《C语言程序设计》

第四章　顺序结构程序设计

本章内容要点

- 算法的简单描述
- 数据的输入和输出
- 顺序结构程序示例

本章任务

一个程序的执行通常离不开数据的输入和输出。本章要完成的任务就是在顺序结构程序设计中，正确地进行数据的输入和输出格式控制。

任务分解：

- 根据商品原价和折扣率，计算商品的实际售价。
- 从键盘输入一个大写字母，要求改用小写字母输出。

2.1 算法

2.1.1 算法的概念

1. 算法

算法(Algorithm)一词源于算术(Algorism)。粗略地说，算术方法是一个由已知推求未知的运算过程。后来人们引申开来，把进行某一工作的方法和步骤称为算法。因此，算法反映了计算机的执行过程，是对解决特定问题的操作步骤的一种描述。

2.1 算法

2.1.1 算法的概念

2. 简单算法举例

【例3.1】求1×2×3×4×5(即5!)。

最原始的方法如下。

步骤S1：先求1×2，得到结果2。

步骤S2：将步骤1得到的乘积2乘以3，得到结果6。

步骤S3：将6再乘以4，得24。

步骤S4：将24再乘以5，得120。

这样的算法虽然正确，但太繁琐。改进的算法如下。

S1：使t=1。

S2：使i=2。

S3：使t×i，乘积仍然放在在变量t中，可表示为t×i→t。

S4：使i的值加1，即i+1→i。

S5：如果i≤5，则返回重新执行步骤S3以及其后的S4和S5；否则，算法结束。

如果计算100!，则只需将S5中的i≤5改成i≤100即可。

如果求1×3×5×7×9×11，算法也只需按如下方式做很少的改动。

S1：1→t。

S2：3→i。

S3：t×i→t。

S4：i+2→t。

S5：若i≤11，返回S3，否则，结束。

该算法不仅正确，而且对于计算机来说，是较好的算法，因为计算机是高速运算的自动机器，实现循环轻而易举。

2.1 算法

3.1.1 算法的概念

2. 简单算法举例

【例3.2】输入3个数，求其最大值。

问题分析：设num1、num2、num3存放3个数，max存放其最大值。为求最大值，就必须对3个数进行比较，可按如下步骤去做。

(1)输入3个数num1、num2和num3。
(2)先把第1个数num1的值赋给max。
(3)将第2个数num2与max比较，如果num2>max，则把第2个数num2的值赋给max(否则不做任何工作)。
(4)将第3个数num3与max比较，如果num3>max，则把第3个数num3的值赋给max(否则不做任何工作)。
(5)输出max的值，即最大值。

从该例中可以看出，首先分析题目，然后寻找一种实现这个问题所要完成功能的方法，这种方法的具体化就称为算法。因此可以说，算法是由一套明确的规则组成的一些步骤，它指定了操作顺序并通过有限个步骤来解决问题、得出结果。

2.1 算法

2.1.1 算法的概念

3. 算法的特性

(1)有穷性

(2)确定性

(3)有效性

(4)有零个或多个输入

(5)有一个或多个输出

2.1 算法

2.1.2 算法的表示

算法的表示方法很多，常用的有自然语言、传统流程图、N-S结构图、伪代码等。

1. 用自然语言表示
2. 用传统流程图表示
3. N-S结构图表示
4. 用伪代码表示

2.1 算法

2.1.2 算法的表示

1. 用自然语言表示

自然语言就是人们日常使用的语言，可以是中文、英文等。用自然语言表示算法通俗易懂，但一般篇幅冗长，表达上往往不易准确，容易引起理解上的“歧义性”。所以，自然语言一般用于算法较简单的情况。

2. 用传统流程图表示

用一些图框表示各种操作，用箭头表示算法流程。用图形表示算法直观形象、易于理解。美国标准化协会ANSI规定了一些常用的流程图符号，如图3.1所示。这些流程图符号已为世界各国程序工作者普遍采用。

2.1 算法

2.1.2 算法的表示

2. 用传统流程图表示

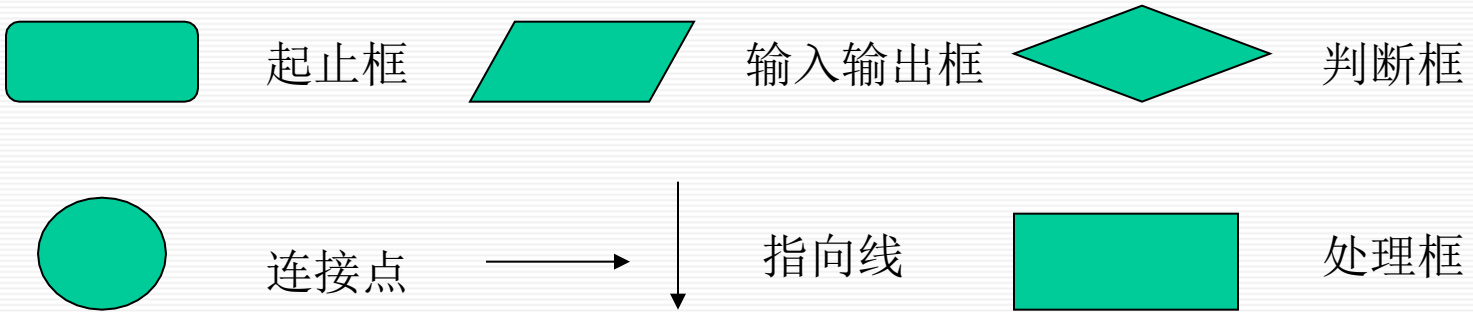


图3.1 流程图符号

起止框：表示算法的开始和结束。一般内部只写“开始”或“结束”。

处理框：表示算法的某个处理步骤，一般内部常常填写赋值操作。

输入输出框：表示算法请求输入输出需要的数据或算法将某些结果输出。一般内部常常填写“输入…”，“打印/显示…”。

菱形框(判断框)：作用主要是对一个给定条件进行判断，根据给定的条件是否成立来决定如何执行其后的操作。它有一个入口，两个出口。

连接点：用于将画在不同地方的流程线连接起来。同一个编号的点是相互连接在一起的，实际上同一编号的点是同一个点，只是画不下才分开画。

2.1 算法

2.1.2 算法的表示

2. 用传统流程图表示

下面给出3种基本结构及与其对应的流程图。

顺序结构：其对应的流程图见图3.2。

分支结构：其对应的流程图见图3.3和图3.4。

循环结构：其对应的流程图见图3.5和图3.6。

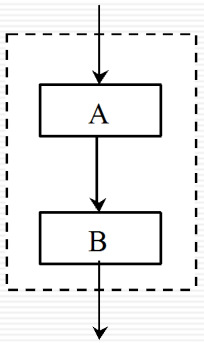


图3.2 顺序结构

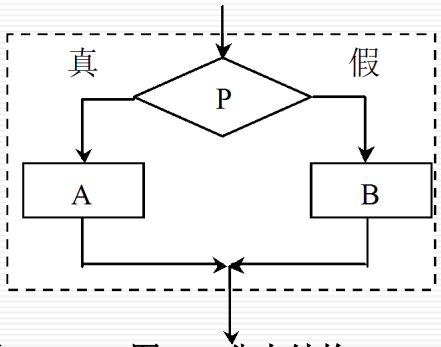


图3.3 分支结构一

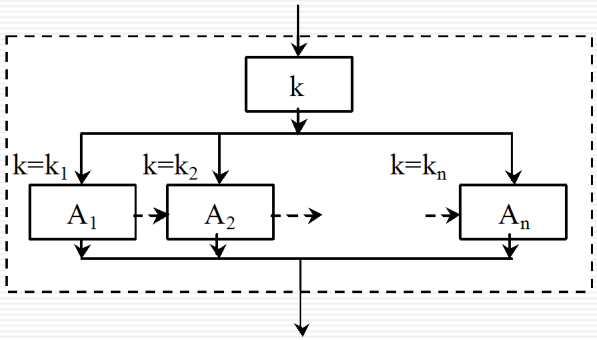


图3.4 分支结构二

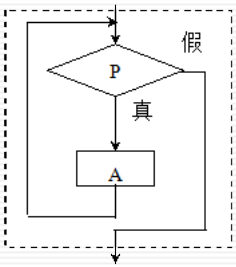


图3.5 循环结构一

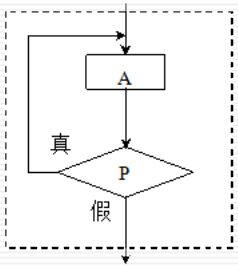


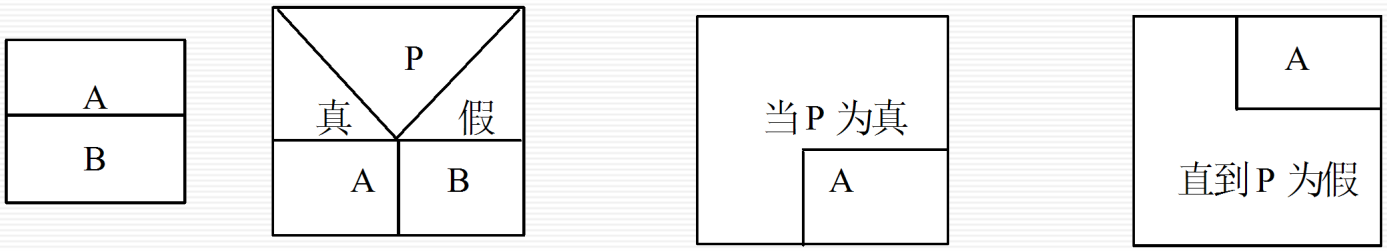
图3.6 循环结构二

2.1 算法

2.1.2 算法的表示

3. 用N—S结构图表示

三种基本结构对应的N-S图如图3.7所示。



(a)顺序结构 (b) 分支结构 (c) 循环结构

图3.7 三种基本结构对应的N-S图

2.1 算法

2.1.2 算法的表示

4. 用伪代码表示

伪代码是用介于自然语言和计算机语言之间的文字和符号来描述算法。伪代码不用图形符号，书写方便，格式紧凑，便于向计算机语言算法过渡。

【例3.3】用伪代码描述例3.2，获得求3个数中的最大值的算法。

伪代码如下：

```
input num1, num2, num3
num1→max
if num2>max then num2→max
if num3>max then num3→max
print max
```

2.1 算法

2.1.3 结构化程序设计方法

- **顺序结构：** 顺序结构是最简单的基本结构。在顺序结构中，要求顺序地执行且必须执行由行后顺序排列的每一个最基本的处理单位。
- **选择结构：** 在选择结构中，根据逻辑条件的成立与否，分别选择执行不同的处理。
- **循环结构：** 循环结构一般分为当型循环和直到型循环。
  - **当型循环：** 在当型循结构中，当逻辑条件成立时，就反复执行处理A(称为循环体)，直到逻辑条件不成立时结束(见图3.5)。
  - **直到型循环：** 在直到型循环结构中，反复执行处理A，直到逻辑条件成立结束(即逻辑条件不成立时继续执行)(见图3.6)。

解决本章任务一

【例3.4】根据商品原价和折扣率，计算商品的实际售价。程序框图如图3.8所示。

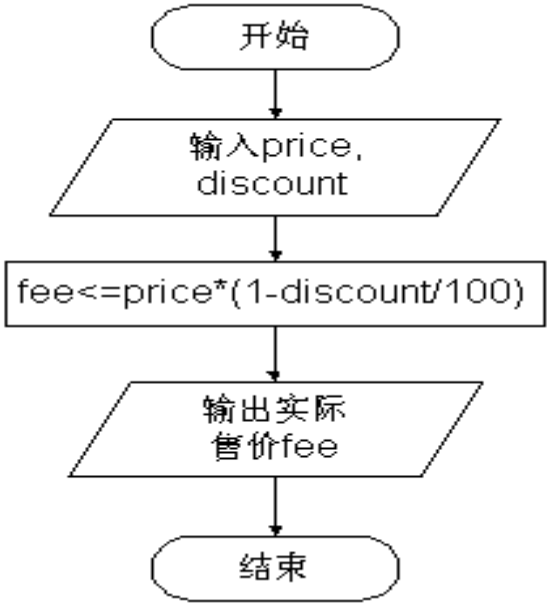


图3.8 程序框图

程序代码如下：

```
main()
{
float price, discount, fee;
printf("Input Price,Discount:");
scanf("%f%f", &price,
&discount);
fee = price * (1 - discount / 100);
printf("Fee=%.2f\n", fee);
}
```

运行结果：

```
Input Price,Discount:
100 10
Fee=90.00
```

3.2 C语句

3.2.1 控制语句

C语句主要包括控制语句、表达式语句、赋值语句、函数调用语句、复合语句、空语句等，其中存在包含关系。

控制语句用于控制程序的流程，以实现程序的各种结构。它们由特定的语句定义符组成。如表3.1所示，C语言有9种控制语句，可分成以下3类。

条件判断语句：if语句、switch语句。

循环执行语句：do-while语句、while语句、for语句。

转向语句：break语句、goto语句、continue语句、return语句。

3.2 C语句

3.2.1 控制语句

语 句	名 称
if-else	条件语句
for	循环语句
while	循环语句
do-while	循环语句
continue	结束本次循环语句
break	中止执行switch或循环语句
switch	多分支选择语句
goto	转向语句
return	从函数返回语句

表3.1 C语言的控制语句

3.2 C语句

3.2.2 表达式语句

表达式语句：表达式语句由表达式加上分号“;”组成。
其一般形式为：
表达式;

执行表达式语句就是计算表达式的值。

表达式语句可分为：
赋值语句
函数调用语句
空语句

3.2 C语句

3.2.3 特殊语句

C语言中还包括一些其他语句，如复合语句等。
把多个语句用花括号{}括起来组成的语句称复合语句。在程序中可以把复合语句看成是一条语句，而不是多条语句。

例如：

```
{
x = y + z;
    a = b + c;
    printf("%d  %d", x, a);
}
```

3.3 数据输出

在C语言中，所有的输入输出都是通过调用标准库函数中的输入输出函数来实现的。本节介绍向标准输出设备输出数据的printf函数。

printf函数称为格式输出函数，其功能是按用户指定的格式，把指定的数据输出到标准输出设备上。

3.3.1 输入输出的概念

3.3.2 格式输出函数(printf)

从计算机向外部设备(如显示器、打印机、磁盘等)输出数据称为“输出”， 从外部设备(如键盘、鼠标、扫描仪、光盘、磁盘)向计算机输入数据称为“输入”。输入/输出是以计算机主机为主体而言的。

3.3 数据输出

3.3.2 格式输出函数(printf)

printf函数称为格式输出函数。其功能是按用户指定的格式，把指定的数据显示到显示器屏幕上。

1. printf函数调用的一般形式

printf函数调用的一般形式为：

printf("格式控制字符串", 输出列表);

其中“格式控制字符串”用于指定输出格式。格式控制串可由格式字符串和非格式字符串两种组成。格式字符串是以%开头的字符串，在%后面跟有各种格式字符，以说明输出数据的类型、形式、长度、小数位数等。例如：

%d——表示按十进制整型输出
%ld——表示按十进制长整型输出
%c——表示按字符型输出

非格式字符串在输出时原样照印，在显示中起提示作用。
在“输出列表”中给出了各个输出项，要求格式字符串与各输出项在数量和类型上应该一一对应。