

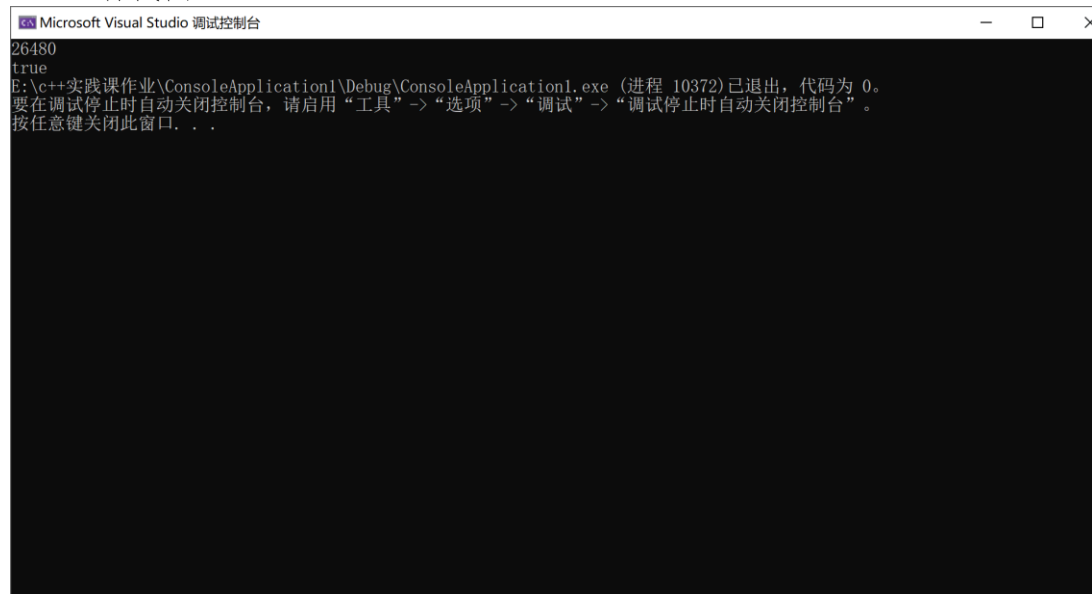
课程名称	高级语言程序设计		实验名称	高级语言程序设计实践	
班级		姓名		学号	
仪器组号			实验日期		
实验报告要求：1.实验目的 2.实验内容（题目描述，源代码，运行截图，调试情况） 3.实验体会					
<div> <div>一、实验目的</div> <div> 1. 认识了解 c++函数语法 2. 掌握条件语句和循环语句 </div> <div>二、实验内容</div> <div> 1、第一题 <div>1.1 题目描述</div> <div> <div>A. 问题描述</div> <div>编写具有如下原型的函数：<code>bool f(long x)</code>;其功能为：若整数 <code>x</code> 仅由偶数字（0、2、4、6、8）组成时（如 <code>x=26480</code>），函数返回 <code>true</code>，否则返回 <code>false</code>（如当 <code>x=22034</code> 时）。并编制主函数对它进行调用。</div> <div>B. 输入</div> <div>只能是一个整数数字，中间不能有空格等其他字符</div> <div>C. 输出</div> <div>只输出 <code>true</code> 或者 <code>false</code>，后面不要换行</div> </div> <div>1.2 源代码</div> <div> <pre> #include <iostream> using namespace std; bool fun(int n) { int i = 0; while (n) { i = n % 10; n /= 10; if (i == 0 i % 2 == 0) ; else return false; } return true; } </pre> </div> </div> </div>					

```

int main()
{
    int x;
    cin >> x;
    if (fun(x))
        cout << "true";
    else
        cout << "false";
    return 0;
}

```

1.3 运行截图



1.4 调试情况

2、第二题

2.1 题目描述

A. 问题描述、

编写具有如下原型的函数：`void find(int i, int n);` 实现从 `i` 开始找起，连续找出 `n` 个素数并显示在屏幕上。如：实参为 10 和 8 后，输出的 8 个素数应为：11, 13, 17, 19, 23, 29, 31, 37。

B. 输入

输入两个整数，前一个是查找的起始数字，后一个是指定查找几个素数。

C. 输出

输出指定数量的素数，每个素数后面加一个空格作为分隔符，最后一个数字后面可以有一个空格，但不需要换行。

2.2 源代码

```

#include <iostream>
using namespace std;

void find(int i, int n)

```

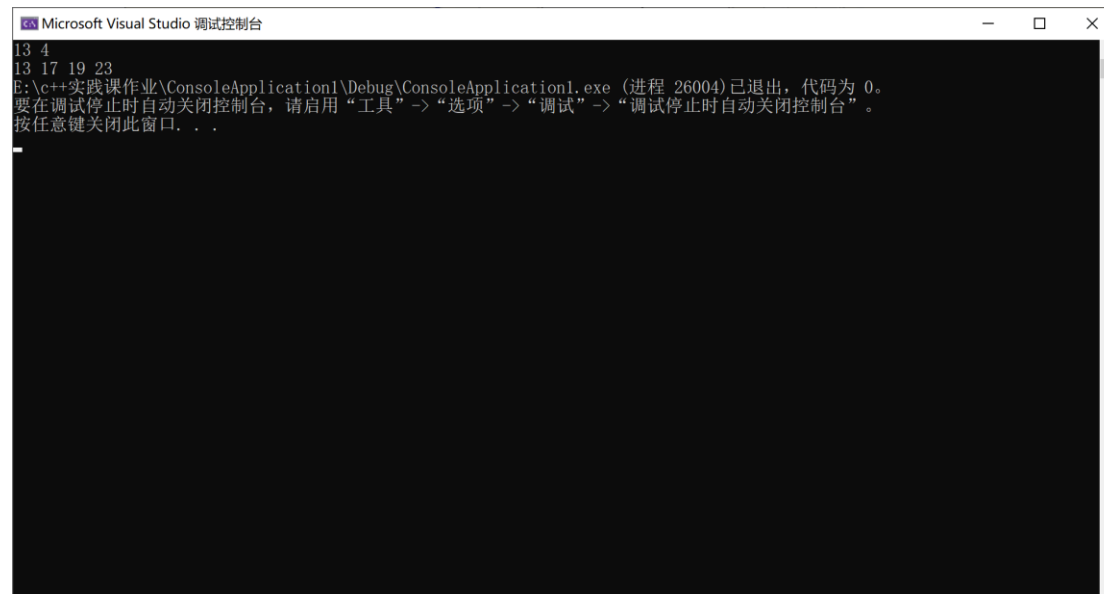
```

{
    int j, k, t;
    t = 0;
    for (j = i;; j++)
    {
        for (k = 2; k < j; k++)
        {
            if (j % k == 0)
            {
                break;
            }
            if (k == j - 1)
            {
                cout << j << " ";
                t++;
            }
        }
        if (t == n)
            break;
    }
}

int main()
{
    int i, n;
    cin >> i >> n;
    find(i, n);
    return 0;
}

```

2.3 运行截图



2.4 调试情况

3、第三题

3.1 题目描述

A. 问题描述

输入正整数 `start` 和 `end`，找出从 `start` 到 `end` 这一区间段内哪一个数 `n` 的因子和最大，并将该 `n` 及其因子和 `maxSum` 作为结果输出。例如，当 `start=10`，`end=15` 时，所求的 `n` 应该为 12，而 `maxSum` 应该为 16。

要求：设计函数计算 `n` 的因子和，其中因子和为包括 1 但不包括 `n` 本身的所有因子之和。

B. 输入

两个整数，用来指定查找的起始数字和结束数字。

C. 输出

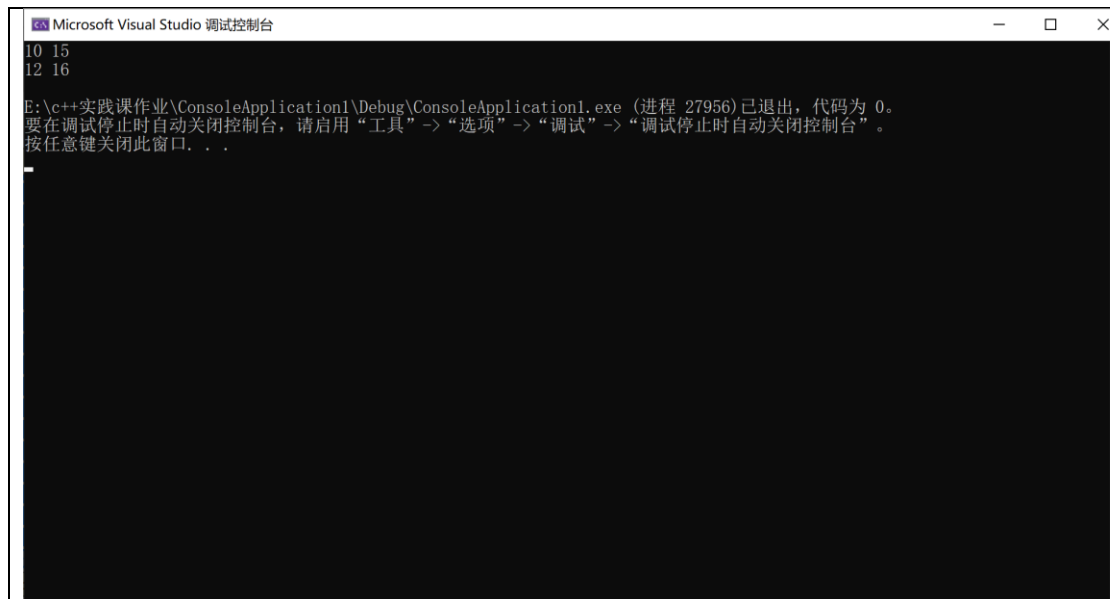
两个整数，中间用一个空格隔开，前一个是因子和最大的数字，后一个是因子和，最后一个数字后面不要有空格和换行。

3.2 源代码

```
#include <iostream>
using namespace std;

int main()
{
    int start, end, i, j, sum, maxSum, n;
    maxSum = 0;
    n = 0;
    cin >> start >> end;
    for (j = start; j < end + 1; j++)
    {
        for (i = 1, sum = 0; i < j; i++)
        {
            if (j % i == 0)
            {
                sum += i;
            }
        }
        if (maxSum < sum)
        {
            maxSum = sum;
            n = j;
        }
    }
    cout << n << " " << maxSum << endl;
    return 0;
}
```

3.3 运行截图



3.4 调试情况

4、第四题

4.1 题目描述

A. 问题描述

编写具有如下原型的函数 `symm: bool symm(long n)`; 用来判断正整数 `n` 是否为“回文数”（正读与反读为大小相同的数），若是返回 `true`，否则返 `false`。并编写主函数，通过调用 `symm`，求出 `n` 以内所有满足下述特征的 `m`: `m` 和 `7*m` 以及 `3*m*m` 都是“回文数”，如 1, 11, 88, ...（因为 `m=1`, `7*m=7`, `3*m*m=3`; `m=11`, `7*m=77`, `3*m*m=363`; `m=88`, `7*m=616`, `3*m*m=23232`; ...）。

B. 输入

搜索范围。

C. 输出

输出整数，每个整数后面有一个空格用于分隔，最后一个整数后可以有一个空格，但不要换行。

4.2 源代码

```
#include <iostream>
#include <cmath>

using namespace std;

bool symm(long n)
{
    int m[100] = {};
    int i = 0;
    int j = 0;
    if (n < 0)
    {
        return false;
    }
}
```

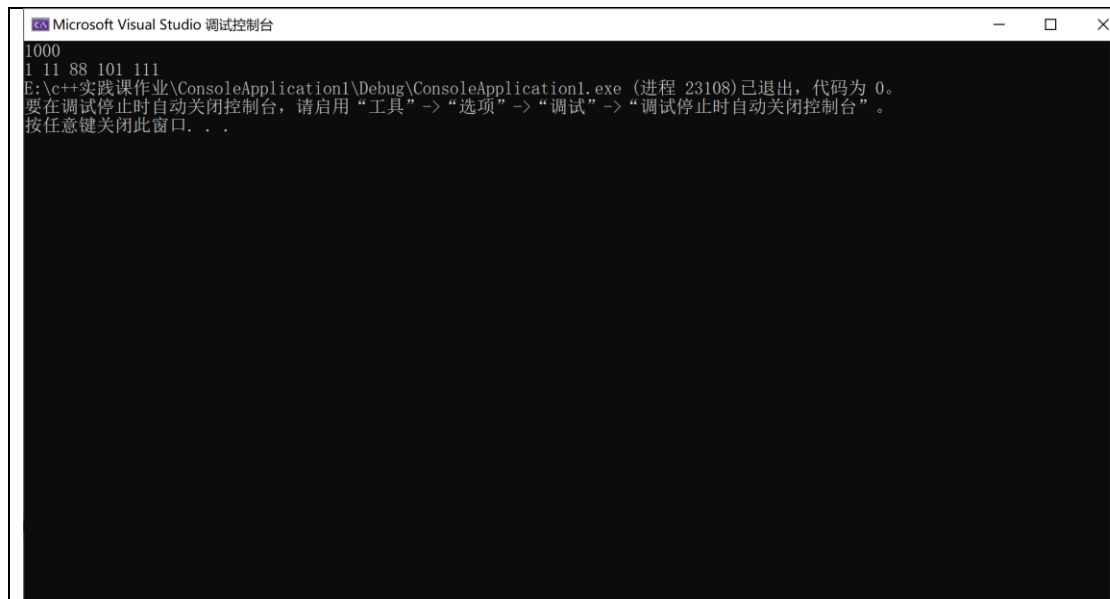
```

while (n)
{
    m[i] = n % 10;
    n = n / 10;
    i += 1;
}
while (j != i / 2)
{
    if (m[j] != m[i - 1 - j])
    {
        return false;
    }
    j += 1;
}
return true;
}

int main()
{
    long n, m;
    cin >> n;
    for (m = 1; m < n; m++)
    {
        if (symm(m) && symm(7 * m) && symm(3 * m * m))
        {
            cout << m << " " << " ";
        }
    }
    return 0;
}

```

4.3 运行截图



4.4 调试情况

5、第五题

5.1 题目描述

A. 问题描述

键盘输入正整数 n ，求出 n 与其反序数 x 之和并输出。例如，输入 2038， $n+x = 2038 + 8302 = 10340$ ，输出应为 10340。要求：编写函数实现数据转换成反序数值。

B. 输入

输入一个整数。

C. 输出

输出一个整数。

5.2 源代码

```
#include <iostream>
#include <cmath>

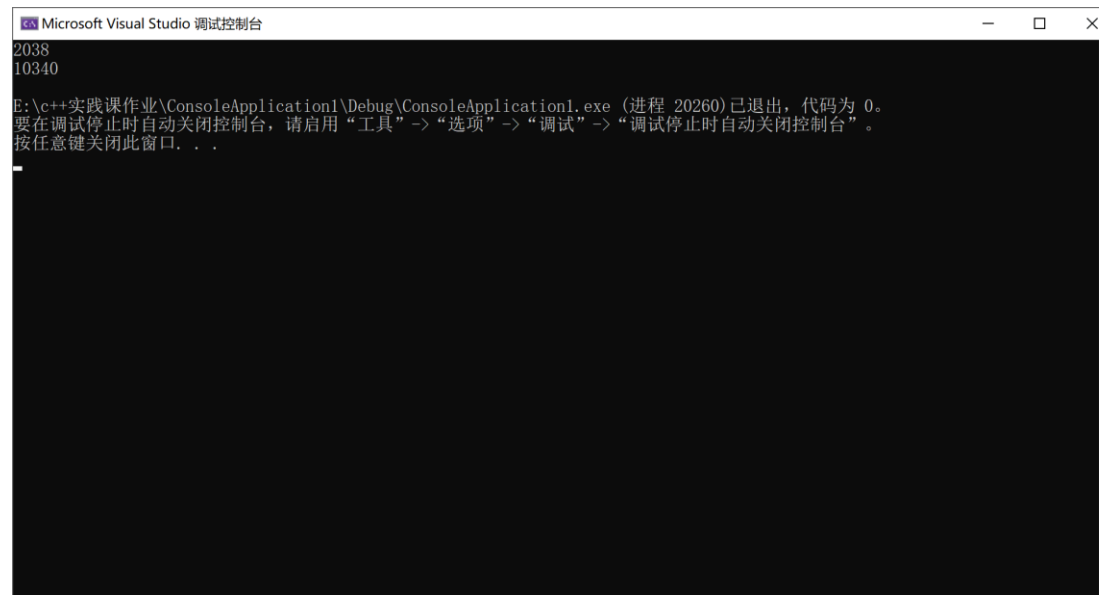
using namespace std;

int fun(int n)
{
    int i, j, sum = 0, t = 0;
    for (i = 0;; i++)
    {
        if (n / (int)pow(10, i) == 0)
        {
            break;
        }
    }
    for (j = i - 1; j >= 0; j--)
    {
```

```
        t = n % 10;
        n /= 10;
        sum += t * pow(10, j);
    }
    return sum;
}

int main()
{
    int n;
    cin >> n;
    cout << n + fun(n) << endl;
    return 0;
}
```

5.3 运行截图



5.4 调试情况

三、实验体会

通过这次实验,我更加明白了 c++理论中的一些编程规范和 c++语言特性,掌握了基本编程知识,以后会更加认真的学习 c++理论知识,并不断实践和练习,在 debug 中不断学习。