

北京信息科技大学计算机学院软件工程系

# 程序设计基础(C 语言)

---

## 实验指导书

**hbs@bistu.edu.cn**

**2017/10/13**



## 目录

|                     |     |
|---------------------|-----|
| 图形目录 .....          | III |
| 表格目录 .....          | V   |
| 1 C 语言的运行环境 .....   | 1   |
| 1.1 实验目的 .....      | 1   |
| 1.2 上机内容 .....      | 1   |
| 1.3 分析和讨论 .....     | 3   |
| 1.4 作业 .....        | 3   |
| 2 C 语言数据类型和运算 ..... | 4   |
| 2.1 实验目的 .....      | 4   |
| 2.2 上机内容 .....      | 4   |
| 2.3 分析和讨论 .....     | 6   |
| 2.4 作业 .....        | 6   |
| 3 简单的 C 程序设计 .....  | 7   |
| 3.1 实验目的 .....      | 7   |
| 3.2 上机内容 .....      | 7   |
| 3.3 分析和讨论 .....     | 8   |
| 3.4 作业 .....        | 8   |
| 4 选择结构程序设计 .....    | 10  |
| 4.1 实验目的 .....      | 10  |
| 4.2 上机内容 .....      | 10  |
| 1.1 分析和讨论 .....     | 12  |
| 4.3 作业 .....        | 12  |
| 5 循环结构程序设计 .....    | 14  |
| 5.1 实验目的 .....      | 14  |
| 5.2 上机内容 .....      | 14  |
| 5.3 分析和讨论 .....     | 14  |
| 5.4 作业 .....        | 14  |

|     |            |    |
|-----|------------|----|
| 6   | 数组 .....   | 16 |
| 6.1 | 实验目的 ..... | 16 |
| 6.2 | 上机内容 ..... | 16 |
| 6.3 | 分析和讨论..... | 16 |
| 6.4 | 作业 .....   | 17 |
| 7   | 函数 .....   | 18 |
| 7.1 | 实验目的 ..... | 18 |
| 7.2 | 上机内容 ..... | 18 |
| 7.3 | 分析和讨论..... | 25 |
| 7.4 | 作业 .....   | 25 |
| 8   | 指针 .....   | 27 |
| 1.1 | 实验目的 ..... | 27 |
| 1.2 | 上机内容 ..... | 27 |
| 1.3 | 分析和讨论..... | 27 |
| 1.4 | 作业 .....   | 27 |
| 9   | 结构体 .....  | 28 |
| 1.1 | 实验目的 ..... | 28 |
| 1.2 | 上机内容 ..... | 28 |
| 1.3 | 分析和讨论..... | 28 |
| 1.4 | 作业 .....   | 28 |
| A.  | 附录 .....   | 1  |
|     | 参考文献 ..... | 1  |
|     | 索引 .....   | 1  |

## 图形目录

|                    |    |
|--------------------|----|
| 图 1-1 3 题运行结果..... | 1  |
| 图 1-2 5 题运行结果..... | 1  |
| 图 1-3 调试程序 .....   | 2  |
| 图 1-4 调试程序 .....   | 2  |
| 图 1-5 运行结果 .....   | 3  |
| 图 2-1 1 题运行结果..... | 4  |
| 图 2-2 2 题运行结果..... | 4  |
| 图 2-3 3 题运行结果..... | 5  |
| 图 3-1 2 题运行结果..... | 7  |
| 图 3-2 3 题运行结果..... | 7  |
| 图 3-3 4 题运行结果..... | 8  |
| 图 4-1 1 题运行结果..... | 10 |
| 图 4-2 2 题运行结果..... | 11 |
| 图 4-3 3 题运行结果..... | 12 |
| 图 7-1 1 题运行结果..... | 18 |
| 图 7-2 2 题运行结果..... | 18 |
| 图 7-3 设置断点 .....   | 20 |
| 图 7-4 程序暂停 .....   | 21 |
| 图 7-5 调试程序 .....   | 22 |
| 图 7-6 调试程序 .....   | 23 |
| 图 7-7 调试程序 .....   | 24 |
| 图 7-8 4 题运行结果..... | 25 |

|                     |    |
|---------------------|----|
| 图 7-9 4 题运行结果 ..... | 25 |
|---------------------|----|

## 表格目录

未找到图形项目表。





# 1 C 语言的运行环境

## 1.1 实验目的

- 熟悉使用 C 语言运行环境（VC6.0），初步了解一个 C 语言程序的建立、编辑、编译、链接、调试和运行等。
- 掌握 C 语言程序的基本语法和格式，了解 C 语言特点，能够编写一个简单的 C 语言程序。

## 1.2 上机内容

1. 启动 VC6.0
2. 创建一个新文件，类型为 C++ Source File，新文件名称为：my\_1.c，位于 D:\MyCProgram 目录下（也可位于其它目录）
3. 编写程序。该程序在屏幕上显示如下字符串：

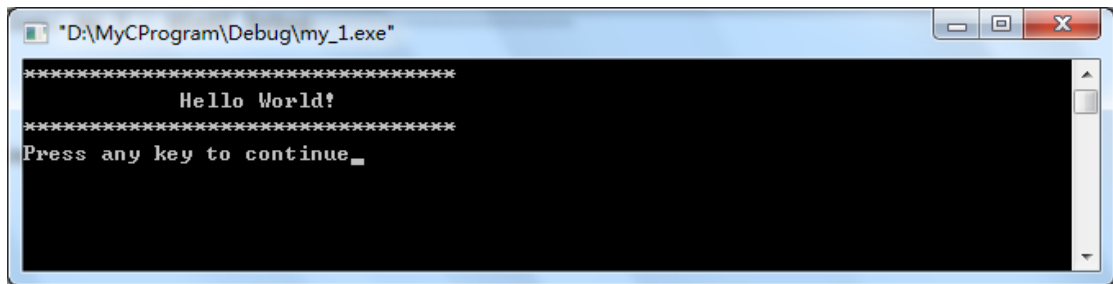


图 1-1 3 题运行结果

4. 编译、链接和运行程序，并记录结果。
5. 修改程序，实现下面功能：输入两个数，然后把这两个数的和、差显示在屏幕上，如下所示：

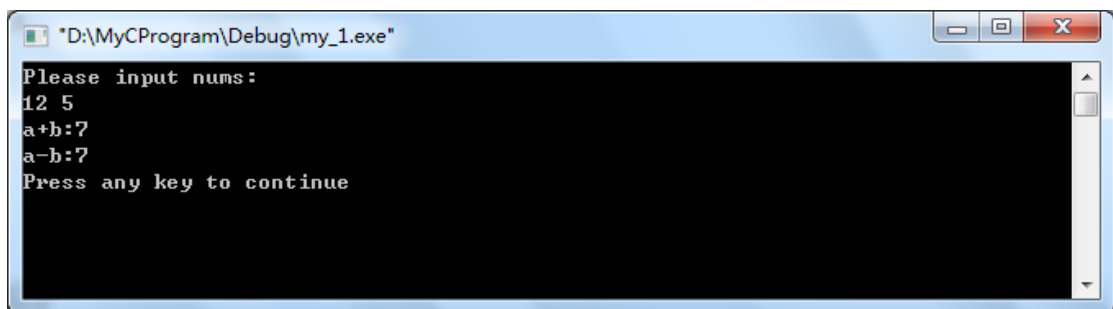


图 1-2 5 题运行结果

6. 单步调试程序：按下 F10 按钮，出现如下界面(如果不出现，请依次单击菜单：View->Debug Windows->Variables)，注意左侧的黄色箭头，该箭头指示当前程序即将要运行的代码行。

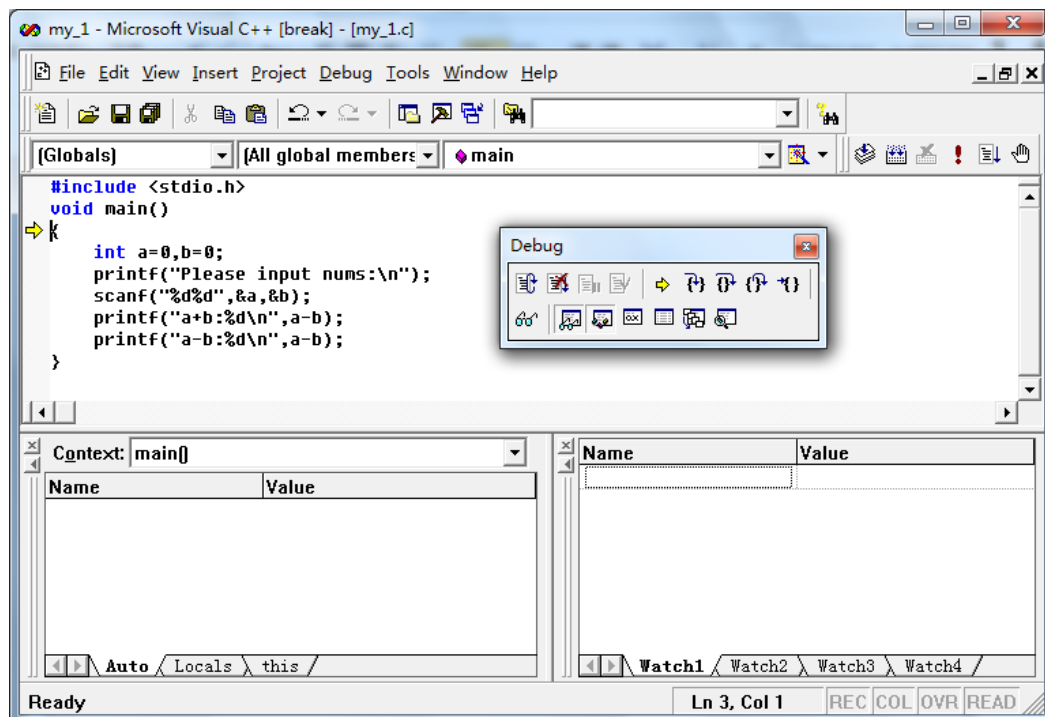


图 1-3 调试程序

7. 再次按下 F10，出现如下界面，注意图中下方窗口中所显示的当前变量值，请思考当前 a、b 的值为什么出现这样的结果，为什么提倡在定义变量的时候对其进行初始化。

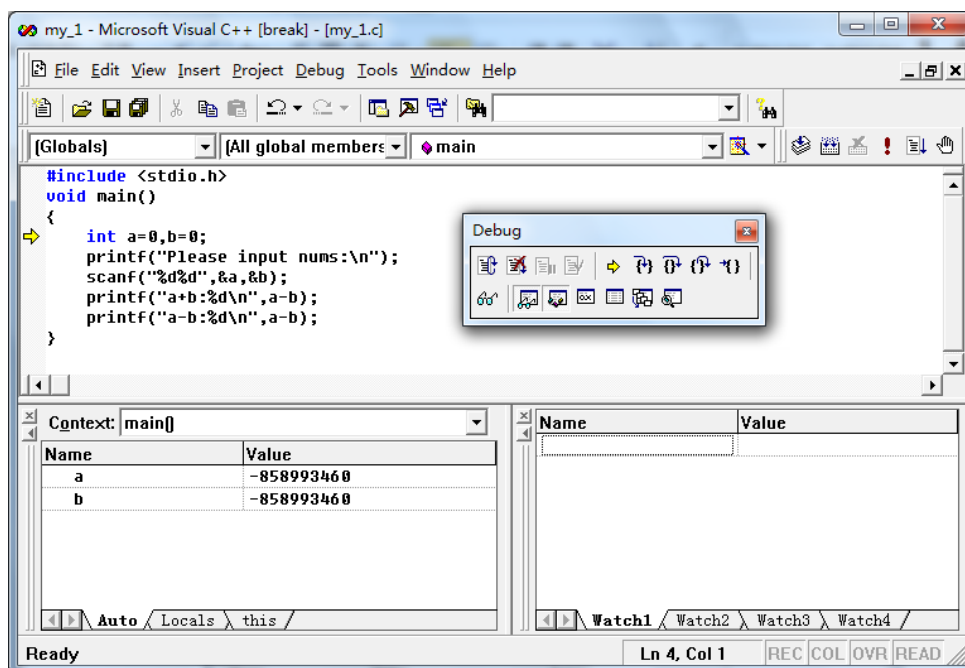


图 1-4 调试程序

8. 依次按下 F10,并记录 a、b 的值。
9. 当运行到 `scanf` 行时,按下 F10,注意到系统提示在程序中输入值。请转到命令提示符窗口中输入两个整数,如下所示:

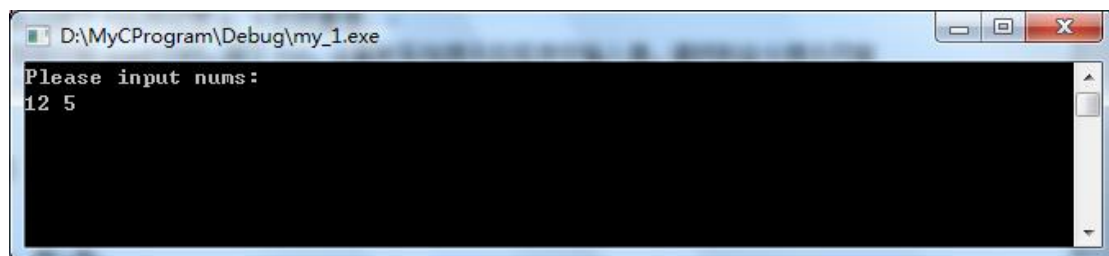


图 1-5 运行结果

10. 依次按下 F10, 并记录 a、b 的值, 并记录每步程序运行的结果。

### 1.3 分析和讨论

1. C 程序的构成
2. 变量的定义
3. 函数的定义, `main` 函数的写法和作用
4. `#include` 语句的作用和写法
5. 如何单步调试 C 程序? 如何观察变量值。

### 1.4 作业

1. 编写程序, 输入 a、b、c 三个整数, 输出其中最大者。
2. C 程序首先执行那个函数?

## 2 C 语言数据类型和运算

### 2.1 实验目的

- 掌握 C 语言数据类型，能够熟悉定义整型、浮点型、字符型等类型变量。
- 熟悉数据类型转换。
- 熟悉有关算术运算、算术表达式，特别是自增（++）和自减（--）运算符的使用。初步了解优先级概念。
- 进一步熟悉 C 语言特性，熟悉 VC 编程环境。

### 2.2 上机内容

1. 启动 VC6.0。请编程实现 p47 第 2 题中的第 4 中存法，算出存 n 次一年期的本息和。

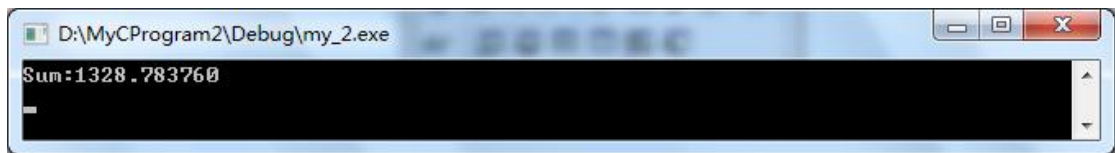


图 2-1 1 题运行结果

2. 编程实现简单凯撒密码的加密。即用户输入 5 个字母（原始文字，即明文）和密钥，显示其加密后的密文。如下图所示。

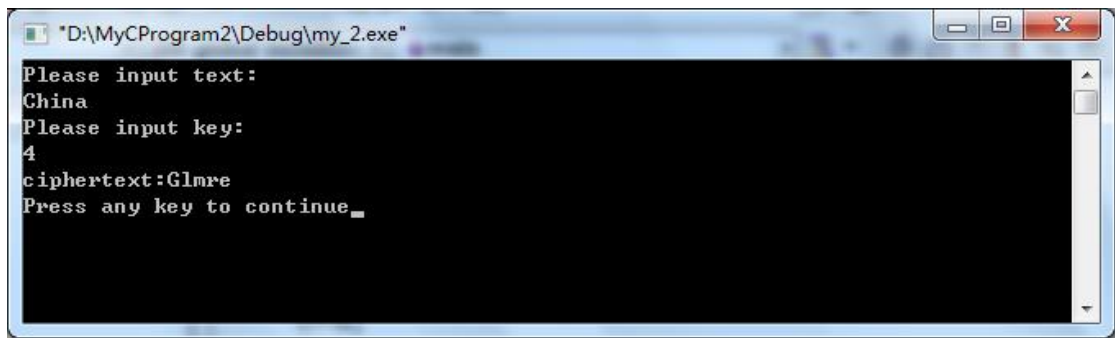


图 2-2 2 题运行结果

3. 编程实现简单凯撒密码的解密。即用户输入 5 个字母（即密文）和密钥，显示其解密后的明文。如下图所示。

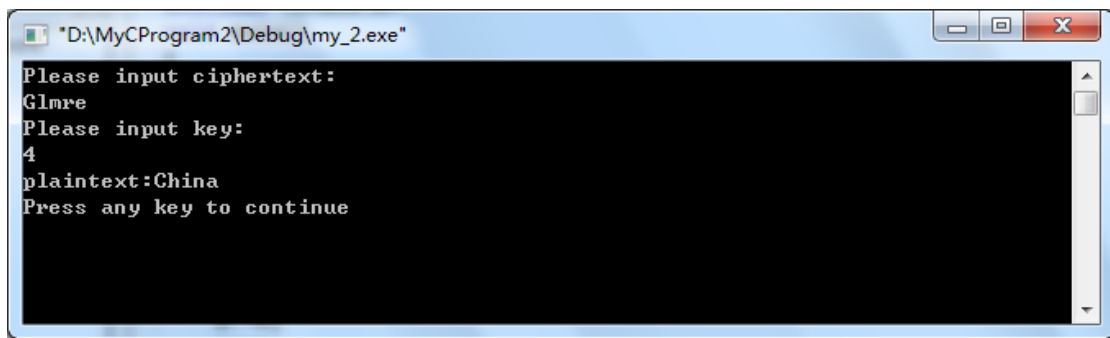


图 2-3 3 题运行结果

4. 请回答 p47 第 4 题。
5. 分析和运行程序，写出运行结果。

1) 分析和运行程序：

```
main()
{
    int i, j, m, n;
    i=8;
    j=10;
    m=++i;
    n=j++;
    printf("%d,%d,%d,%d", i, j, m, n);
}
```

2) 将第 6-7 行改为下面语句，先分析再运行：：

```
m=i++;
n= ++ j;
```

3) 在 2) 的基础上，将 printf 语句改为：

```
printf("%d,%d", ++ i, ++ j);
```

4) 再将 printf 语句改为：

```
printf( "%d,%d,%d,%d" , i, j, i++, j++);
```

5) 程序改为：

```
main()
{
    int i, j, m=0, n=0;
    i=8;
    j=10;
    m+= i ++;
    n -= --j;
    printf("i=%d,j=%d,m=%d,n=%d", i, j, m, n);
}
```

```
}
```

## 2.3 分析和讨论

1. `float` 和 `double` 区别和使用场合?
2. `int` 和 `char` 类型和什么情况下可以通用?
3. 自增和自减运算符放在变量前和后 (前缀和后缀) 有何区别?

## 2.4 作业

1. 求下面算术表达式的结果类型:`a+b*c-d`。其中 `a` 为 `float`, `b` 为 `int`, `c` 为 `char`, `d` 为 `double`.
2. 写出下面表达式运算后 `a` 的值, 设原来 `a=10`。设 `a` 和 `n` 已定义成整型(`int`)变量。

(1) `a+=a`

(2) `a-=2`

(3) `a*=2+3`

(4) `a/=a+a`

(5) `a%= (n%=2)`, `n` 的值等于 5

(6) `a+=a-=a*=a`

先自己分析, 再试着用程序求解, 看得到的结果是否一致。

## 3 简单的 C 程序设计

### 3.1 实验目的

- 了解算法概念，理解和掌握使用流程图、伪代码等方法表示算法。
- 掌握赋值运算符和表达式。
- 掌握 C 语言中输入输出的概念，学会使用 printf、scanf 等函数使用。
- 理解和掌握 C 语言程序中的顺序结构。

### 3.2 上机内容

1. 试使用流程图、伪代码表示出将 1 到 100 之间的所有的偶数和，即 $\sum_{i=1, i+=2}^{100} i$ 。
2. 使用 putchar、getchar 等实现将用户输入的三个小写字母转换成大写字母，如图所示。

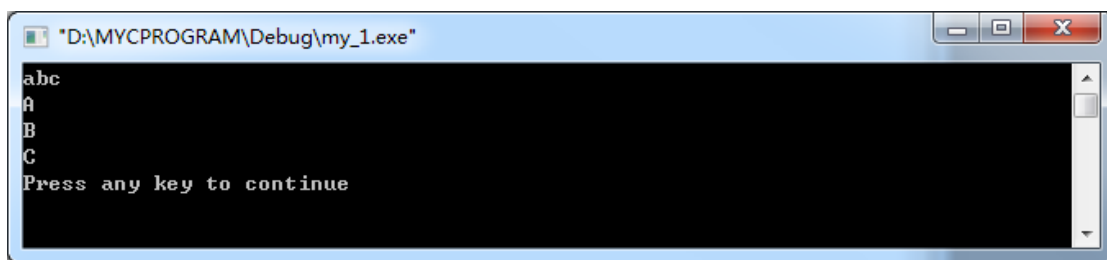


图 3-1 2 题运行结果

3. 使用 printf、scanf 等实现将用户输入的三个小写字母转换成大写字母及其 ASCII 码值，如图所示。

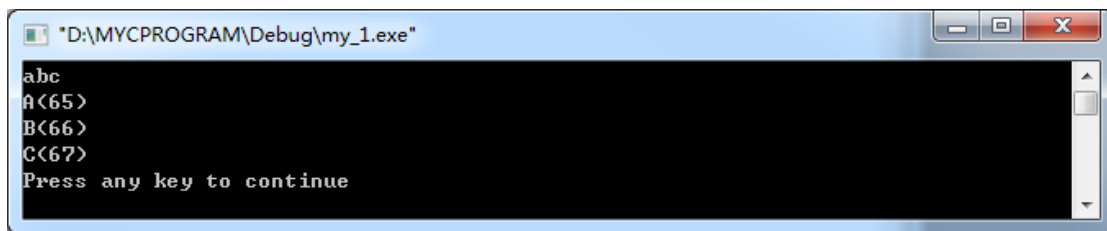


图 3-2 3 题运行结果

4. 输入一个 3 位数整数(即 100-999)，编程将其逆序输出，例如输入 123，输出 321。并用伪代码写出你的算法。

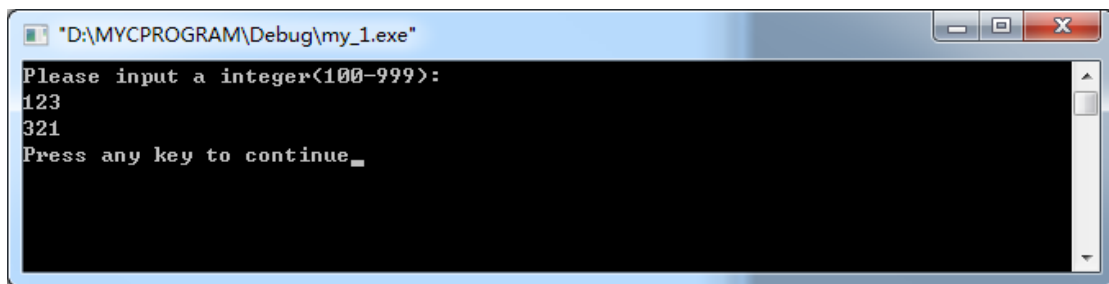


图 3-3 4 题运行结果

提示：利用整数除法和求余运算。

5. P87 页 5 题（其中  $r$  和  $h$  要求用 `scanf` 输入），公式如下：

圆周长：  $s = 2\pi r$

圆面积：  $a = \pi r^2$

圆球表面积：  $a = 4\pi r^2$

圆球体积：  $v = \frac{4}{3}\pi r^3$

圆柱体积：  $v = h\pi r^2$

### 3.3 分析和讨论

1. 程序的三种基本结构是什么？并用图形说明之。
2. 空格对应的 ASCII 码是 0 吗？空格和整数 0、字符 '0' 一样吗？简述之。
3. 下面代码有错误吗？如有请改正。

```
#include<stdio.h>
void main()
{
    Printf("Please input 2 nums:\n)
    scanf("a=%d%c",a,b),
    printf("sum:"a+b)
}
```

在上面代码中，如果要求必须按照下面的格式输入数据，请问如何写 `scanf` 语句？

- a)  $a=12, b=34$
- b) 12 34
- c) 12,34

### 3.4 作业

1. 输入一个正整数（范围仅限 33-126），求出该数所对应的字符。





## 4 选择结构程序设计

### 4.1 实验目的

- 理解选择结构的作用
- 掌握关系表达式和逻辑表达式，并能够正确运用与计算。
- 掌握 if、else、switch 等语句。

### 4.2 上机内容

1. 编程实现 *sign* 函数：

$$\text{sign}(x) = \begin{cases} 1(x > 0) \\ 0(x = 0) \\ -1(x < 0) \end{cases}$$

输入一个整数，如果该数大于 0，则输出 1，如果该数等于 0，则输出 0，否则输出-1。  
如图所示。

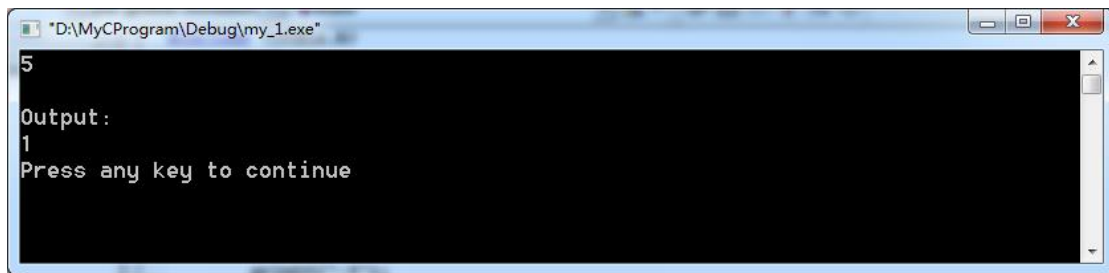


图 4-1 1 题运行结果

2. 给出一百分制成绩，根据成绩输出其等级，其等级标准如下：

- 90 分以上（包括 90）： A
- 80 至 90 分（包括 80）： B
- 70 至 80 分（包括 70）： C
- 60 至 70 分（包括 60）： D
- 60 分以下： E

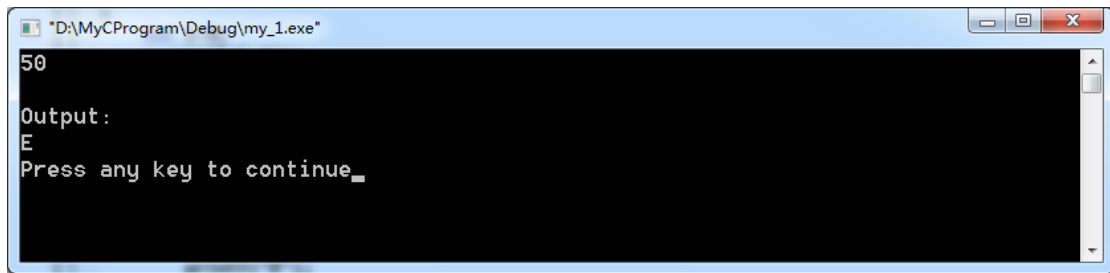


图 4-2 2 题运行结果

3. 上题请用 `switch` 语句实现。

提示：成绩为 `0~100`，如果一个一个的使用 `case` 语句去实现，则很繁复；可以先把成绩除以 `10`，这样输出范围就变成了 `0~10`，然后再使用 `case` 语句。

4. 某程序已经给出了下面的伪代码，请实现该程序。

```
1. 输入 3 个整数 a,b,c。
2. if a==0 then
    if b!=0 then
        print "一次方程"
    else if c!=0 then
        print "error"
    end if
else if b2-4ac==0 then
    print "两个相等的实根"
else if b2-4ac>0 then
    print "两个不等的实根"
else if b2-4ac<0 then
    print "两个共轭复根"
end if
```

5. 给出一个不多于 5 位的正整数，要求：

1. 求出它是几位数；
2. 分别输出每一位数字

3. 逆序输出，例如原数为 123，则其逆序输出为 321

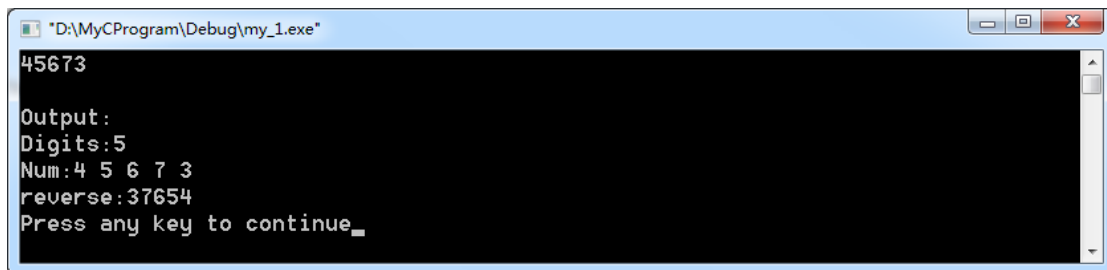


图 4-3 3 题运行结果

### 1.1 分析和讨论

1. 有如下代码：

```
#include <stdio.h>
void main()
{
    int a=1;
    scanf("%d",&a);
    if(a!=0)
        printf("yes");
    else if(a==0)
        printf("no");
    else
        printf("no");
}
```

- 上述代码是否正确？
  - 代码中第一个 if 语句（即第 6 行）能否改写成:if(a)？
  - 上述代码能否再进一步简化？
2. c 语言如何表示“真”与“假”？系统如何判断一个量的“真”与“假”？

## 4.3 作业

- 编写程序，输入 a、b、c 三个整数，按照从小到大顺序输出。
- 写出下面各逻辑表达式的值。设 a=3,b=4,c=5（在计算的时候要考虑运算符的优先级问题）
  - $a+b>c \&\&b = c$
  - $a || b+c \&\&b-c$
  - $! (a>b) \&\&!c || 1$

(4)  $!(x=a)\&\&(y=b)\&\&0$

(5)  $!(a+b)+c-1\&\&b+c/2$

## 5 循环结构程序设计

### 5.1 实验目的

- 掌握循环结构特点，能够正确的使用循环结构，掌握 do...while、while、for 语句。
- 正确地设定循环条件，控制循环次数
- 掌握学会循环嵌套。

### 5.2 上机内容

1. 使用循环结构计算  $8!$
2. 使用循环结构实现  $s = 1! + 2! + 3! + \dots + 8!$ 。
3. 使用循环结构实现  $s = 1! + 3! + 5! + 7!$ 。
4. 求数列  $s = \frac{1}{2} + \frac{2}{3} + \frac{3}{4} + \dots + \frac{99}{100}$ 。
5. 求  $S_n = a + aa + aaa + \dots + \underbrace{a \dots a}_n$  之值，其中  $a$  是一个数字。例如：  
 $2+22+222+2222+22222$ （此时  $a = 2, n = 5$ ）， $a, n$  均由键盘输入。
6. 输出所有的“水仙花数”，详见 p137。

### 5.3 分析和讨论

1. while 语句是否可以改写为 for 语句？如果可以，请改写下面程序。

```
#include <stdio.h>
void main()
{
    int sum=0,i=0;
    while(i<10)
    {
        i++;
        sum+=i;
    }
    printf("%d",sum);
}
```

2. for 语句中的表达式 1、表达式 2、表达式 3 可否为空？
3. 简述 break、continue 区别，请用程序说明之。

### 5.4 作业

1. 一个数如果恰好等于它的真因子（即除了自身以外的约数，包括 1）之和，这个数就称为完全数(Perfect number)。第一个完全数是 6，它的真因子为 1,2,3， $6=1+2+3$ 。求 1000 之内的所有完全数。

## 6 数组

### 6.1 实验目的

- 掌握一维数组的使用，包括定义、初始化、输入输出等。
- 掌握二维数组的使用，包括定义、初始化、输入输出等。
- 掌握字符数组、字符串的使用。
- 掌握与数组有关的算法（例如冒泡排序等）

### 6.2 上机内容

1. 已知一个班上 10 个学生的《高等数学》成绩，要求输入这 10 个学生的成绩，存放在一个一维数组中：
  - a) 求出平均成绩。
  - b) 要求找出成绩最高的学生的成绩和该生的序号。
  - c) 使用冒泡排序：对成绩按照降序进行排序，即成绩高的在前，低的在后。
2. 有三个学生，上高等数学、英语、程序设计基础、软件工程导论等 4 门课，要求输入全部学生的各门课成绩，然后
  - a) 分别求出每门课的平均成绩。
  - b) 求出平均成绩最高的学生的序号和各门课成绩
  - c) 按照平均成绩排序输出，平均成绩最高的在最前，平均成绩最低的在最后。
3. 一个已排好序（降序）的数组，今输入一个数，要求按原来排序的规律将它插入到数组中。

如：1 2 4 6 7 9 输入 5 。

插入到数组中后：1 2 4 5 6 7 9

4. 用户首先输入一个字符串，然后统计该字符串中英文大写字母、英文小写字母、数字、空格以及其它字符各有多少个。
5. 用户输入 5 个字符串后，按照升序次序输出。

### 6.3 分析和讨论

1. 在什么情况下使用数组？
2. 简单分析冒泡排序算法。如果对  $n$  个数进行排序，则需要进行几趟比较，每趟需要进行几次两个数的比较。你能想出其它的排序算法吗？
3. 简述“打擂台”算法。



4. 试比较字符数组和字符串的异同。
5. 简单总结一下 C 函数库中的字符串处理函数的使用方法。

#### 6.4 作业

1. 将一个数组中的值按照逆序重新存放。例如原来顺序为 8、6、5、4、1，要求改为 1、4、5、6、8。
2. 字符串编程。编写一个程序，将两个字符串连接起来。1) 用 `strcat` 函数；2) 不用 `strcat` 函数

## 7 函数

### 7.1 实验目的

- 掌握函数的原型声明、定义。
- 掌握函数调用，包括嵌套调用、递归调用等。
- 掌握调用函数时参数传递方式。

### 7.2 上机内容

1. 写一个判断素数的函数，在主函数输入一个整数，输出是否素数的信息。

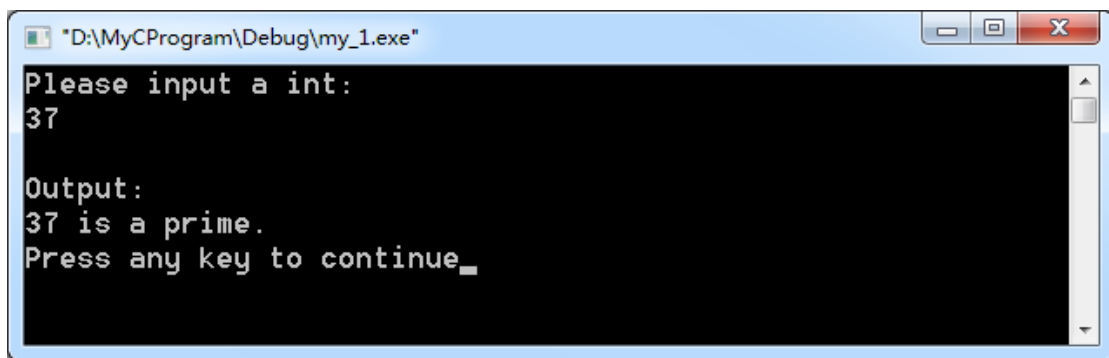


图 7-1 1 题运行结果

2. 编写一个函数（函数名为 **reverse**），使输入的字符串按反序存放。例如输入 **HELLO**，则输出 **OLLEH**。要求在主函数 **main** 中输入和输出字符串。

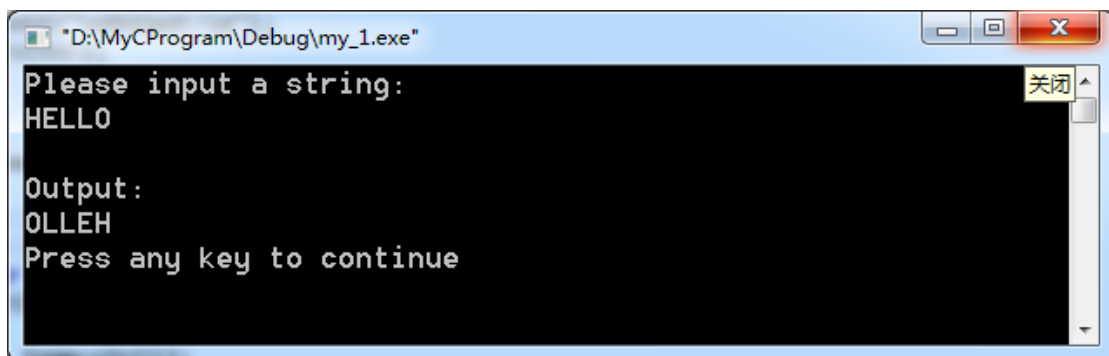


图 7-2 2 题运行结果

3. 下面以例子来说明 **VC6** 中如何设置断点、观察变量。  
实现最小公倍数和最大公约数的计算。

解题思路:

两个数  $m, n$  的最小公倍数最大可能值是两个数的成绩  $m*n$ ，最小可能值是  $\max(m, n)$ 。

两个数  $m, n$  的最大公约数的最大可能值是  $\min(m, n)$ ，最小值是 1。我们可以根据这个思路使用循环语句实现。

键入以下代码:

```
#include<stdio.h>
int GCD(int m,int n);
int LCM(int m,int n);
void main()
{
    int x=0,y=0;
    int r=0;
    printf("input 2 int:\n");
    scanf("%d%d",&x,&y);

    printf("\nOutput:\n");

    r=GCD(x,y);
    printf("GCD:%d\n",r);

    r=LCM(x,y);
    printf("LCM:%d\n",r);
}

int GCD(int m,int n)
{
    int i;
    for(i=m<n?m:n;i>0;i--)
        if(m%i==0 && n%i==0)
            return i;
    return 1;
}

int LCM(int m,int n)
{
    int i;
    for(i=m>n?m:n;i<m*n;i++)
        if(i%m==0 && i%n==0)
            return i;
    return m*n;
}
```

设置断点：所谓断点是指在调试状态下程序运行到断点时会暂停，这时用户可以观察变量。首先将光标移动到将要设置断点的行，然后按下 **F9** 或点击工具栏上的“小手”图标。可以看到 VC 中代码窗口左侧出现了一个棕色的原点。如图所示：

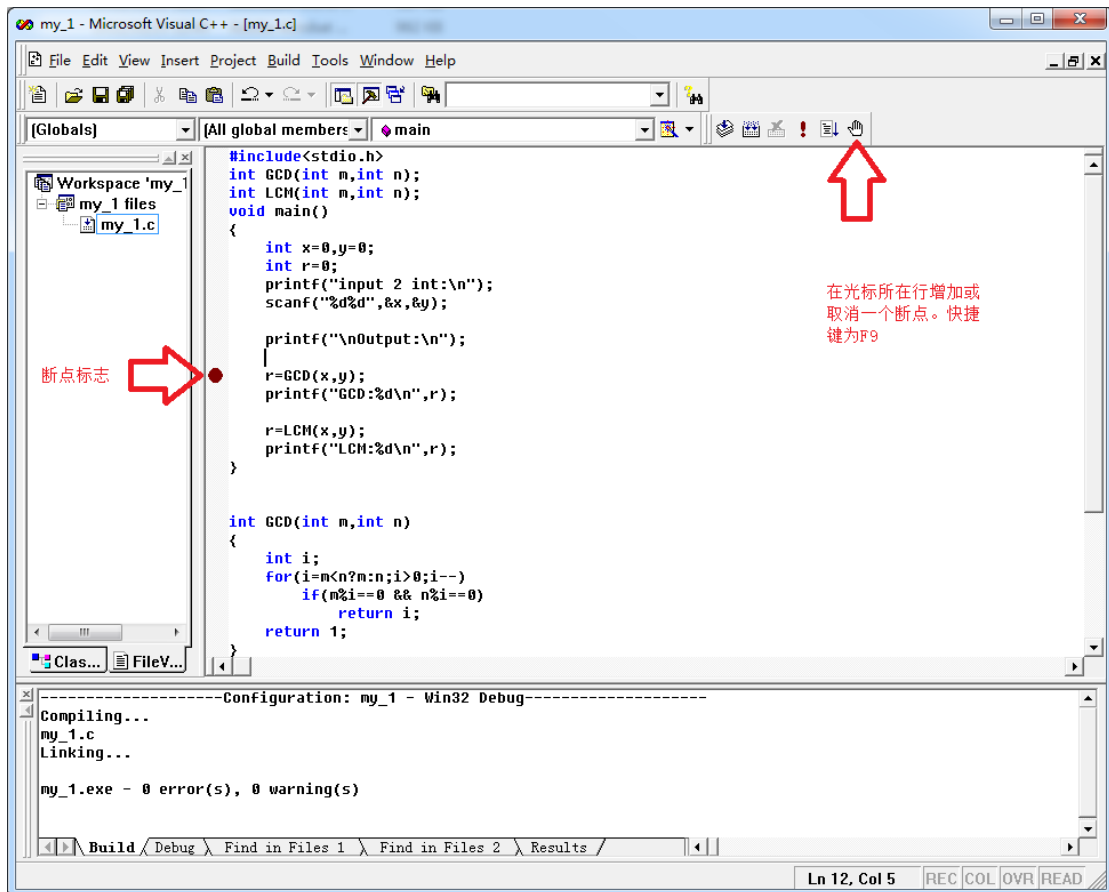


图 7-3 设置断点

执行程序（按下 **F5** 或点击），可以看到，程序运行到该行时暂停，如下图所示，注意断点棕色圆形上的黄色箭头，表示程序即将要执行该行。

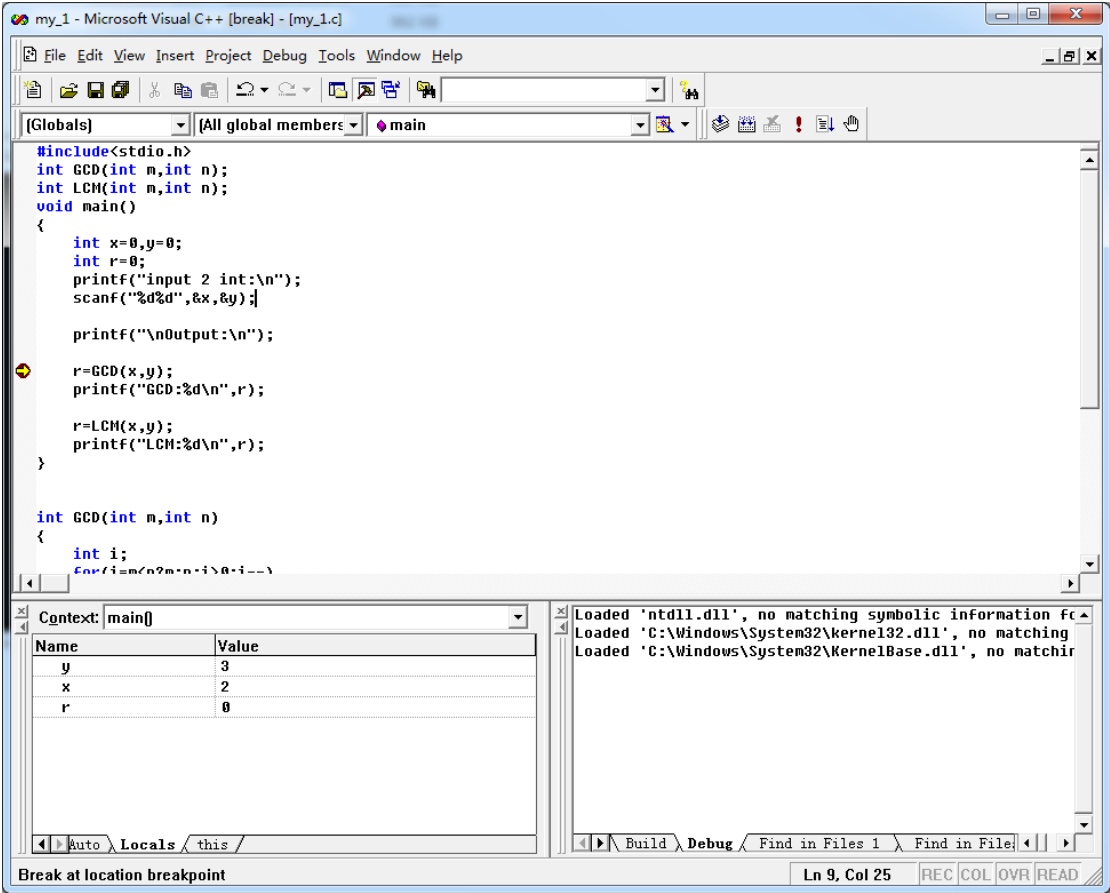


图 7-4 程序暂停

下面观察变量值。在上图中左下方的变量窗体（如果不出现，请依次单击菜单：View->Debug Windows->Variables）中，可以看到 x 为 2，y 为 3。在该窗口底部有三个标签栏：Auto 表示自动列出当前上下文中的变量值，Locals 表示列出局部变量值，this 现在不用关心。

现在程序即将要调用函数 GCD，如果按下 F10，则 VC 不会进入 GCD 函数内部，而是直接将函数调用完毕后将结果返回，然后函数返回值赋给 r。有的时候我们想跟进到函数内容，我们选择按下 F11。

表 7-1 VC6.0 中快捷键小结

| 快捷键 | 说明                      |
|-----|-------------------------|
| F5  | 运行程序                    |
| F9  | 设置/取消断点                 |
| F10 | 逐行执行程序，遇到函数调用时不会进入到函数内容 |
| F11 | 逐行执行程序，遇到函数调用时进入到函数内部   |

按下 F11 后，观察到程序流程进入到了 GCD 内部。

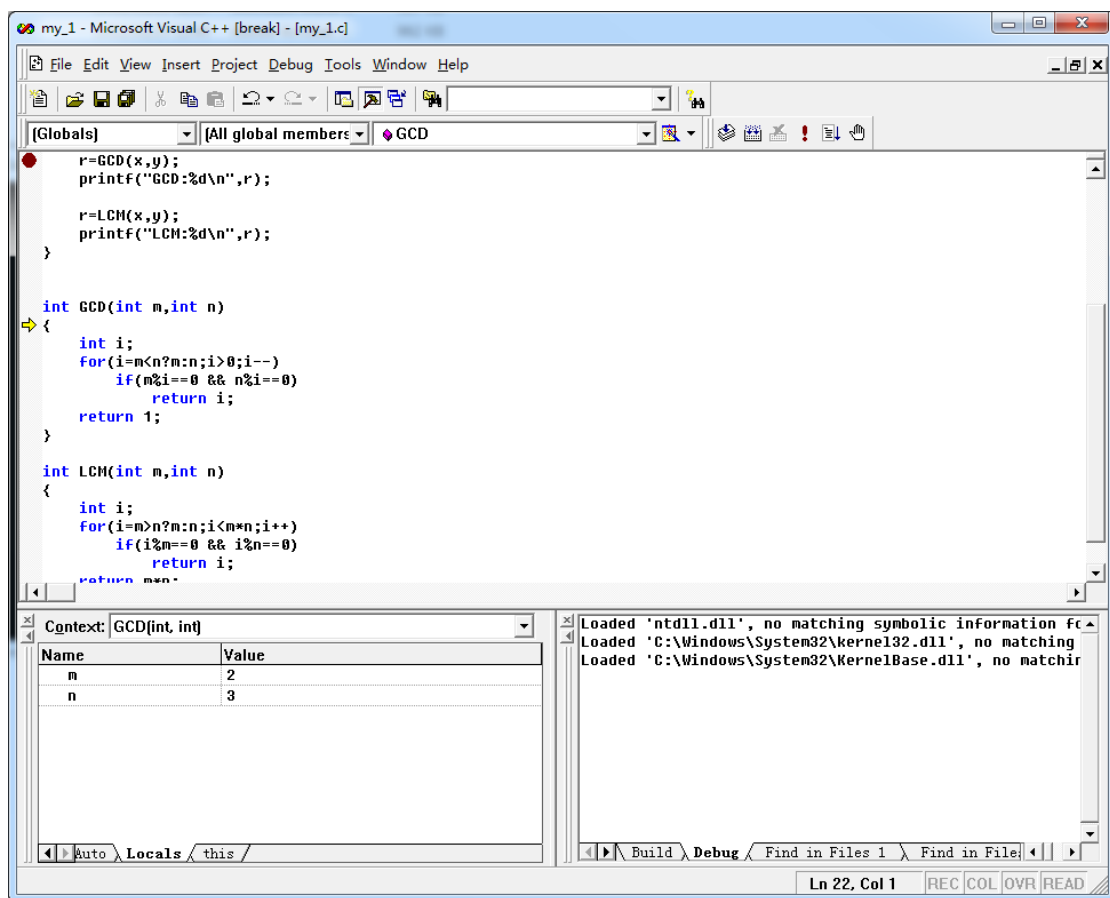


图 7-5 调试程序

现在可以看到，`m` 值为 2，`n` 为 3。即此时函数参数传递的方式是单向值传递，`m` 的值为 `x` 的值，`n` 的值为 `y` 的值。

一直按 `F10`，直到程序返回到 `main` 中，如下图所示：



可以观察到 **x** 的值和 **y** 的值仍然为原来的值，而没有发生变化。这就是所谓的**传值**。按下 **F10**，程序将 **GCD** 函数的返回值赋给 **r**。如下图所示：

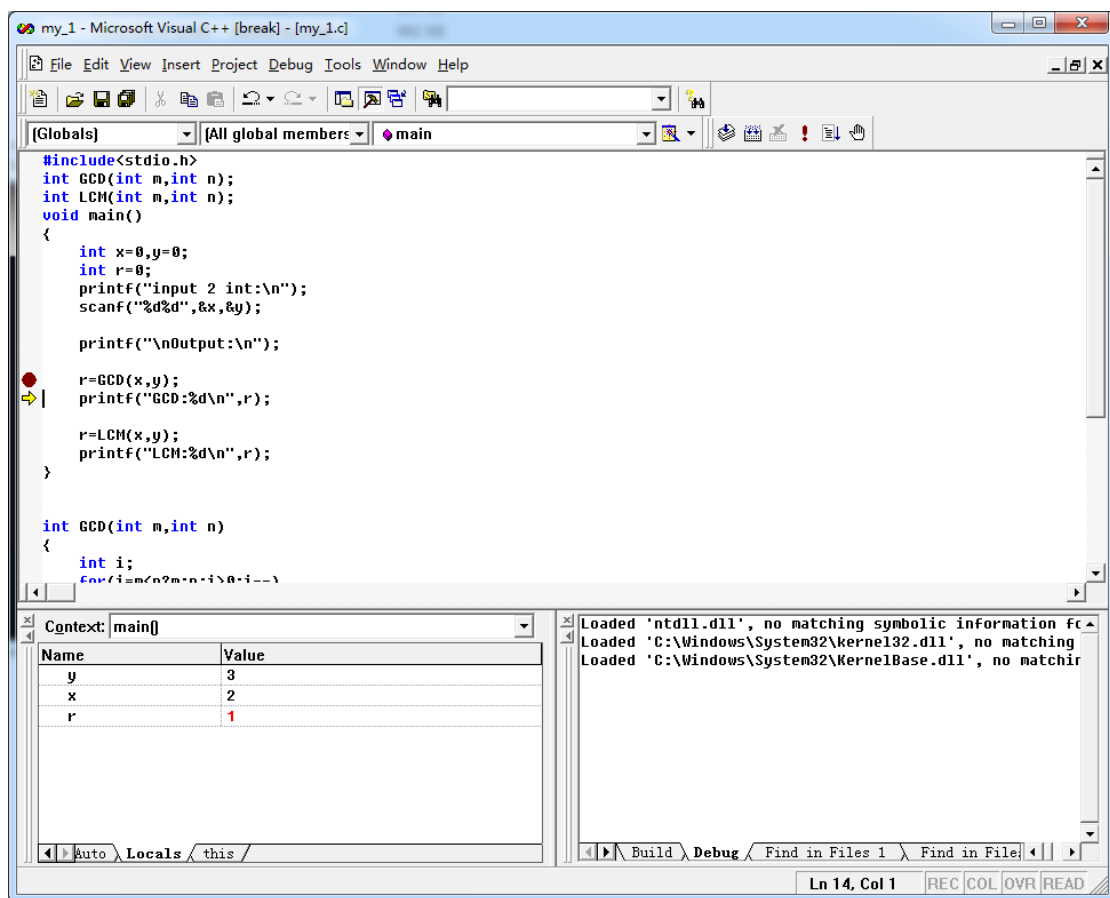


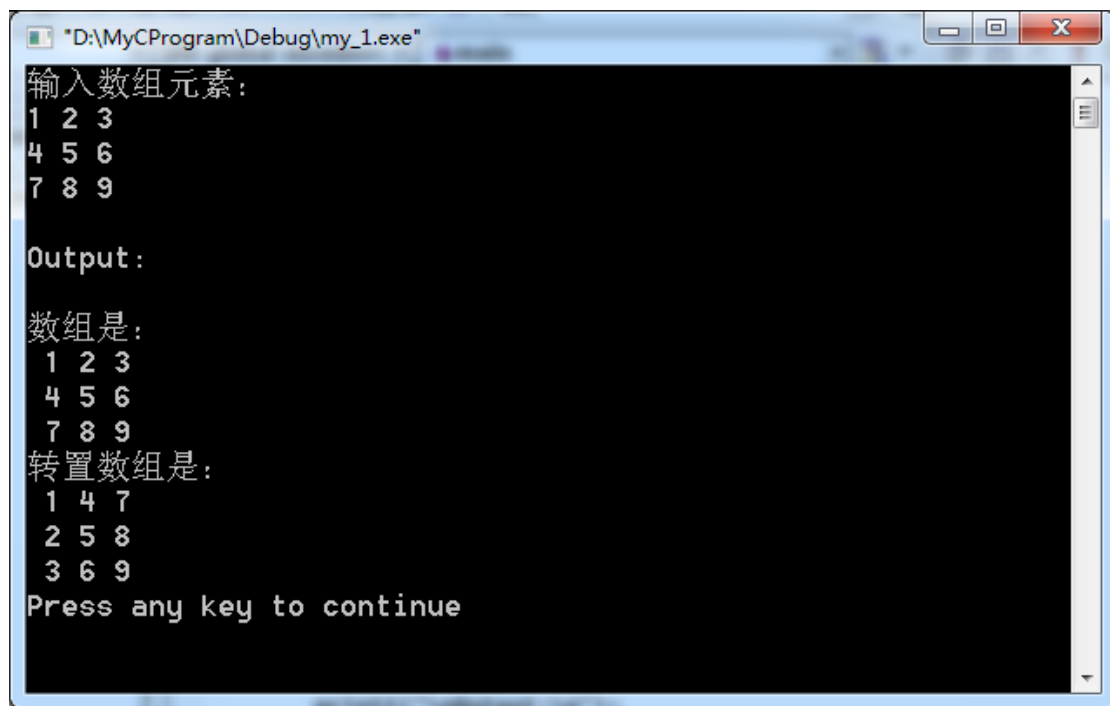
图 7-7 调试程序

注意到 `r` 的值由原来的 0 变为了 1，1 使用红色表示，表示值发生了变化，提醒程序员注意。

下面调用 LCM 函数的过程中如何借助 VC6 提供的功能观察变量值、进入函数等，请记录各个变量值。

4. 写一函数，是给定的一个二维数组（ $3 \times 3$ ）转置，即行列互换。



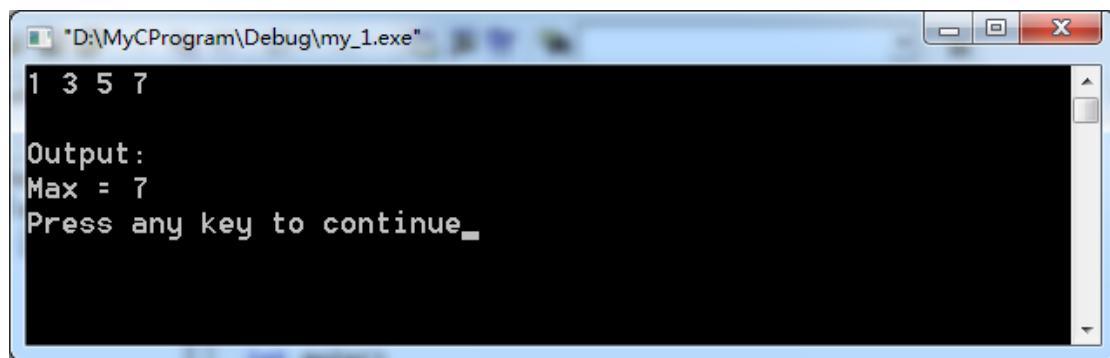


```
"D:\MyCProgram\Debug\my_1.exe"
输入数组元素:
1 2 3
4 5 6
7 8 9

Output:
数组是:
1 2 3
4 5 6
7 8 9
转置数组是:
1 4 7
2 5 8
3 6 9
Press any key to continue
```

图 7-8 4 题运行结果

5. 输入 4 个整数，找出其中最大的数，使用函数的递归调用来处理，并记录函数各变量值。



```
"D:\MyCProgram\Debug\my_1.exe"
1 3 5 7

Output:
Max = 7
Press any key to continue_
```

图 7-9 4 题运行结果

### 7.3 分析和讨论

1. 变量的作用域和生存期指的是什么，请用例子说明之。
2. 如何向函数传递整个数组？
3. 函数原型有几种形式，应该写在什么位置？

### 7.4 作业

1. 给出年、月、日，计算该日是该年的第  $n$  天。



## 8 指针

### 8.1 实验目的

- 掌握指针的概念，地址即指针，指针即地址。
- 掌握指针变量的概念，并会定义和使用指针变量。
- 学会指针变量作为函数参数的使用方法。
- 正确使用数组的指针和指向数组的指针变量。
- 正确使用字符串的指针和指向字符串的指针变量。

### 8.2 上机内容

本节所有实验内容均要求使用指针方法实现。

1. 写一函数，把 ASCII 码为奇数且为小写的字符转换为大写字符（通过指针变量作为函数参数来实现）。
2. 写一函数，求字符串的长度。在 main 函数中输入字符串，并输出其长度。（不要使用 strlen 函数）。
3. P247 第 12 题。
4. 写一函数，将一个 3\*3 的整型二维数组转置，即行列互换。

### 8.3 分析和讨论

1. 两个指针变量相加、减有意义吗？如果有意义，请解释之。
2. 指针变量加上或减去一个整数有意义吗？如果有意义，请解释之。
3. 指针变量能否有空值？其含义是什么？怎样表示？
4. 指针和指针变量是一回事吗？请解释之。
5. 数组和指针什么关系？请举例说明如何使用指针指向数组元素。
6. 函数参数可以是指针吗？

### 8.4 作业

1. P247 第 4 题。

## 9 结构体

### 9.1 实验目的

- 掌握结构体的概念和使用
- 掌握结构体数组的概念和使用。
- 使用 `malloc` 和 `free` 等动态申请和释放内存

### 9.2 上机内容

1. 有三个学生，每个学生信息包括姓名、年龄、专业、班级、高等数学成绩、英语成绩、程序设计基础成绩等，编写一个程序，实现这三个学生信息的输入与输出，使用结构体数组实现。
2. 改写上面程序，但学生的人数由用户输入获得，使用链表实现。

### 9.3 分析和讨论

1. 简述使用链表实现和使用结构体数组实现上机内容的异同。
2. 如何在链表中加入、删除、移动、修改节点？

### 9.4 作业

2. 改写 1.2 中的第 2 题，统计总分不及格人数及名单。





## A. 附录





## 参考文献

当前文档中没有源。

## 索引

未找到索引项。