**实验1 输入输出**

**一、实验目的**

（1）掌握在Visaul C++ 环境如何编辑、编译和运行C源程序；

（2）掌握各种类型数据的输入输出的方法，能正确使用各种格式转换符。

**二、实验内容和步骤**

为了方便管理自己的C语言程序，在启动Visual C++ 6.0集成开发环境前，首先在E盘创建一个新的文件夹，以便存放自己的C语言程序。

1、启动Visual C++ 6.0集成开发环境

如图1-1所示，单击“开始”→“程序”→Microsoft Visual Studio 6.0→Microsoft Visual C++ 6.0，就可以启动Visual C++ 6.0集成开发环境。

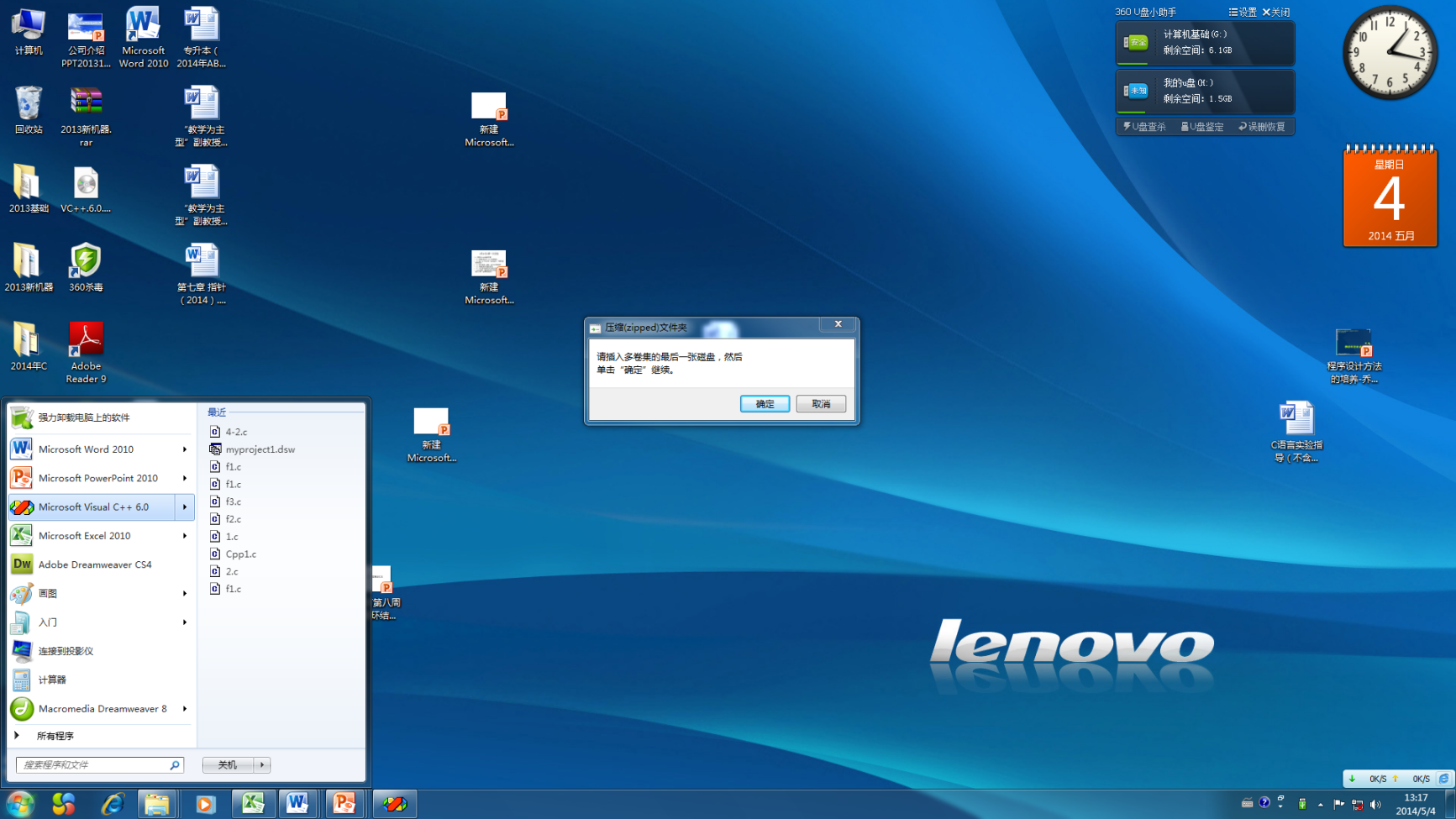


图1-1启动Visual C++ 6.0的方法

启动后的Visual C++ 6.0集成开发环境如图1-2所示。

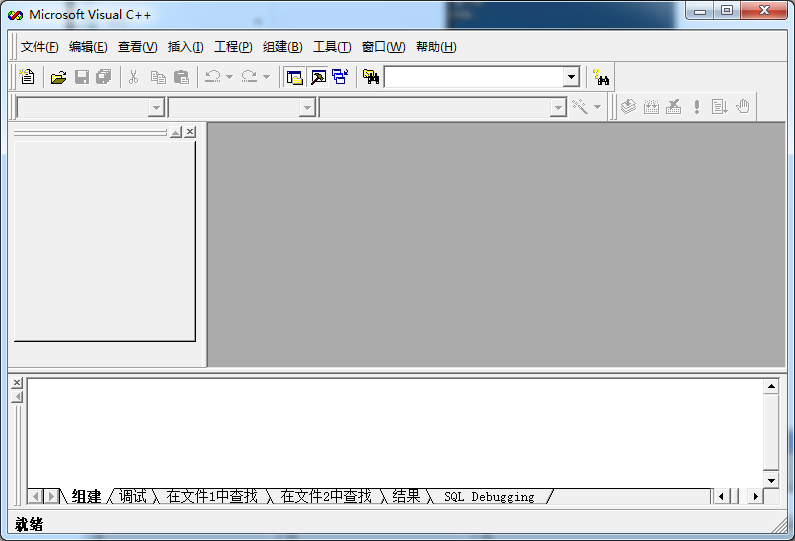
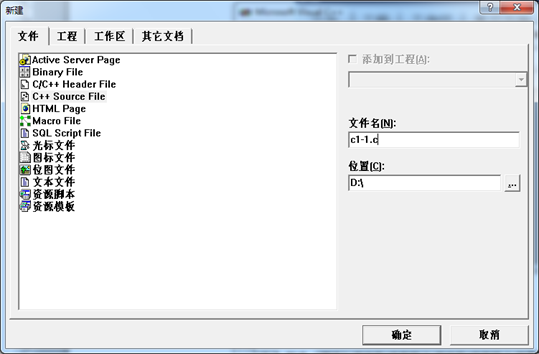


图1-2 Visual C++ 6.0 中文版集成开发环境

2、开始一个新程序

（1）创建文件

单击主菜单中的“文件”→“新建”菜单命令，弹出“新建”对话框，在“新建”对话框中选择“文件”选项卡。在左边列出的选项中，选择“C++ Source File”；在右边的相应对话框中，输入文件名称“c1-1.c”及保存的位置，如图1-3所示。单击“确定”按钮。图1-3创建新的C源文件

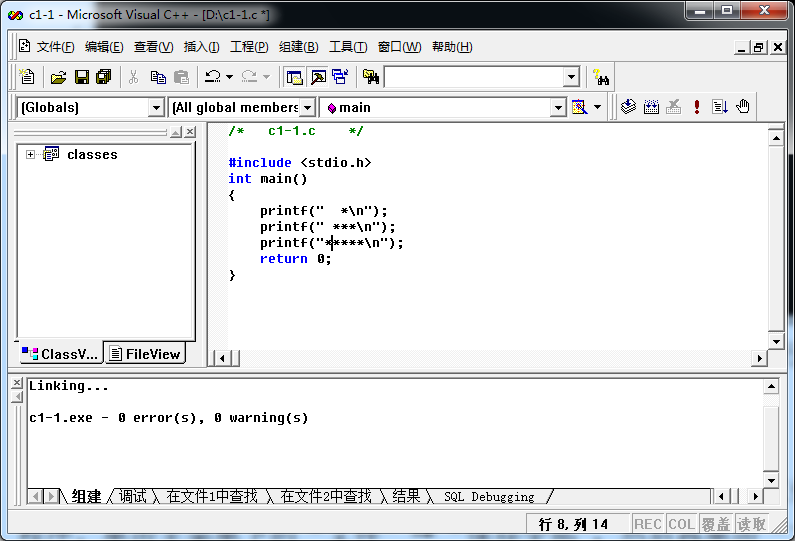
进入Visual C++ 6.0集成环境的代码编辑窗口，如图1-4所示。

图1-4 Visual C++ 6.0集成环境的代码编辑窗口

（2）代码编辑

在Visual C++ 6.0代码编辑窗口中，输入如下所示的源代码，完成后如图1-4中所示。

程序代码：

/\* c1-1.c \*/

#include <stdio.h>

int main( )

{

printf(" \*\n");

printf(" \*\*\*\n");

printf("\*\*\*\*\*\n");

return 0;

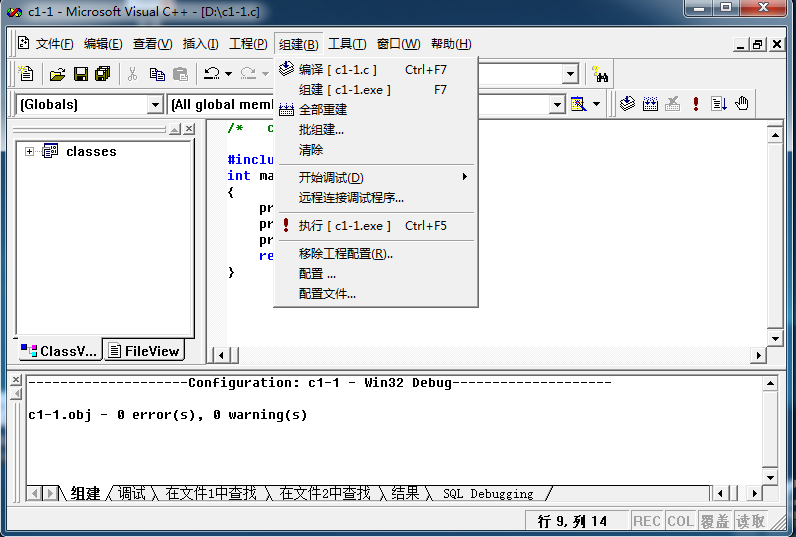
}

（3）程序的编译、连接与运行

将C语言源代码编译成计算机能执行的目标代码。

单击主菜单下的“组建”→“编译 [c1-1.c]”（或者是工具栏上的按钮或按快捷键Ctrl+F7），此时将弹出一个对话框，询问是否创建一个项目工作区，选择“是（Y）”。Visual C++ 6.0集成开发环境会自动在c1-1.c文件所在文件夹中建立相应的项目文件。

编译时，在下方的输出框中将显示出相应的编译说明，如图1-5所示。

图1-5 Visual C++ 6.0 集成环境下编译源程序

如果代码编译无误，最后将显示：

c1-1.obj - 0 error(s), 0 warning(s)

这说明编译没有错误（error）和警告（warning），生成目标文件c1-1.obj，程序编译顺利完成。目标文件（.obj）不能被计算机直接执行，接下来将目标文件（.obj）和相关的库函数或目标程序连接成为可执行程序（.exe）。

单击主菜单下的“组建”→“批组建”命令，将弹出如图1-6所示的对话框。

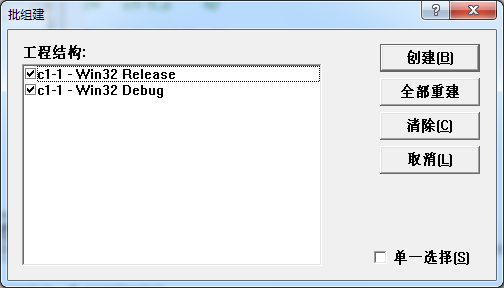


图1-6 Visual C++ 6.0 集成环境下批组建对话框

确保选中“c1-1-Win32 Release”复选框，这样生成的可执行文件才是发行版的程序，否则生成的是调试（Debug）版的程序。

单击“创建”按钮，生成可执行文件c1-1.exe。如果在“批组建”对话框中选中了两个复选框，可以看到程序中生成了两个c1-1.exe可执行文件，一个文件为调试版本，存储在与c1-1.c同一文件夹下的Debug文件夹中；另一个是发行版本，保存在与c1-1.c同一文件夹下的Release文件夹中。

这一步只是为了生成发行版的程序文件，只有在程序准备发行时才需要执行这种编译。在通常情况下，可以单击主菜单下的“组建”→“组建 [c1-1.exe]”（或工具栏按钮或按快捷键F7），直接生成调试版本程序就可以了。

编译、连接完成后， c1-1.exe已经是一个独立的可执行程序，可以在Windows 资源管理器中直接执行，也可以在Visual C++ 6.0集成开发环境中运行。

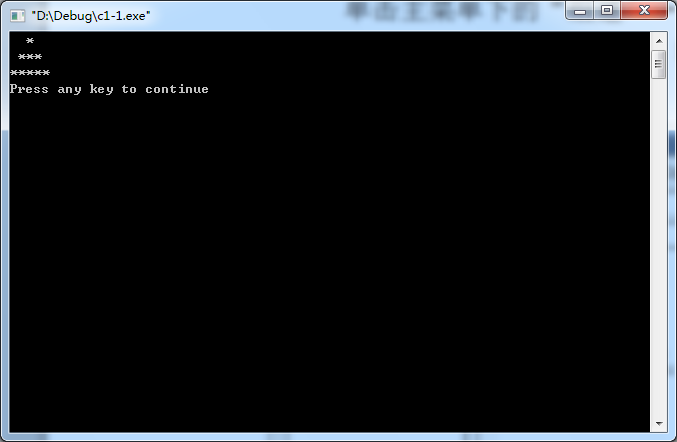
单击主菜单下的“组建”→[执行[c1-1.exe]（或工具栏按钮或按快捷键Ctrl+F5），此时弹出一个控制台程序窗口，程序正确运行，如图1-7所示。按任意键后返回Visual C++ 6.0集成开发环境。

图１－7 Visual C++ 6.0 集成环境下程序运行结果

（4）关闭工作空间

当第1～3步工作完成后，应该将工作保存下来，并关闭工作空间，以便做下一个新的程序。单击主菜单下的“文件”→“保存全部”，然后再单击“文件”→“关闭工作空间”菜单命令，询问确认要关闭所有文档窗口，选择“是（Y）”。

(5) 打开已有的文件

在打开已有的文件时一定要先确认关闭了工作空间，然后单击主菜单下的“文件”→“打开”命令，选择创建的c1-1.c文件，然后打开。

**3、输入并运行以下程序,掌握基本输入输出语句的使用**

关闭所有工作空间，点击Visual C++ 6.0窗口的关闭按钮即可退出Visual C++ 6.0集成环境，按照以上步骤再次进入Visual C++ 6.0集成环境，**输入以下程序，运行并查看输出结果。**

（1）整型数据的定义和赋值、输出

/\* c1-2-1.c \*/

#include <stdio.h>

int main()

{

int x,y,sum;

printf(“Input x and y:\n”);

x=5;

y=6;

sum=x+y;

printf(“x+y=%d”,sum);

return 0;

}

将该程序的x,y改成从键盘输入，重新调试该程序，源程序命名为c1-2-2.c

（2）浮点数数据的定义和赋值、输出

/\* c1-3-1.c \*/

#include <stdio.h>

int main()

{

float fa,fb;

fa=3141.592678;

fb=123.4;

printf("float类型数据的打印结果：\n”);

printf(“fa=%f\t fb=%f\n",fa,fb);

return 0;

}

将该程序的fa,fb改成从键盘输入，重新调试该程序，源程序命名为c1-3-2.c

（3）字符数据类型的定义和赋值、输出

/\* c1-4-1.c \*/

#include <stdio.h>

int main()

{

char c1,c2;

c1=‘A’;

c2=‘B’;

printf(“c1=%c;c2=%c”,c1,c2);

return 0;

}

将该程序的x,y改成从键盘输入，重新调试该程序，源程序命名为c1-4-2.c

**4．输入并运行以下程序,掌握格式化输入输出语句的使用**

（1）输入并运行下面的程序，掌握scanf（）函数输入多个整型数据时，格式说明中无分隔符的正确使用。

/\* c1-5.c \*/

#include <stdio.h>

int main( )

{ int i,j;

printf("Enter i,j\n");

scanf("%d%d",&i,&j);

printf("i=%d,j=%d\n",i,j);

return 0;

}

*注意：*运行程序时，先执行printf("Enter i,j\n")，当调用格式输入函数scanf（）时，返回用户屏幕，屏幕上会有提示Enter i,j，等待用户从键盘上输入两个整数。当调用格式输入函数scanf（）时，首先返回用户屏幕，等待用户从键盘上输入两个整数并回车，程序才能继续向下执行。

从键盘上为变量i,j赋值32和18时，两个整数之间*可用空格、Tab或回车键分隔*。试一试，用其它的分隔符输入时各个变量，能否得到正确值。*注意：*printf("Enter i,j\n");语句对下面的输入语句起提示作用。

（2）输入并运行下面的程序，掌握scanf（）函数输入多个整型数据时，格式说明中逗号分隔符的正确使用。

/\* c1-6.c \*/

#include "stdio.h"

int main( )

{ int i,j;

printf("Enter i,j\n");

scanf("%d,%d",&i,&j);

printf("i=%d,j=%d\n",i,j);

return 0;

}

*注意：*程序运行时，从键盘上为变量i,j赋值32和18，两个整数之间必须*用逗号分隔*。试一试用其它的分隔符输入时各个变量能否得到正确值。

（3）输入并运行下面的程序，掌握格式scanf( )函数中普通字符按原样输入的使用方法。

/\* c1-7.c \*/

#include <stdio.h>

int main( )

{ float i,j;

scanf("i=%f,j=%f",&i,&j);

printf("i=%f,j=%f\n",i,j);

return 0;

}

从键盘上输入i=12.5,j=-4<回车>，观察程序运行的结果。试一试用其它的方式输入时各个变量能否得到正确值。

（4）输入并运行下面的程序，如果格式scanf( )函数中格式与变量类型对应关系错误，将不能得到正确的结果。

/\* c1-8.c \*/

#include <stdio.h>

int main( )

{ int i,j;

scanf("%f,%f",&i,&j);

printf("i=%d,j=%d\n",i,j);

return 0;

}

运行程序，注意观察程序运行时出现的问题。

（5）输入一个大写字母A，将它转换为小写字母a，输出小写字母a及对应的ASCII码值97。要求输出格式为：j=a,j=97。完善下面程序中的输出语句：

/\* c1-9.c \*/

#include <stdio.h>

int main( )

{ int j;char i;

scanf("%c",&i);

j=i+32; /\* 大小写ASCII码值相差32 \*/

***输出语句;***

return 0;

}

（6）熟悉并改正出现的错误！！！以下程序有很多语法错误，请改正！

/\* c1-10.c \*/

#includ<stdio.h>

mian()

{

intbottom,high,area; /\*bottom表示底，high表示高，area表示面积

bottom=4

High=6;

area=bottom\*high/2；

print=(“The atra is %d\n,area);

**实验2顺序程序设计**

**一、实验目的**

（1）掌握C语言中使用最多的一种语句——赋值语句的使用方法。

（2）掌握C语言的顺序结构程序设计

（3）掌握C语言的各种数据类型以及整型、字符型、实型变量的定义；

（4）掌握C语言中有关算术运算符及表达式的使用。

**二、实验内容和步骤**

**1、输入程序，按要求观察并分析程序运行的结果**

（1）浮点数数据的定义和赋值、输出

/\* c2-1.c \*/

#include <stdio.h>

int main()

{

float fa,fb,fc,fsum;

double d;

fa=3141.592678;

fb=6.026e-27;

fsum=fa+fb ;

d=31415926.78;

printf("float类型数据的打印结果：\n");

printf("fa=%f\t fb=%f\t fsum=%7.2f\n",fa,fb,fsum);

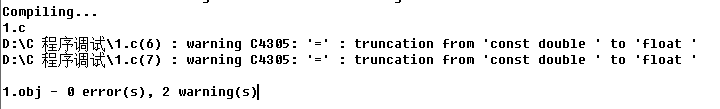
printf("double类型数据的打印结果：\n ");

printf("d=%lf, d=%e\n",d,d);

return 0;

}

分析:编译时为什么会出现如下的warning信息:



（2）字符类型和整型类型的互通性，分析输出结果

/\* c2-2.c \*/

#include <stdio.h>

int main()

{

char c1='a',c2='b',c3='c';

c1=c1+2;

c2=c2+8;

c3=c3-32;

printf("char类型数据的打印结果：\n ");

printf("c1=%c\t c2=%c\t c3=%c\n",c1,c2,c3);

printf("char类型数据和int类型的关系：\n");

printf(" c1=%d\t c2=%d\t c3=%d",c1,c2,c3);

return 0;

}

（3）整型数据类型的范围试验，分析b和d的值

/\* c2-3.c \*/

#include <stdio.h>

int main()

{

int a,b,c,d ;

a = 2147483647; /\* 设定最大正整数 \*/

b = a+1;

c = -2147483648; /\* 设定绝对值最大负整数 \*/

d = c-1;

printf("正数范围a=%d , b=%d\n", a, b) ;

printf("负数范围c=%d , d=%d\n", c, d) ;

return 0;

}

(4)其它数据类型试验

/\* c2-4.c \*/

#include <stdio.h>

int main()

{ int a,b;

unsigned c ,d;

long e;

a = 100;

b = -100;

e = 50000;

c = a;

d = b;

printf(“%d,%d\n”, a, b);

printf(“%u,%u\n”, a, b); //按无符号格式输出a,b

printf(“%u,%u\n”, c, b);

a=e;

printf("%d\n",a);

return 0；

}

请对照程序和运行结果分析：

1)将一个负整数赋给一个无符号的变量，会得到什么结果。

2)将一个大于2147483647的长整数赋给整型变量，会得到什么结果。

3)将一个长整数赋给无符号变量，会得到什么结果（分别考虑该长整数的值大于或等于4294967295 以及小于4294967295 的情况）。

同学们可以改变程序中各变量的值，以便比较。

例如： a = 4294967300, b = -3147483647, e = 4294967295, f = 4294967300。

(5)算术运算符“/”、“%”

/\* c2-5.c \*/

#include "stdio.h"

int main( )

{ int a,b;

a=2;

b=1%a;

printf("%d\n",1/a);

printf("b=%d\n",b);

printf("%f %f\n",(float)(1/a),(float)b);

return 0;

}

观察程序运行结果，从中掌握类型转换运算符的使用。

(6)输入以下程序，运行程序并按要求修改程序。

/\* c2-6.c \*/

#include <stdio.h>

int main()

{ int a ;

float d ;

char c1 ;

double f ;

long m ;

unsigned p ;

a = 61;

c1 = 'a';

d = 3.56;

f = 3157.890121;

m = -2147483647;

p =-2147483647;

printf(" a=%d \n c1=%c \n d=%6.2f \n", a, c1, d);

printf(" f=%15.12f \n m=%ld \n p=%u \n", f, m, p);

return 0;

}

①运行此程序并分析结果(特别注意变量p输出的值)。

②改用scanf函数输入数据而不用赋值语句，scanf函数如下：

scanf(“%d,%c,%f,%lf,%ld,%u”,&a,&c1,&d,&f,&m,&p);

输入的数据如下：**61,a,3.56,3157,0.123456789,-2147483648,-2147483647**↙

请分析运行结果（说明：lf和ld格式符分别用于输入double型和long型数据）。

*注意：*程序运行时，scanf输入各类不同数据时采用的分隔符的情况。

③在将printf语句改为：

printf(“ a=%d \n c1=%c\n d=%15.6f\n”,a,c1,d);

printf(“ f=%f\n m=%d\n p=%d\n”, f, m, p);

运行程序。

④将p改用%o格式符输出。

⑤将scanf函数中的%lf和%ld改为%f和%d，运行程序并观察分析结果。

（7）将a,b两个变量的值交换后，如果按“a=2,b=1”的格式输出，完善下面程序中的输出语句。

/\* c2-7.c \*/

#include <stdio.h>

int main( )

{ int a=1,b=2,t;

t=a;

a=b;

b=t; /\* 以上三条赋值语句实现a,b变量值的交换 \*/

***输出语句;***

return 0;

}

（8)计算输出当x=2.5,a=7,y=4.7时，表达式x + a % 3 \* (int) (x + y) % 2 / 4的运算结果z的值。填写缺失的关键字，并观察以下程序的运行结果。

/\* c2-8.c \*/

#include <stdio.h>

int main( )

{

① a=7;

② x=2.5,y=4.7，z；

z= x + a % 3 \* (int) (x + y) % 2 / 4;

printf("z=%f\n",z);

return 0；

}

程序不完整，请删除横线后填空并运行程序。

（9）不同数据类型的混合运算

/\* c2-9.c \*/

#include <stdio.h>

int main( )

{ int n=100;

float f ;

double d ;

f=n+2.5;

printf("f=%f\n",f);

n=f/2;

printf("n=%d\n",n);

n=(1/2)\*f;

printf("n=%d\n",n);

d=f\*2;

printf("d=%lf\n",d);

return 0;

}

观察程序运行结果，从中掌握赋值中的自动类型转换，并关注该程序在VC6.0下编译时出现的警告信息。

**2、编写程序**

（1）编写一个程序，程序的功能是：从键盘上输入一个华氏温度，能够输出相应的摄氏温度。华氏温度F与摄氏温度c的转换关系为：

5

c= ──(F-32)

9

要求输出保留2位小数。

*提示：*先定义变量，应考虑定义的变量F和c应为什么类型？哪个变量存放原始数据，哪个变量存放计算结果？再输入华氏温度，应考虑输入格式与键盘操作的一致性。然后计算转换后的摄氏温度，应注意怎样将一个数学表达式写成c的算术表达式。最后输出计算结果，考虑怎样实现输出保留两位小数。

*程序框架：*

*#include <stdio.h>*

*int main( )*

*{ 定义需要的变量类型;*

*键盘输入已知华氏温度F;*

*由赋值语句实现转换为华氏温度c;*

*输出语句;*

*}*

（2）编写一个程序，两次调用getchar()函数读入两个字符分别赋给c1和c2，再分别用putchar( )函数和printf( )函数输出这两个字符。

*提示：*编程时先定义两个字符型变量c1、c2，然后两次调用getchar( )函数分别将读入的两个字符赋给c1和c2，再分别调用putchar( )和printf( )函数输出c1和c2。特别注意：*输入两个字符时必须是连续的*，两个字符之间不能有分隔字符，否则系统会把分隔字符作为第二个字符读入。

*程序框架：*

*#include <stdio.h>*

*int main( )*

*{ 定义需要的变量（注意类型）;*

*用getchar从键盘读入c1;*

*用getchar从键盘读入c2;*

*用putchar输出变量c1,c2;*

*用printf输出变量c1,c2;*

*return 0;*

*}*

（3）设圆半径r=1.5，圆柱高h=3，求圆周长、圆面积、圆球表面积、圆球体积、圆柱体积。用scanf输入数据，输出计算结果。输出时要有文字说明，取小数点后两位数字。

注意：要根据需要定义数据类型。

**3.扩展实验**

（1）从键盘输入一个同学的学号及高数、英语、C语言3门课的成绩，计算平均分，并写入文件score\_avg.txt中。

例如：从键盘输入数据：1001 90 80 70，文件score\_avg.txt中的记录应该为：

学号 高数 英语 C语言 平均分

1001 90 80 70 80.00

（2）改写（1）中的程序：学号及3门课成绩从文件score.txt中读入（而不是从键盘读入），写入文件score\_avg.txt中。

score.txt文件格式如下:

