**辽 宁 大 学 信息 学 院 实 验 报 告**

|  |  |
| --- | --- |
| **实验题目：** | **实验五 函数** |
| **专业班级：** | 19级计算机6班 |
| **学号（序号）：** | 191491603（39） |
| **姓 名：** | 曹宇宸 |
| **实验时间：** | 2020年5月22日 |

1. **实验目的**

通过本实验，使学生掌握多个函数之间数据传递的基本方法，充分利用函数，实现程序的模块化。

2．**实验要求**

* 1. 掌握定义函数的方法；
  2. 掌握通过值传递调用函数的方法；
  3. 掌握递归函数的设计方法；
  4. 掌握几个函数放在同一个工程中的不同源程序文件中的程序设计方法。

3．**实验内容**

运用函数编写应用程序，上机调试并写出实验报告。

1. 编写程序计算2和-3的0到5次幂。

具体要求：

①      编写一个power(int x,int n)函数计算x的n次幂；

    ​power函数的形式为：

    ​int power(int x,int n)

    ​ {

    ​编写代码实现求x的n次幂;

    ​  }

②     在主函数中多次调用power函数，得到所要求的结果。

int main()

{

int i;

for(i=0;i<=5;i++)

{

printf("2 raised to the power %d: %-6d ",i, power(2,i));

printf("-3 raised to the power %d: %d \n",i, power(-3,i));

}

}

**【答】源程序：**

#include<stdio.h>

#include<math.h>

int power(int x, int n)

{

if (n == 0) return 1;

else

{

int result = x;

int i = 0;

for (i = 1; i < n; i++) result = result \* x;

return result;

}

}

int main()

{

int i = 0;

for (i = 0; i <= 5; i++)

{

printf("2 raised to the power %d: %-6d ", i, power(2, i));

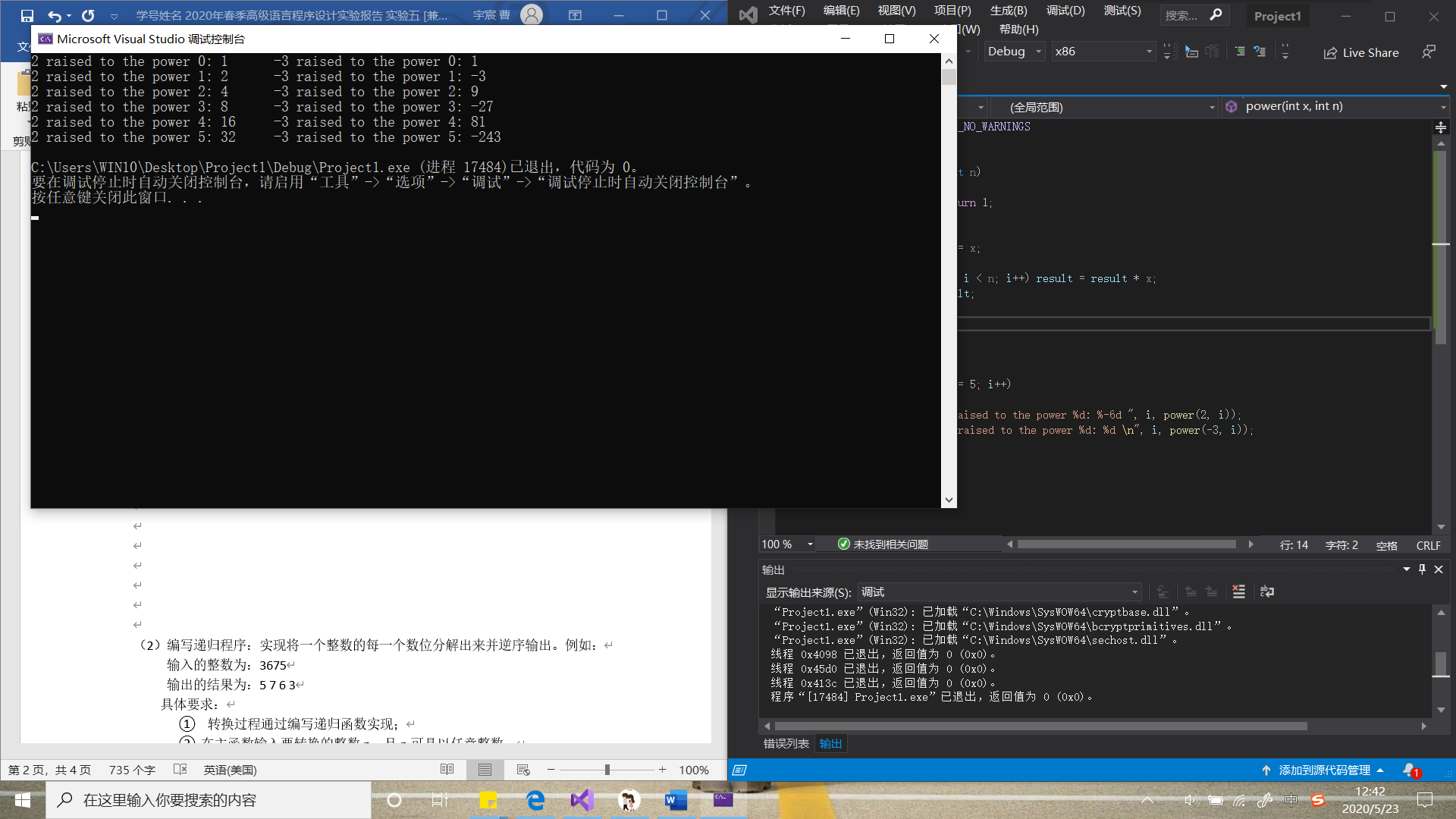
printf("-3 raised to the power %d: %d \n", i, power(-3, i));

}

return 0;

}

**【答】该程序运行结果：**



（2）编写递归程序：实现将一个整数的每一个数位分解出来并逆序输出。例如：

输入的整数为：3675

输出的结果为：5 7 6 3

具体要求：

1. 转换过程通过编写递归函数实现；
2. 在主函数输入要转换的整数n，且n可是以任意整数。

**【答】源程序：**

#include<stdio.h>

int revered(int x)

{

int y=0;

if (x % 10 == 0)

y = x;

else

y = x % 10;

return y;

}

int main()

{

int n;

int c;

scanf("%d", &n);

if (n == 0)

printf("%d ", n);

else

while (n)

{

c = revered(n);

n /= 10;

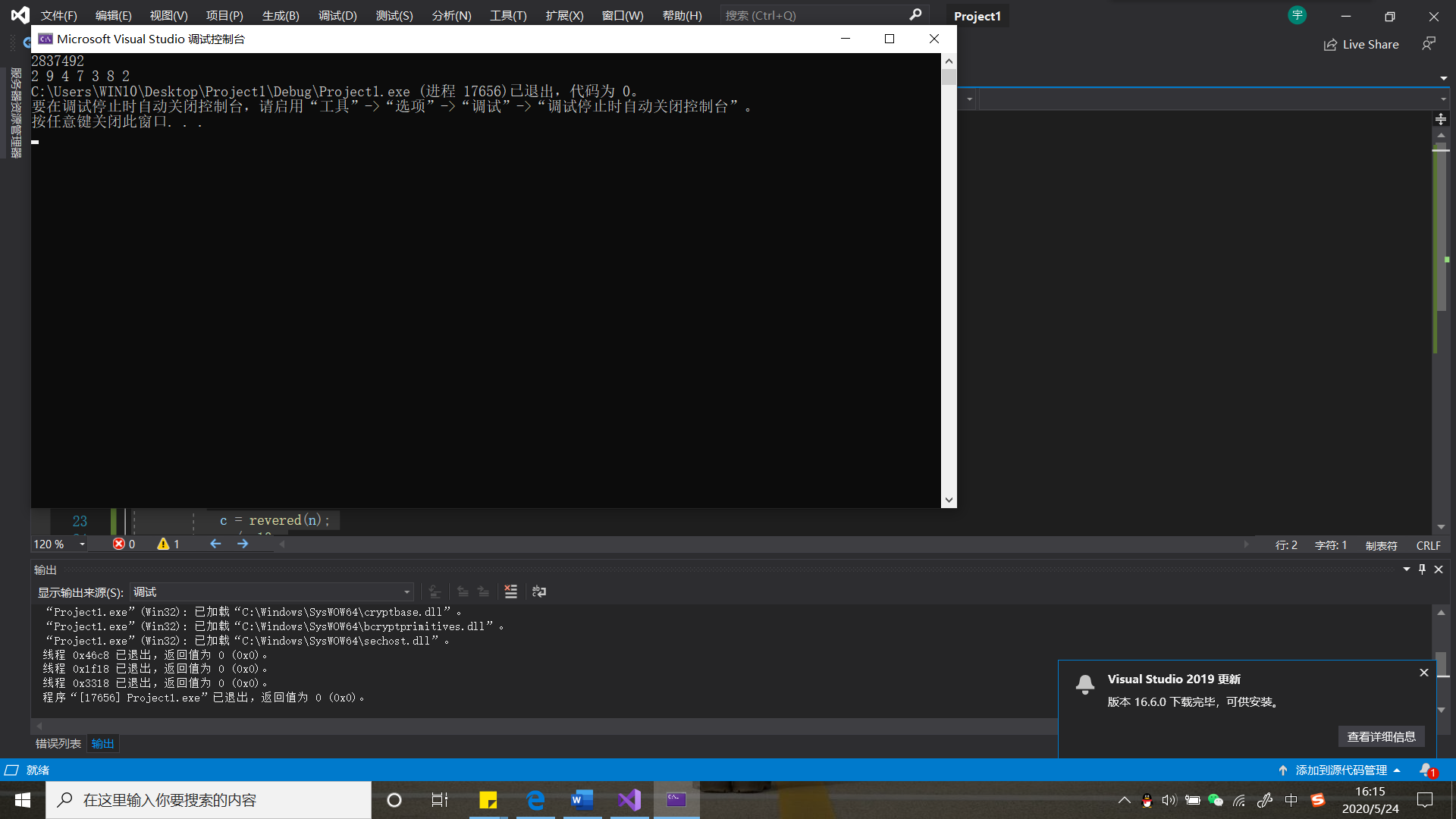
printf("%d ", c);

}

return 0;

}

**【答】该程序运行结果：**



（3）编写程序求两个整数的最大公约数和最小公倍数。

具体要求：

1. 编写两个用户自定义函数分别求出最大公约数和最小公倍数；
2. 将main函数和两个自定义函数放在同一个源程序文件(.c)中进行编译、链接并运行；
3. 将main函数和两个自定义函数分别放在不同的源程序文件(.c)中进行编译、链接并运行；

**【答】源程序：**

#include<stdio.h>

int Maximum\_number\_of\_conventions(int num1, int num2)//最大公约数

{

int max = 0;

for (int i = 1; i < num1; i++)

if (num1 % i == 0 && num2 % i == 0)

max = i;

return max;

}

int Minimum\_common\_multiple(int num1, int num2)

{

int max = Maximum\_number\_of\_conventions(num1, num2);

int min = 0;

min = num1 \* num2 / max;

return min;

}

int main()

{

int x = 0, y = 0;

scanf("%d%d", &x, &y);

int max = 0;

max=Maximum\_number\_of\_conventions(x, y);

int min = 0;

min = Minimum\_common\_multiple(x, y);

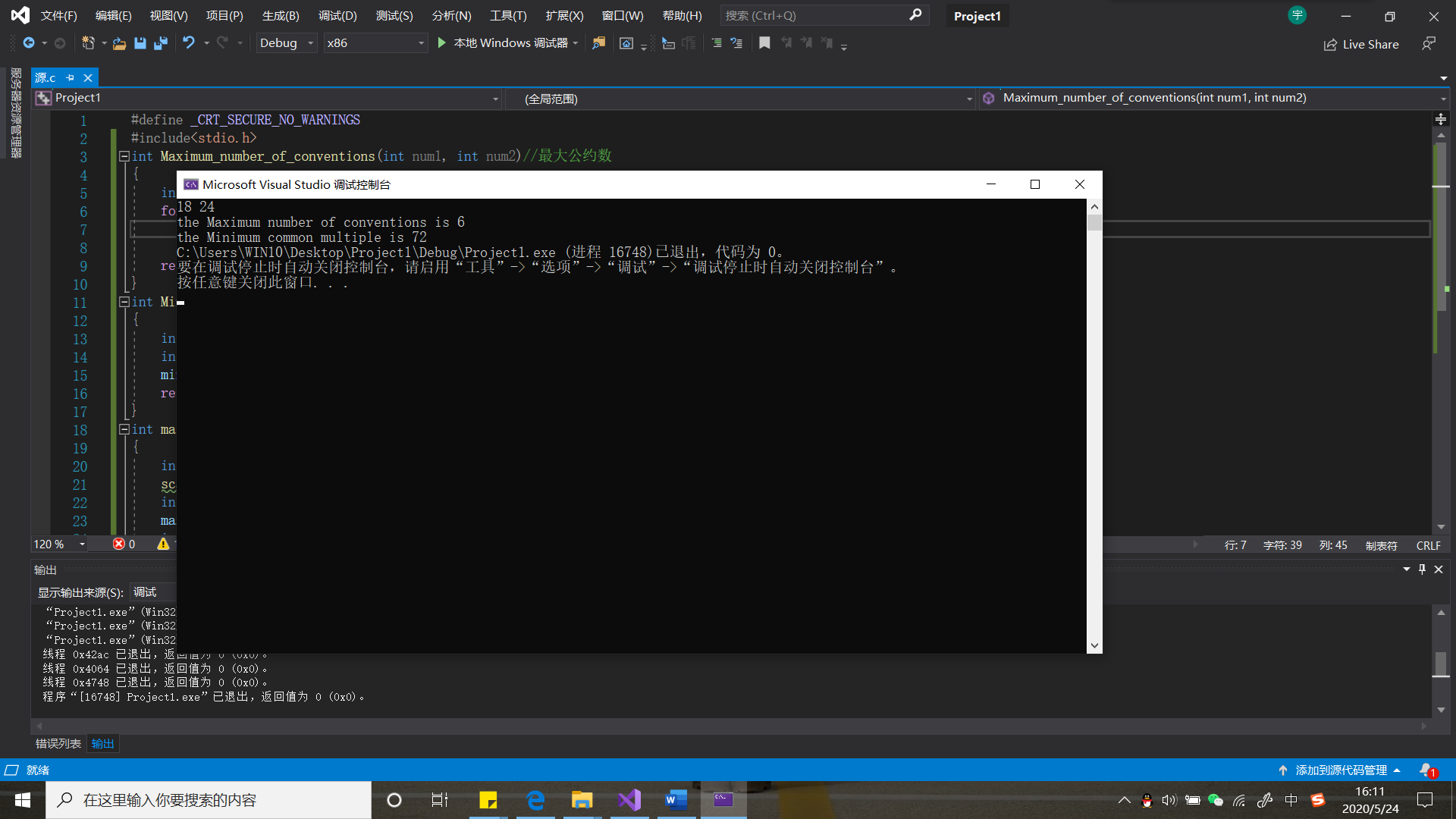
printf("the Maximum number of conventions is %d\n", max);

printf("the Minimum common multiple is %d", min);

return 0;

}

**【答】该程序运行结果(一个源文件)：**



**【答】该程序运行结果(两个或三个源文件)：**

