理解程序的功能。

(2) 有穷性。算法仅有有限的操作步骤并且在有限的时间内完成。

(3) 确定性。算法的每一个步骤都是确定的,无二义性。

(4) 有效性。算法的每一个步骤都能得到有效的执行,并得到确定的结果。例如,如果一个算法将 0 作为除数,则该算法无效。

(5) 有 0 个或多个输入,有 1 个或多个输出。没有输出的算法没有任何意义。

算法的表示方法有多种,这里仅介绍常用的自然语言法、流程图法。

【例 1.5】输出两个实数中较大的一个数。

方法 1:用自然语言描述。

步骤 1:输入两个实数,分别存入变量 a 和 b 中;

步骤 2:比较 a 和 b 的值,如果 a 大于 b,则输出 a 的值,否则输出 b 的值。

可以看到,用自然语言描述易于理解,但冗长,难于描述复杂算法。

方法 2:用流程图表示,如图 1.7 所示。

流程图是用框图和流程线来表示程序的执行过程。常用的符号如表 1.1 所示。可以看到,用流程图进行描述,直观、形象、易于理解,是目前使用较广泛的一种方法。

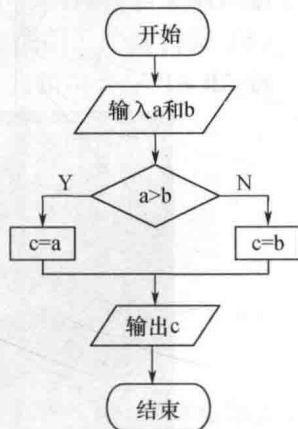


图 1.7 用流程图表示算法

表 1.1 流程图常用的符号

图 形	名 称	说 明
	流程线	表示算法的流程方向
	开始、结束框	算法的开始和结束表示