

中等职业教育国家规划教材配套教学用书

C 语言学习辅导 与上机实习

(计算机及应用专业)

余先锋 主编



高等教育出版社

策划编辑	陈 红
责任编辑	李 刚
封面设计	刘晓翔
版式设计	张 岚
责任校对	存 怡
责任印制	

内容提要

本书是依据中等职业教育国家规划教材《编程语言基础——C 语言》(林东、余先锋主编)编写的,同时参考了教育部考试中心制定的全国计算机等级考试大纲、二级教程——C 语言程序设计以及相关的考试参考书,结合其中的试卷、上机题等资料。本书针对 C 语言二级考试的特点进行了详细的辅导,内容涉及程序结构、基本概念、基本语句、程序设计方法、数组、函数、指针、文件等。

本书不仅可以作为主教材的辅助教材、练习册或上机实习册,而且可以作为全国计算机等级考试二级、C 语言的复习资料。

图书在版编目(CIP)数据

C 语言学习辅导与上机实习 / 余先锋主编. — 北京: 高等教育出版社, 2003.7
ISBN 7 - 04 - 012582 - X

.C... 余... .C 语言 - 程序设计 - 专业学校 - 教学参考资料 .TP312

中国版本图书馆 CIP 数据核字(2003)第 043456 号

出版发行 高等教育出版社
社 址 北京市西城区德外大街 4 号
邮政编码 100011
总 机 010 - 82028899

购书热线 010 - 64054588
免费咨询 800 - 810 - 0598
网 址 <http://www.hep.edu.cn>
<http://www.hep.com.cn>

经 销 新华书店北京发行所
排 版 高等教育出版社照排中心
印 刷

开 本 787×1092 1/16
印 张 14
字 数 330 000

版 次 年 月 第 版
印 次 年 月 第 次印刷
定 价 17.80 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

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) 82028899 转 6897 (010) 82086060

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

高等教育出版社法律事务部

邮编：100011

购书请拨打读者服务部电话：(010) 64054588

前 言

本书是中等职业教育国家规划教材《编程语言基础——C 语言》一书的配套实习教材,编写过程中参考了教育部考试中心制定的全国计算机等级考试大纲、二级教程——C 语言程序设计以及相关的考试参考书,结合其中的试卷、上机题等资料,针对 C 语言二级考试的特点进行了详细的辅导,适合中等职业学校学生使用。

本书根据《编程语言基础——C 语言》一书的教学思路,以 Turbo C 2.0 作为教学软件,内容主要分为两篇,第一篇是与原书有关章节相对应的知识要点、习题、练习。在这部分内容当中,对原书各个章节的知识要点进行了进一步的阐释,对 C 语言的基本特性、结构和基本的应用方法给出了相关知识点的分析,包括本章的必备知识、学习要点、例题精解等几个方面的内容,以及大量的选择题、填空题,并对所有的习题进行解答,对所有习题的要点进行分析。通过习题与相关的参考答案及答案分析,加深对 C 语言程序设计的理解和掌握。在这一部分内容中,根据全国计算机等级考试 C 语言程序设计中笔试的特点,对有关题型进行了分析,指出解题的要点,使学生对 C 语言程序设计考试有一个比较全面的了解并使学生熟悉考试的过程,帮助学生在将来的等级考试中发挥出应有的水平。第二篇是上机练习部分,这一部分根据课程的相关章节,给出相应的上机练习题目和内容,提出上机练习的目的和要求并对上机练习内容给出分析和提示。通过上机练习,加深学生对程序内容和方法的理解。

本书可作为 C 语言程序设计的教学参考书、上机指导和计算机等级考试指导书。学生通过对本书中内容的学习,能对主教材的程序设计思想有更深入的理解,较好地掌握计算机程序设计的基本概念和基本方法,掌握程序设计的基本要求和答题思路,更好地掌握 C 语言程序设计考试方法和技巧。同时,由于现在中等职业学校学生通常都需要通过计算机等级考试,因此也希望通过本书中有关例题、练习和上机演练,对学生通过计算机等级考试有一定的帮助。

书中附录部分是近几年全国计算机等级考试 C 语言程序考试试卷及试卷的参考答案,提供给读者练习使用。

本书第一部分内容由陈琳老师编写,上机练习部分由江南老师编写。余先锋老师整理附录中的等级考试试卷及答案,并对全书进行了统稿和审阅。本书由福建电子工业学校高级讲师陈国先老师审稿。

由于编者水平及编写时间有限,书中难免存在缺点和错误,殷切希望广大师生批评指正。

编者

2003 年 3 月 12 日于福州

目 录

第 1 章	C 语言概述	1
1.1	程序的构成	1
1.2	头文件、数据说明、函数的开始和结束标志	1
1.3	源程序的书写格式	2
1.4	C 语言的风格	2
第 2 章	基本数据类型与顺序程序设计	6
2.1	C 语言的基本数据类型及其定义方法	6
2.2	C 语言运算符的种类、运算符优先级和结合性	7
2.3	不同类型数据间的转换与运算	8
2.4	C 语言表达式类型和求值规则	9
2.5	位运算符的含义和使用以及简单的位运算	11
2.6	表达式语句、空语句、复合语句	18
2.7	数据的输入与输出以及输入输出函数的调用	19
2.8	goto 语句和语句标号的使用	21
第 3 章	分支程序设计	34
3.1	用 if 语句实现选择结构	34
3.2	用 switch 语句实现多分支选择结构	35
第 4 章	循环程序设计	44
4.1	for 循环语句	44
4.2	while 和 do while 循环语句	45
4.3	continue 语句和 break 语句	45
第 5 章	数组	58
5.1	一维数组和多维数组的定义、初始化和引用	58
5.2	字符串和字符数组	60
5.3	多维数组的定义、初始化和引用	63
第 6 章	函数	80
6.1	库函数的正确调用	80
6.2	函数的定义、类型和返回值	81
6.3	形式参数与实际参数之间参数值的传递	82
6.4	函数的正确调用、嵌套调用、递归调用	82
6.5	局部变量和全局变量	83
6.6	变量的存储类别、作用域和生存期	83
6.7	内部函数与外部函数	85

第 7 章 结构体和共用体	98
7.1 宏定义	98
7.2 “文件包含”处理	99
7.3 结构体和共用体类型数据的定义和引用	100
第 8 章 指针	108
8.1 指针与指针变量的概念、指针与地址运算符	108
8.2 指针与其他数据类型	109
第 9 章 文件	137
9.1 文件类型指针(FILE 类型指针)	137
9.2 文件的打开和关闭(fopen, fclose)	137
9.3 文件的读写	138
9.4 文件的定位	139
第 10 章 上机实验	143
实验一 熟悉 Turbo C 2.0 集成开发环境并运行简单的 C 程序	144
实验二 数据类型、运算符和表达式	146
实验三 赋值语句	147
实验四 选择语句	148
实验五 单重循环控制	150
实验六 多重循环控制	152
实验七 一维数组、字符数组与排序	154
实验八 二维数组	156
实验九 函数	158
实验十 编译预处理	161
实验十一 指针	162
实验十二 结构体和共用体	163
实验十三 位运算	166
实验十四 文件	167
附录一 全国计算机等级考试二级笔试试题——基础知识和 C 语言程序设计	169
2000 年下半年全国计算机等级考试二级笔试试题——基础知识和 C 语言程序设计	169
2001 年上半年全国计算机等级考试二级笔试试题——基础知识和 C 语言程序设计	179
2001 年下半年全国计算机等级考试二级笔试试题——基础知识和 C 语言程序设计	190
2002 年上半年全国计算机等级考试二级笔试试题——基础知识和 C 语言程序设计	199
附录二 全国计算机等级考试二级笔试试题参考答案——基础知识和 C 语言程序设计	210
2000 年下半年全国计算机等级考试二级笔试试题参考答案——基础知识和 C 语言程序设计	210
2001 年上半年全国计算机等级考试二级笔试试题参考答案——基础知识和 C 语言程序设计	211
2001 年下半年全国计算机等级考试二级笔试试题参考答案——基础知识和 C 语言	

程序设计	211
2002 年上半年全国计算机等级考试二级笔试试题参考答案——基础知识和 C 语言	
程序设计	212

第 1 章 C 语言概述

【必备知识】

1. 程序的构成, main 函数和其他函数
2. 头文件、数据说明、函数的开始和结束标志
3. 源程序的书写格式
4. C 语言的风格

【学习要点】

1.1 程序的构成

一般来说,程序可以认为是用来求解给定问题的解题步骤,而步骤的形式应符合一个程序设计语言的语法结构。这里将由高级语言编写的程序称为源程序。源程序是一个纯文本文件,它可以通过某种高级语言提供的文本编辑器来建立,也可通过其他文字处理软件建立。一个源程序不能直接运行,它必须通过编译、连接后才能运行。一个 C 源程序在编译时,将检查源程序是否有语法错误。若有错误,程序员找出错误原因,再重新进行编辑修改源程序;若没有错误,则生成一个后缀为 .obj 的二进制文件(称为目标文件)。连接后,把此 .obj 文件与 C 语言提供的各种库函数连接起来生成一个后缀为 .exe 的可执行文件。

一个 C 语言源程序主要由若干函数组成,其中有且仅有一个函数名为 main 的主函数。main 函数可以放在某些子函数的前面或后面。但放在 main 函数后面的子函数在使用前必须说明。C 程序总是从 main 函数处开始运行,其余函数则被 main 函数或其他函数调用。

1.2 头文件、数据说明、函数的开始和结束标志

C 语言本身提供了一些函数,称为标准函数。C 程序中使用这些标准函数时,必须用 #include 指令包含相应的头文件。在符合 ANSI 标准的头文件中,还包括库函数的完整原型,从而提供 stronger 的类型检查,减少出现类型匹配错误的可能。

在 ANSI 标准 C 中,用户命名的变量、函数、数组、文件、结构体、联合体等称为标识符。它们在使用之前的适当位置进行说明。标识符由若干个字符组成,其中第一个字符必须是字母或下划线,随后只能取字母、数字或下划线。注意:标识符区分大小写,且不应和 C 语言的保留关键字(例如 for、while、do、main 等)相同,中间不能有空格。

ANSI 标准 C 中共有 32 个关键字,它们都是小写的,是由系统预先定义的具有特定含义的标识符。常用的标识符有:

include、define、main、if、else、switch、case、for、do、while、exit、break、continue、goto、struct、union、void、default、static、auto、return、extern、register、enum、typedef、sizeof、short、signed、unsigned。

在函数的首部后面是函数体。函数体以左花括号“{”开始,以右花括号“}”结束。其间可以有定义(说明)部分和执行语句部分。

1.3 源程序的书写格式

C 语言源程序的书写格式自由度较高,十分灵活。主要特点如下:

(1) #include 和#define 是 C 语言中的指令,一般放在程序的开头。但它们不是语句,在其后不能加分号。

(2) 一行可以写多条语句;一条语句也可占多行(如 if、for、do 语句等),为了便于阅读,此时应采用缩进格式:

例 1 - 1:

```
scanf("% d",&d);
for(k = i = 0; i < 10; i++ )
{   f(d)
    {   k++ ;
        printf("% d",k);
    }
    scanf("% d",&d);
}
```

每条语句末尾应加一个分号。

(3) C 语言源程序中不使用行号,仅在 goto 语句中使用语句标号。

1.4 C 语言的风格

(1) C 语言的注释风格

注释可以写在几行语句之上,它一般由“/*”和“*/”来限定。编译器会忽略“/*”和“*/”的所有字符。注释的使用有助于提高程序的阅读性,便于修改和维护程序。

(2) 声明函数参数的两种风格

函数参数名放在函数名后的一对圆括号中。而函数参数的类型说明可同时置于圆括号中,也可放在圆括号后,函数最外层“{”之前。例如:

```
void  ins( int a, int b)          /* 无分号* /
{
...
}
```

或

```
void ins(a,b)                   /* 无分号* /
int a,b;                         /* 有分号* /
{
```

```
...  
}
```

注意:圆括号后不能有分号。

(3) C 语言的特点

C 语言允许对位、字节、字和指针直接操作,程序编写灵活、精练。但它对数据类型的检查和转换要求不严格,存在不安全因素;不具备自动数据边界检查功能。因此,在对数组等具有边界的数据类型赋值时,要防止出现越界现象,否则会出现不可预料的后果。

【例题精解】

1. 下列标识符组中,合法的用户标识符为_____。

- (A) b - a 和 xyz (B) CCP 和 int (C) hiy 和 p#d (D) abc 和 INT

答:选 D。

A 中的 b - a 包含减号“ - ”,不是下划线“ _ ”,是非法标识符;B 中保留字 int 不能作为标识符;因为 C 语言的标识符区分大小写,所以 INT 与 int 不同,INT 可以作为用户标识符,D 中均为合法标识符,C 中的第 2 个标识符含“ # ”,是非法标识符。

2. 下列程序的输出是_____。

```
#include "stdio.h"  
main( )  
{ printf("%d", null);  
}
```

- (A) 0 (B) 1 (C) - 1 (D) 变量未定义错误

答:选 D。

本题的关键是要弄清以下两点:

(1) 要把本题中的 null 和 C 语言中预定义标识符 NULL 区别开来。NULL 在头文件 stdio.h 中定义为宏名,它代表字符'\0'或空指针值,而 null 是小写字母拼写,不能将它当做 NULL,而只是作为一般的用户标识符。

(2) C 语言规定,程序中用到的所有变量必须在使用之前进行定义。而本程序中对 null 作输出处理之前,未给出明确的变量定义,C 编译程序不认识它,是一个未知标识符(Undefined symbol 'null' in function main),故选择 D。

3. 以下叙述正确的是_____。

- (A) C 程序由主函数组成
(B) C 程序由函数和过程组成
(C) C 程序由函数组成
(D) 在 C 程序中,无论是整型还是实型,只要在允许的范围内都能准确无误地表示

答:选 C。

C 程序由若干函数组成,其中必有一个且仅有一个主函数,每个 C 程序都是从主函数开始执行的,其他函数既可以放在主函数之前,也可以放在主函数之后(但要在使用之前说明)。C 程序中没有过程,其他高级语言程序中使用的过程在 C 程序中以函数的形式出现。在一个 C 程序的开头,可以包含若干头文件或进行宏定义。所以,选项 C 的描述是正确的,而选项 A 和 B 的描

述是错误的。

和其他高级程序设计语言一样,C 程序可以使用整型或实型数据。在允许的范围内,整型数据能准确无误地表示,但不是所有的实型数据都能准确表示,它受到数值精度的限制,有一定的误差。所以选项 D 的描述是错误的。

【习题】

一、选择题

1. 构成 C 语言程序的基本结构单位是_____。
(A) 函数 (B) 过程 (C) 复合语句 (D) 语句
2. 以下说法中正确的是_____。
(A) C 程序总是从第一个函数开始运行
(B) C 程序中,要调用的函数必须在 main() 函数中定义
(C) C 程序中,总是从 main() 函数开始执行
(D) C 程序中的 main() 函数必须放在程序的开始部分
3. C 语言中的标识符只能由字母、数字、下划线组成,下述说法正确的是_____。
(A) 标识符中的第一个字符必须是字母
(B) 标识符中的第一个字符必须是字母、下划线
(C) 标识符中的第一个字符必须是字母或下划线
(D) 标识符中的第一个字符必须是下划线
4. 当 C 语言的源程序一行写不下时,可以_____。
(A) 用分号换行
(B) 用逗号换行
(C) 用回车换行符换行
(D) 可在任意的分隔符或空格处换行
5. 以下不是 C 语言中的关键字的是_____。
(A) case (B) typedef (C) static (D) null
6. 以下选项中不合法的用户标识符是_____。
(A) __123 (B) printf (C) A \$ (D) Dim

二、填空题

1. 在 Turbo C 环境中用 RUN 命令运行一个程序时,所运行的程序的后缀是_____。
2. C 语言源程序文件的后缀是_____,经过编译后,生成文件的后缀是_____,经过连接后,生成文件的后缀是_____。
3. 结构化程序由_____,_____,_____三种基本结构组成。
4. 函数体由符号_____开始,用符号_____结束。函数体的前面是_____,其后是_____。

【习题解答】

一、选择题

1. 答:选 A。

所有 C 语言程序都是由一个或多个函数构成的,每一个 C 程序至少包含一个主函数(main 函数),因此,函数是 C 程序的基本单位(C 语言中没有过程)。C 程序的全部功能都由函数来实

现,从这个意义上说,C 程序是由函数组成的。

2. 答:选C。

C 程序中,总是从 `main()` 函数开始执行,故选项 A 是错误的,选项 C 则是正确的。C 语言中是不允许嵌套定义函数的,即使在 `main()` 函数中也不能定义别的函数,但可以在函数内部声明或调用别的函数,故选项 B 也不对。C 语言中的 `main()` 函数可以放在程序的任何地方,当然不能在某个函数体内。若 `main()` 函数所调用的函数在 `main()` 之后定义,则要在 `main()` 函数中或之前声明。

3. 答:选 C。

C 语言中,标识符由字母、数字、下划线组成,其中第一个字符可以是字母,也可以是下划线,故选 C 是对的。A 或 D 的说法均不确切,故不能选 A 或 D。因为一个字符不可能既是字母,又是下划线,所以选 B 也是错误的。

4. 答:选D。

不能用分号换行,因为分号代表一条语句的结束,也不能在一条语句的中间插入分号,因此,选项 A 是错误的;不能用逗号换行,插入逗号不可能换行,随意插入逗号,可能会出现语法错误,C 语言中,逗号一般出现在逗号表达式中或参数表中以及变量说明中;可以在任意的分隔符或空格处换行,但不能在保留字或标识符中间换行,因此选项 D 是对的,而选项 C 是错误的。

5. 答:选D。

case、typedef 和 static 均为 C 语言的关键字 (或保留字), 而 null 不是。NULL 在头文件 stdio.h 中定义为宏名, 它代表字符 '\0', 而 null 是小写字母拼写, 可以作为用户标识符。

6. 答:选 C。

因为标识符不能含有字符“\$”，所以不能将 A \$ 作为用户标识符使用；printf 是标准标识符，可以作为用户标识符使用，但失去了原来含义，即不能作为标准输出函数使用；Dim 也是合法标识符，它不是 C 语言的保留字。123 也是合法的标识符。

二、填空题

1. c

```
2. .c .obj .exe
```

3. 顺序 分支(选择) 循环

4. {	定义(说明)部分	语句执行部分
------	----------	--------

第 2 章 基本数据类型与顺序程序设计

第一部分：基本数据类型

【必备知识】

1. C 的基本数据类型及其定义方法
2. C 运算符的种类、运算符优先级和结合性
3. 不同类型数据间的转换与运算
4. C 表达式类型和求值规则
5. 位运算符的含义及使用,简单的位运算

【学习要点】

2.1 C 语言的基本数据类型及其定义方法

C 语言中的 4 种基本数据类型有:字符型(char)、整型(int)、浮点型(float)和双精度浮点型(double),它们所占字节数分别为 1、2、4、8。

char、int 之前还可加上类型修饰符 short(短型)、long(长型)、signed(有符号型)和 unsigned(无符号型),double 之前可加上 long,例如 long int、unsigned int。这些类型的数据的准确长度取决于具体的编译器,故程序员在编程时最好不要假设它们具有固定的长度。若想了解某个类型所占字节数,可使用运算符 sizeof,格式为:

sizeof(类型标识符)

例 2 - 1:

```
#include "stdio.h"
main( )
{ printf("%d", sizeof(short int));
  }          /* 在屏幕上输出整型表达式 sizeof(short int)的值* /
```

运行结果为 2,可知一个 short int 类型数据在内存中占 2 个字节。

ASCII 标准字符集定义了 128 个字符,扩展 ASCII 字符集定义了 256 个字符。在机器内部字符是以整数形式存放的。程序员可以用字符型变量保存 0 ~255 间的整数(只占 1 字节),在 - 32 768 ~32 767 间的整数可保存在 int 类型的变量中,范围在 - 2 147 483 648 ~2 147 483 647 之间的整数,可用长整型 long int 或 long 表示。例如,求 100! 的值,应用 double 型变量保存运行结果,而不能使用 long int 或 float 型变量。

例 2 - 2:

```
#include "stdio.h"
main( )
```

```
{    int i;double t= 1.0 ;
    for( i= 1 ;i< =100 ;i+ + ) t= t* i;
    printf("% 7.5 f",t);
}
```

运行结果:9.3326e + 157

若求 34!, 可使用 float 型变量保存结果, 程序如下:

```
#include " std io.h "
m ain( )
{    int i; float t= 1.0 ;
    for(i= 1 ;i< =34 ;i+ + ) t= t* i;
    printf("% 7.5 f",t);
}
```

运行结果:2.9523e + 38

若将 34 改成 35, 运行时将出现错误

F loating point error: 0 verflow . (浮点溢出错误)

若程序改为:

```
#include " std io.h "
m ain( )
{    int i; long t= 1.0 ;
    for(i= 1 ;i< =34 ;i+ + ) t= t* i;
    printf(" \n% ld \n",t);
}
```

则输出结果为 0, 显然不正确。

因此, 应根据实际问题选择适当的数据类型。

2.2 C 语言运算符的种类、运算符优先级和结合性

1. C 语言运算符, 按要求运算对象的个数可分成表 2 - 1 中的几类。

表 2 - 1

优先级	种类	例	结合方向
最高	初等运算符	() [] - > .	从左到右
次之	单目运算符	+ + - - ! ~ sizeof 等	从右到左
再次	双目运算符	* / % && != 等	从左到右
最低	三目运算符	?:	从右到左

所有单目运算符的优先级均相同, 除表 2 - 1 列出的以外, 还有以下单目运算符:

- 正号 +
- 负号 -

取地址 &
指针 *
强制类型转换 (类型)

说明:同一优先级的运算符的运算次序由结合方向决定。

2. 其他双目运算符如表 2 - 2 所示。

表 2 - 2

双目运算符	运 算 符	优先级	
算术运算符	* / %	3	高 ↑
	+ -	4	
	n (左移) m (右移)	5	
关系运算符	< <= > >=	6	
	!= ==	7	
逻辑 / 位运算符	& (位与)	8	↓ 低
	^ (位异或)	9	
	(位或)	10	
	&&	11	
		12	
赋值运算符	= += -= *= /= %= m = n = & = ^= =	14	

此外,初等运算符的优先级为 1,最高;单目运算符的优先级为 2,次之;条件运算符的优先级为 13,介于逻辑运算符与赋值运算符之间;逗号运算符的优先级为 15,最低。

2.3 不同类型数据间的转换与运算

1. 自动类型转换



图 2 - 1

在一个表达式中出现多种数据类型的常量或变量时,C 编译程序会自动根据表达式的求值顺序,逐个将每对操作数的数据类型统一。其方法是所谓的“数据类型提升”,即将小尺寸数据类型转换为大尺寸数据类型,见图 2 - 1。

运算结果的数据类型与转换后的共同数据类型一致。

注意:图 2 - 1 中的箭头方向表示数据类型级别的高低,即由低向高转换。例如 int 型和 double 型数据进行运算时,直接将 int 型数据暂时转换成 double 型数据,运算结果为 double 型,但这种暂时的转换不会改变变量定义时的数据类型。

在赋值运算时,当赋值运算符“ = ”两边的数据类型不一致时,C 编译程序会自动将右边的数据类型转换为左边的数据类型。若右边数据的优先级高,则取低字节部分数据。

例 2 - 3:

```
#include " std io . h "
```

```

main( )
{
    int t;char c;
    t=259; c = t;
    printf(" \n% d,% d\n?", t,c);
}

```

运行结果:

259,3

259 的二进制表示为

0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

c 取低字节(00000011)₂, 等于十进制的 3。

2. 强制类型转换

```

#include "stdio.h"
main( )
{
    int t=1;double x=3.5;
    t=(int)x;          /* 强制类型转换,相当于取整,x 仍为 double 型* /
    x=t;               /* 自动类型转换,t 仍为 int 型* /
    printf(" \n% d,% f\n",t,x);
}

```

运行结果:

3,3.000000

例 2 - 4:

```

#include "stdio.h"
main( )
{
    int t=1;float c,f;
    f=t/2;             /* 两整数相除,结果为整数,此时为 0,0 转换为浮点数后赋给 f* /
    c=(float)t/2;      /* 强制类型转换后,变成浮点数除以整数,结果为浮点数* /
    printf(" \n% f,% f\n",f,c);
}

```

运行结果:

0.000000,0.500000

2.4 C 语言表达式类型和求值规则

1. C 语言表达式类型

表达式是由操作数和运算符组成的式子。操作数可以是常量、变量、函数和子表达式。一个常量、变量或函数也可以看作一个表达式,它们均可以单独出现在表达式可以出现的地方。表达式的值由组成它的操作数的值和运算符类型决定。