

## Visual C++ 2010 集成开发环境

Visual C++ 2010 是 Microsoft 公司的 Visual Studio 2010 开发工具箱中的一个 C++ 程序开发包,它不仅支持 C++ 语言的编程,同时也兼容 C 语言的编程,因此,Visual C++ 2010 被广泛地用于各种编程,使用面很广。本部分内容将简要地介绍如何在 Visual C++ 2010 下运行 C 语言程序。

### 1.1 熟悉 Visual C++ 2010 环境

#### 1.1.1 启动 Microsoft Visual Studio 2010

在“开始”菜单中执行 Microsoft Visual Studio 2010→Microsoft Visual Studio 2010 命令,进入 Microsoft Visual Studio 2010(简称 VS 2010)编程环境。如果桌面上有 Microsoft Visual Studio 2010 的快捷方式,则可双击该图标,打开 VS 2010 的集成开发环境窗口,如图 1-1 所示。



图 1-1 Microsoft Visual Studio 2010 中文版主窗口

### 1.1.2 编辑、编译和运行程序

#### 1. 新建项目

在 VS 2010 窗口中,执行“文件(File)”→“新建(New)”命令,在弹出的“新建项目”对话框中选择“Win32 控制台应用程序(Win32 Console Application)”,在“位置(Location)”文本框中输入准备建立的项目的存储路径,在其上的“名称(Name)”文本框中输入准备建立的项目的名字,如图 1-2 所示。

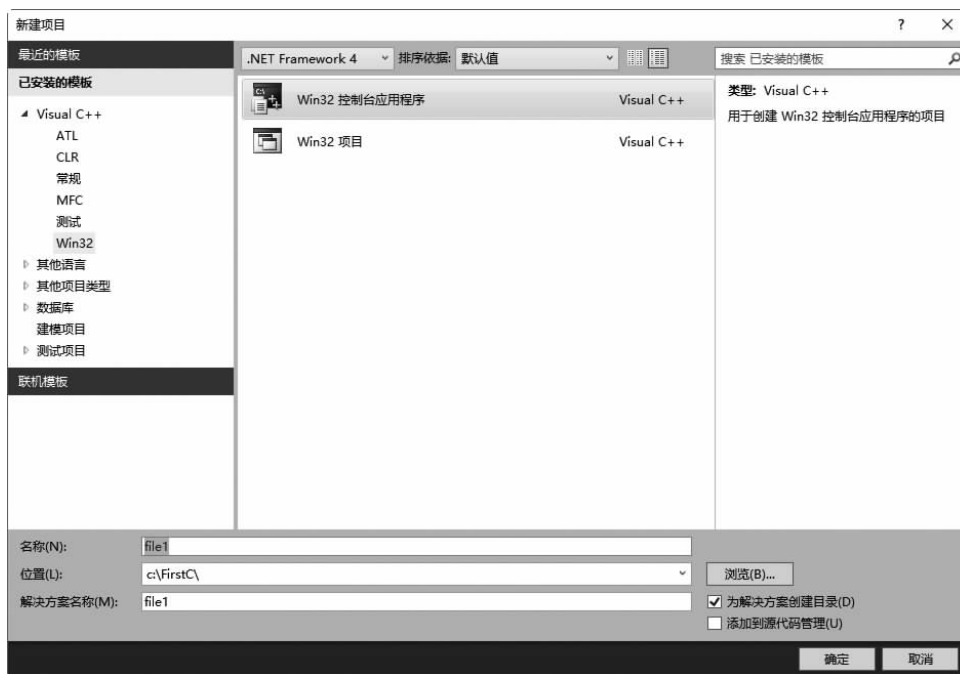


图 1-2 创建项目

单击“确定(OK)”按钮,然后单击“下一步(Next)”按钮,进入图 1-3 所示的“应用程序设置”界面,界面选中“空项目(Empty Project)”复选框,单击“完成(Finish)”按钮。



图 1-3 选择空项目

## 2. 新建文件

在 VS 2010 解决方案资源管理器窗口的“源文件(Source Files)”处右击,在弹出的快捷菜单中选择“添加(Add)”→“新建项(New Item)”命令,显示如图 1-4 所示的添加新项(Add New Item)对话框。在该对话框中选择“C++ 文件(C++ File)”选项,在“名称(Name)”文本框中输入源文件名称,然后单击“添加(Add)”按钮。

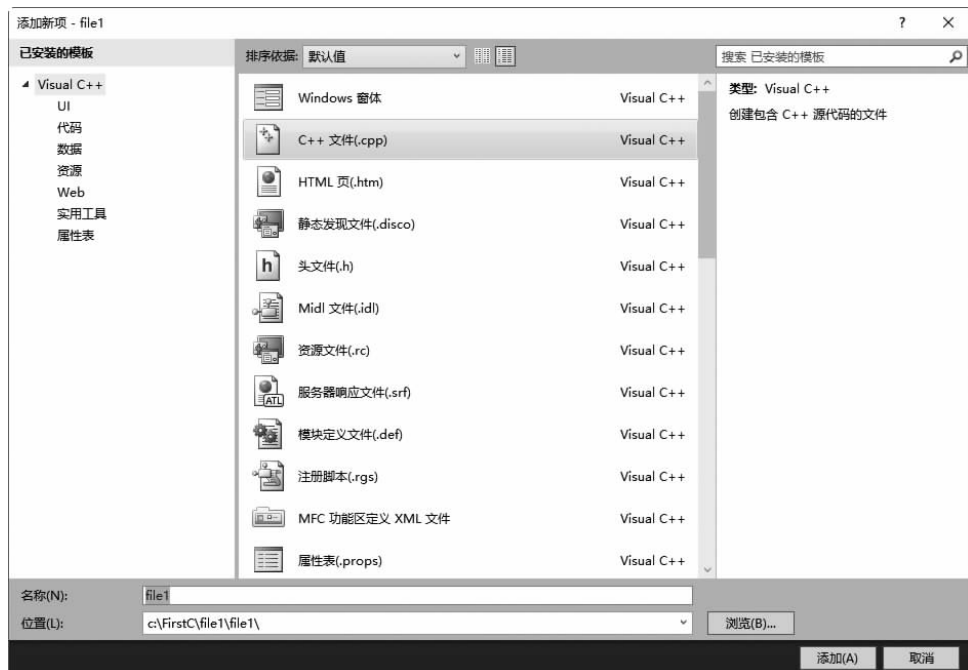


图 1-4 新建文件

## 3. 编辑和保存源文件

在编辑窗口中输入源程序,然后选择“文件(File)”→“保存(Save)”命令或“文件(File)”→“另存为(Save as)”命令保存文件。在编辑窗口中写源程序代码时,一般采用缩进风格,在缩进时既可以使用空格键也可采用 Tab 键来实现。另外,也可以采用 Microsoft Visual Studio 2010 软件自动排版,方法是:先把源程序代码输入编辑窗口,然后选中全部代码,接着按 Alt+F8 快捷键,就可以完成代码的自动排版。

## 4. 生成解决方案

选择“生成(Build)”→“生成解决方案(Build Solution)”命令,或按 Ctrl+F7 快捷键,将对源程序文件进行编译连接。

在输出窗口中显示错误(error)或警告(warning)信息。若有错误,可以通过单击输出窗口右侧的上下滚动按钮,在窗口中依次双击出错信息,在编辑源程序窗口中就会出现一个箭头指向程序中出错的位置,一般在箭头的当前行或上一行可以找到出错语句。在改正错误时,应从第一条错误开始修改,每修改一处错误,重新生成一次,直至出现“0 error(s), 0 warning(s)”。当没有错误与警告出现时,输出窗口所显示的最后一行应该是“成功 1 个,失败 0 个,最新 0 个,跳过 0 个”,如图 1-5 所示。

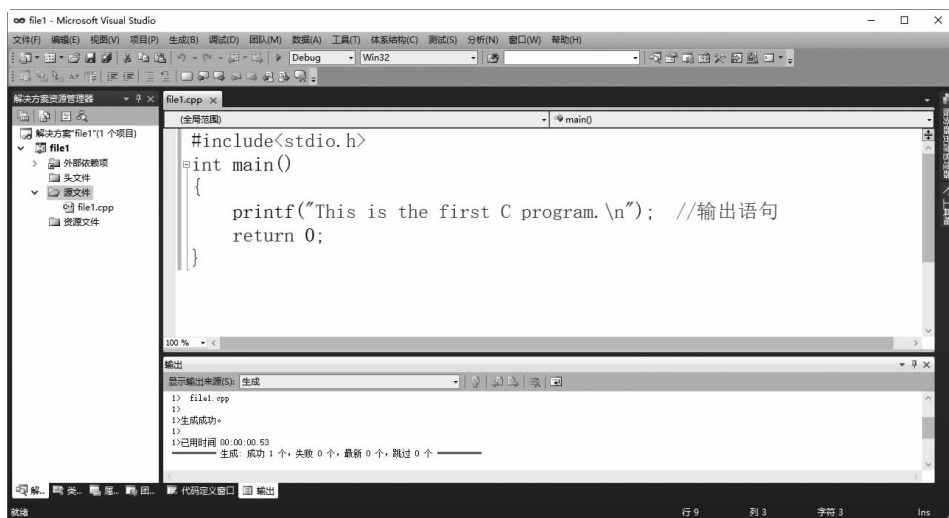


图 1-5 编译后的输出窗口

## 5. 运行程序

生成成功后,自动生成可执行文件。选择“调试(Debug)”→“开始执行(不调试)(Start Without Debugging)”选项,也可按 Ctrl+F5 快捷键运行可执行程序,执行后将出现一个类似于 DOS 窗口的界面,如图 1-6 所示。



图 1-6 执行文件窗口

## 6. 关闭项目

当程序调试成功以后,需要先关闭当前项目,才能进行下一个项目的调试。执行“文件(File)”→“关闭解决方案(Close Solution)”命令,即可关闭当前项目。

## 7. 打开项目

项目关闭后,若想重新打开该项目进行检查或修改,可执行“文件(File)”→“打开(Open)”→“项目或解决方案(Project/Solution)”命令,找到该项目的解决方案名,双击即可。

# 1.2 在 Microsoft Visual Studio 2010 环境下调试 C 语言程序

## 1.2.1 创建 C 语言源程序文件

按照上一节讲述的方法创建名为 prog1.cpp 的源程序文件,如图 1-7 所示。

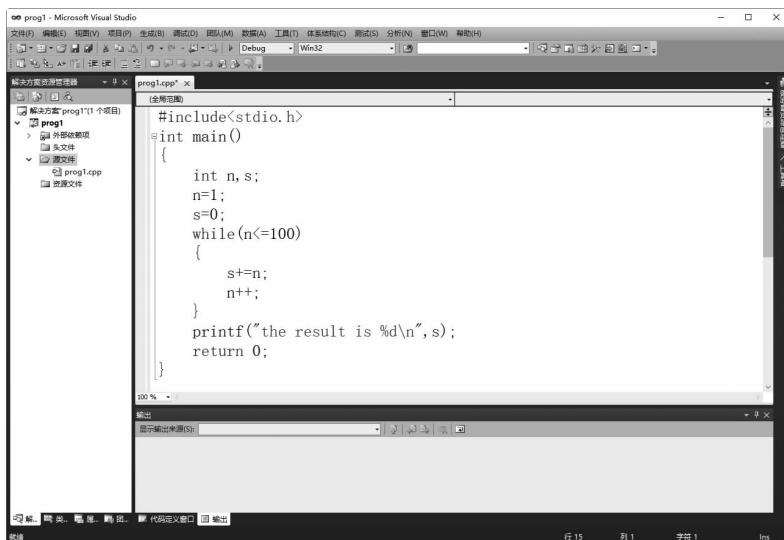


图 1-7 输入、编辑 C 语言源程序窗口

## 1.2.2 调试程序

Visual C++ 2010 是一个完全基于 Windows 系统的开发环境,程序调试可通过单击进行,具体的调试方法有两种。

### 1. 方法一

设置光标或断点,使程序执行到中途暂停,以便观察阶段性结果。

(1)使程序到光标所在行暂停。

①在需暂停的行上单击,定位光标。

②执行“调试(Debug)”→“启动调试(Start Debugging)”命令,或按 F5 键,程序将执行到光标所在行暂停,如图 1-8 所示。如果把光标移动到后面某个位置,再按 F5 键,程序将从当前的暂停位置继续执行到新的光标位置,第二次暂停。

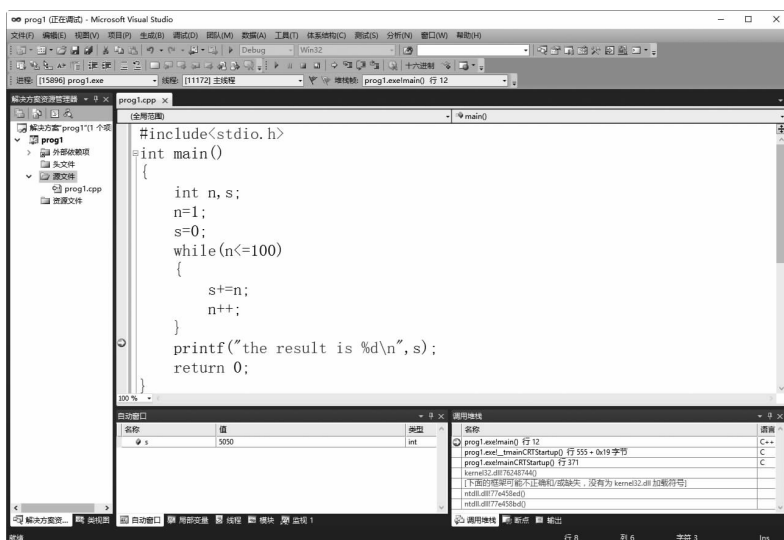


图 1-8 程序运行到光标处暂停

(2)在需要暂停的行上设置断点,断点通常用于调试较长的程序。

①执行“调试(Debug)”→“启动调试(Start Debugging)”命令,先对程序进行编译链接。

②将光标定位在断点所在行,按 F9 键,该操作是一个开关,按一次是设置,被设置了断点的行前面会有一个红色圆点标识,如图 1-9 所示,按第二次是取消设置,红色圆点标识消失。

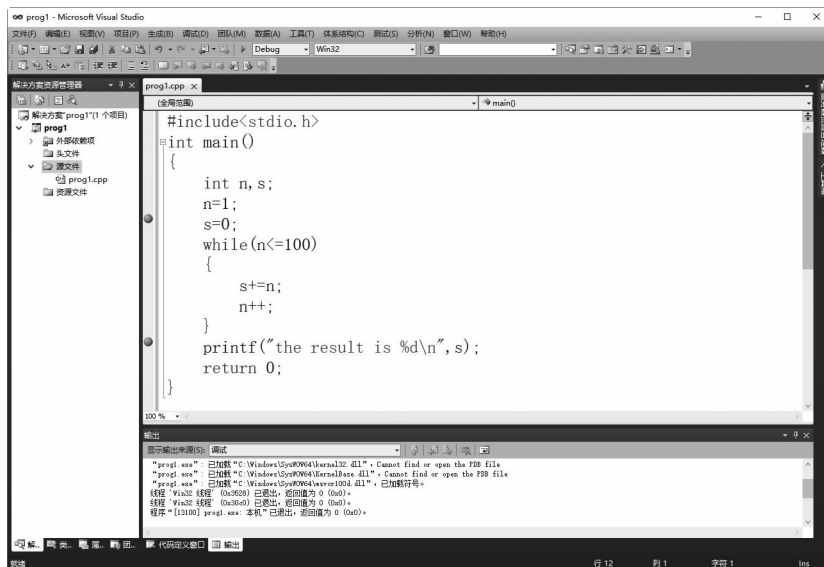


图 1-9 程序设置断点窗口

如果程序中有多个断点想全部取消,可执行“调试(Debug)”→“删除所有断点(Toggle Breakpoint)”命令,或按 Ctrl+Shift+F9 快捷键,随后在弹出的“是否删除所有断点”对话框中单击“是”按钮,将取消所有断点。

## 2. 方法二

当程序执行到某个位置时发现结果已经不正确了,说明在此之前肯定有错误码存在,如果能确定一小段程序可能有错,先按上面步骤暂停在该小段程序的头一行,再输入若干个查看变量,然后单步执行,即一次执行一行语句,逐行检查下来,查看到底是哪一行语句造成结果出现错误,从而能确定错误的语句并予以纠正。单步执行的快捷键是 F11,如果要结束函数的单步执行,可按 Shift+F11 快捷键。

## C 语言程序上机实验

C 语言程序设计是一门实践性很强的课程,对于初学者来说,不断上机调试是一个必不可少的环节,有助于大家深入理解知识点及掌握应用 C 语言解决实际问题的能力。另外,通过上机实验,也有助于学习者熟练掌握编程平台 Visual C++ 2010。

### 2.1 C 语言的基本数据类型与表达式

#### 2.1.1 实验目的

(1)掌握 C 语言数据类型的概念,熟悉如何定义一个整型、字符型、实型变量,以及如何对它们进行赋值。

(2)掌握整型、字符型、实型变量数据输出时所用的格式转换符。

(3)学会使用 C 语言的有关算术运算符,正确使用由这些运算符构成的表达式。

(4)进一步熟悉如何编辑、编译、链接和运行一个 C 语言程序。

#### 2.1.2 实验内容

(1)输入并运行以下程序。

```
#include <stdio.h>
int main( )
{
    char c1,c2,c3;
    c1='x';
    c2='y';
    c3='z';
    printf(" %c, %c, %c\n",c1,c2,c3);
    printf(" %d, %d, %d",c1,c2,c3);
    return 0;
}
```

**【参考解答】**

```
x,y,z
120,121,122
```

(2)输入以下程序,并给出程序运行后的结果。

```
#include <stdio.h>
int main()
{
    int a,b,c,d;
    unsigned u;
    a=12;
    b=-24;
    u=10;
    c=a+u;
    d=b+u;
    printf("a+u= %d,b+u= %d\n",c,d);
    return 0;
}
```

**【参考解答】**

a+u=22,b+u=-14

(3)输入并运行以下程序。

```
#include <stdio.h>
int main( )
{
    int a,c;
    float b;
    a=24,b=128.565;
    c=a+(int)b;
    printf(" %d, %f\n",c,b);
    return 0;
}
```

**【参考解答】**

152,128.565000

(4)输入并运行以下程序。

```
#include <stdio.h>
int main( )
{
    int a=100;
    printf(" %d\n",++a);
    printf(" %d\n",a++);
    printf(" %d\n",a);
    return 0;
}
```

**【参考解答】**

101



101

102

(5)输入并运行以下程序。

```
#include <stdio.h>
int main( )
{
    int a=78,b=34;
    int c,d;
    c=((a>b)?a:b);
    d=((a<b)?a:b*b);
    printf("c= %d,d= %d\n",c,d);
}
```

【参考解答】

c=78,d=1156

(6)输入并运行以下程序。

```
#include <stdio.h>
int main( )
{
    char x='A';
    putchar('A');
    putchar('\101');
    putchar('\n');
    putchar('\x41');
    putchar(x);
    return 0;
}
```

【参考解答】

AA

AA

(7)输入并运行以下程序。

```
#include <stdio.h>
int main( )
{
    int i,j,m=0,n=0;
    i=10;
    j=11;
    m+=i++;
    n-=--j;
    printf("i= %d,j= %d,m= %d,n= %d\n",i,j,m,n);
    return 0;
}
```

**【参考解答】**

i=11,j=10,m=10,n=-10

(8)输入并运行以下程序。

```
#include <stdio.h>
int main( )
{
    int a1,a2;
    float b,c;
    b=23.52;
    c=67.54;
    a1=(int)(b+c);
    a2=(int)b+(int)c;
    printf(" %d, %d\n",a1,a2);
    return 0;
}
```

**【参考解答】**

91,90

(9)输入并运行以下程序。

```
#include <stdio.h>
int main( )
{
    int a=3,b=9,c=8,d=10;
    a+=b;
    c*=d+a;
    d/=a;
    a%=c;
    printf(" %d, %d, %d, %d",a,b,c,d);
    return 0;
}
```

**【参考解答】**

12,9,176,0

(10)编写一个程序,对输入的一个三位整数反向输出。例如,若输入 123,则输出 321。

**【参考解答】**

```
#include <stdio.h>
int main( )
{
    int num,a,b,c;
    scanf(" %d",&num);
    a=num/100;
    b=num%100/10;
    c=num%10;
}
```

```

    num=c * 100+b * 10+a;
    printf("% d\n",num);
    return 0;
}

```

## 2.2 顺序结构程序设计

### 2.2.1 实验目的

- (1)掌握 C 语言中赋值语句的使用。
- (2)掌握数据输入/输出函数的使用,能正确使用各种格式转换符。
- (3)熟悉顺序结构程序中语句的执行过程,并学会调试程序的基本方法。
- (4)能设计简单的顺序结构程序。

### 2.2.2 实验内容

- (1)输入并运行以下程序。

```

#include <stdio.h>
int main()
{
    int a,b;
    a=165;
    b=68;
    printf("% x\n",a);
    printf("% o\n",b);
    return 0;
}

```

#### 【参考解答】

a5  
104

- (2)输入并运行以下程序。

```

#include <stdio.h>
int main()
{
    int a,b,c;
    scanf("% o, % o", &a, &b);
    c=a * b;
    printf("% d, % o\n", c, c);
    return 0;
}

```

#### 【参考解答】

12,10 ✓

80,120

(3) 输入并运行以下程序,分析程序运行结果。

```
#include <stdio.h>
int main()
{
    int a=10;
    long int b=10;
    float x=10.0;
    double y=10.0;
    printf("a= %d,b= %ld,x= %f,y= %lf\n",a,b,x,y);
    printf("a= %ld,b= %d,x= %lf,y= %f\n",a,b,x,y);
    printf("x= %f,x= %e,x= %g\n",x,x,x);
    return 0;
}
```

**【参考答案】**

a=10,b=10,x=10.000000,y=10.000000

a=10,b=10,x=10.000000,y=10.000000

x=10.000000,x=1.000000e+001,x=10

(4) 输入并运行以下程序,分析程序运行结果。

```
#include <stdio.h>
int main()
{
    float a=12.56,b=6.31;
    char x='A';
    printf(" %f, % .2f\n",a,b);
    printf("a+b= %6.2f\n",a+b);
    printf(" %6.2f- %6.2f= %6.2f\n",a,b,a-b);
    putchar(x);
    putchar(x+32);
    printf("\n");
    return 0;
}
```

**【参考答案】**

12.560000,6.31

a+b= 18.87

12.56- 6.31= 6.25

Aa

(5) 输入并运行以下程序,分析程序运行结果。

```
#include <stdio.h>
int main()
{
```

```

int a=7,b=5,c,d;
float x=4.5,y=2.5,z;
z=(x+y)/2+(a+b)/2;
d=a%b*(int)y;
c=(++a)+(b++);
x*=a+b;
printf("%d,%d,%d,%d",a,b,c,d);
printf("%f\t%f\n",x,y);
printf("%f\n",z);
return 0;
}

```

**【参考解答】**

8,6,13,463.000000 2.500000

9.500000

(6)输入并运行以下程序,分析程序运行结果。

```

#include <stdio.h>
int main()
{
    int a=3,b=5,c;
    a+=a-=a*a;
    printf("a= %d\n",a);
    c=(c=2,c+b)*c;
    printf("b= %d,c= %d\n",b,c);
    return 0;
}

```

**【参考解答】**

a=-12

b=5,c=14

(7)编写程序,要求输入一个华氏温度,输出摄氏温度。公式为  $C=5/9(F-32)$ ,其中, $F$  表示华氏温度, $C$  表示摄氏温度。

**【参考解答】**

```

#include <stdio.h>
int main()
{
    float C,F;
    printf("请输入华氏温度:");
    scanf("%f",&F);
    C=5.0/9*(F-32);
    printf("result= %.2f",C);
    return 0;
}

```

(8)编写一个程序,将输入的两个三位正整数 a、b 合并形成一个长整数并存放在 c 中,合并的方法是:将数 a 的百位、十位和个位分别放在数 c 的十万位、千位和十位上,数 b 的百位、十位和个位分别放在数 c 的万位、百位和个位上。例如,当 a=456,b=123 时,运行后的结果为 c=415263。

**【参考解答】**

```
#include <stdio.h>
int main()
{
    int a,b;
    long int c;
    printf("请输入两个整型数:");
    scanf("%d,%d",&a,&b);
    c=a/100*100000+a%100/10*1000+a%10*10;
    c+=b/100*10000+b%100/10*100+b%10;
    printf("result=%ld",c);
    return 0;
}
```

(9)编写程序,定义两个长整型变量,输入两个长整型数,交换两个变量的值后输出。

**【参考解答】**

```
#include <stdio.h>
int main()
{
    long int x,y,z;
    printf("请输入两个长整型数:");
    scanf("%ld,%ld",&x,&y);
    z=x;
    x=y;
    y=z;
    printf("x=%ld,y=%ld",x,y);
    return 0;
}
```

(10)编写程序,输入一个 5 位数,求各位上数字之和后输出。

**【参考解答】**

```
#include <stdio.h>
int main()
{
    int x,a,b,c,d,e,sum=0;
    printf("请输入一个五位整数:");
    scanf("%d",&x);
    a=x/10000;
    b=x%10000/1000;
    c=x%1000/100;
```

```

d=x%100/10;
e=x%10;
sum=a+b+c+d+e;
printf("sum= %d",sum);
return 0;
}

```

## 2.3 选择结构程序设计

### 2.3.1 实验目的

- (1)掌握关系运算符、逻辑运算符及条件运算符的使用,学会 C 语言中条件值真假的判断方法。
- (2)进一步熟悉并掌握运算符的优先级和结合性的有关规定。
- (3)掌握 if 和 switch 等选择结构流程控制语句的用法,学会分析和设计选择结构的程序。
- (4)掌握选择结构程序的上机测试方法。

### 2.3.2 实验内容

- (1)输入并运行以下程序,分析程序运行结果。

```

#include <stdio.h>
int main()
{
    int a=3;
    if(a==5)
        printf(" *** \n");
    else
        printf(" ### \n");
    return 0;
}

```

#### 【参考解答】

```
###
```

- (2)输入并运行以下程序,分析程序运行结果。

```

#include <stdio.h>
int main()
{
    int a=3,b=4,c,d,e;
    c=!a>b;
    d=(a!=b&& a<b);
    e=a>b? a:b;
    printf("c= %d,d= %d,e= %d\n",c,d,e);
    return 0;
}

```

**【参考解答】**

c=0,d=1,e=4

(3)输入并运行以下程序,分析程序运行结果。

```
#include <stdio.h>
int main()
{
    int a,b;
    scanf("%d",&a);
    if(a%2!=0)
    {
        b=a*a;
        printf("b=%d\n",b);
    }
    return 0;
}
```

**【参考解答】**

5 ✓

b=25

(4)输入并运行以下程序,分析程序运行结果。

```
#include <stdio.h>
int main()
{
    int a,b,c;
    scanf("%d,%d,%d",&a,&b,&c);
    switch(a)
    {
        case 1:
        case 2:
        case 3: printf("%d+%d=%d\n",b,c,b+c); break;
        case 4: printf("%d-%d=%d\n",b,c,b-c); break;
        case 5: printf("%d*%d=%d\n",b,c,b*c); break;
        case 6: printf("%d/%d=%d\n",b,c,b/c); break;
    }
    return 0;
}
```

**【参考解答】**

3,4,5 ✓

4+5=9

(5)输入并运行以下程序,分析程序运行结果。

```
#include <stdio.h>
int main()
```



```

{
    char c='k';
    int i=1,j=2,k=3;
    double x=3e+5,y=0.85;
    printf(" %d, %d\n",'a'+5<c,-i-2*j>=k+1);
    printf(" %d, %d\n",1<j<5,x-5.25<=x+y);
    printf(" %d, %d\n",i+j+k== -2*j,k==j==i+5);
    return 0;
}

```

**【参考解答】**

1,0

1,1

0,0

(6) 输入并运行以下程序,分析程序运行结果。

```

#include <stdio.h>
int main()
{
    char c='k';
    int i=1,j=2,k=3;
    double x=3e+5,y=0.85;
    printf(" %d, %d\n",!x*!y,!!!x);
    printf(" %d, %d\n",x||i&&j-3,i<j&&x<y);
    printf(" %d, %d\n",i==5&&c&&(j=8),x+y||i+j+k);
    return 0;
}

```

**【参考解答】**

0,0

1,0

0,1

(7) 编写程序计算下列分段函数的值。

$$y = \begin{cases} 2 & x > 1 \\ 0 & x = 1 \\ -2 & x < 1 \end{cases}$$

**【参考解答】**

```

#include <stdio.h>
int main()
{
    int x,y;
    printf("please input x:");
    scanf(" %d",&x);
    if(x>1)

```

```
        y=2;
    else if(x==1)
        y=0;
    else
        y=-2;
    printf("the result is %d",y);
    return 0;
}
```

(8)编写程序,输入任意一个字符,判断其类别(控制字符、数字字符、大写字母、小写字母或其他字符),并给出提示。

**【参考解答】**

```
#include <stdio.h>
int main()
{
    char c;
    printf("please input a character:");
    c=getchar();
    if(c<32)
        printf("this is a control character\n");
    else if(c>='0'&&c<='9')
        printf("this is a digit\n");
    else if(c>='A'&&c<='Z')
        printf("this is a capital letter\n");
    else if(c>='a'&&c<='z')
        printf("this is a small letter\n");
    else
        printf("this is an other character\n");
    return 0;
}
```

(9)编写程序,输入一个数字(1~7),输出数字对应的英文形式的星期。

**【参考解答】**

```
#include <stdio.h>
int main()
{
    int a;
    printf("please input a integer number:");
    scanf("%d",&a);
    switch(a)
    {
        case 1: printf("Monday\n");break;
        case 2: printf("Tuesday\n");break;
```

```

        case 3: printf("Wednesday\n");break;
        case 4: printf("Thursday\n");break;
        case 5: printf("Friday\n");break;
        case 6: printf("Saturday\n");break;
        case 7: printf("Sunday\n");break;
        default: printf("error\n");
    }
    return 0;
}

```

(10) 编写程序,输入年号和月份,输出该月有多少天。

**【参考解答】**

```

#include <stdio.h>
int main()
{
    int year,month,flag;
    printf("please input year and month:");
    scanf("%d%d",&year,&month);
    if((year%4==0&&year%100!=0)||(year%400==0))
        flag=1;
    else
        flag=0;
    switch(month)
    {
        case 1:
        case 3:
        case 5:
        case 7:
        case 8:
        case 10:
        case 12: printf("31");break;
        case 4:
        case 6:
        case 9:
        case 11: printf("30");break;
        case 2: if(flag) printf("29"); else printf("28"); break;
        default: printf("error");
    }
    return 0;
}

```