|  |  |
| --- | --- |
| **编号:** | **第1章第1次** |

****

信息科学与工程学院实验报告

《面向对象程序设计》

**Object-Oriented Programming**

|  |  |
| --- | --- |
| 姓名： | 颜丙超 |
| 学号： | 202311000415 |
| 班级： | 计工本2301 |
| 教师: | 张庆科 |
| 时间： | 2024年10月13日 |

**《面向对象程序设计》实验报告**

**报告要求：**实验报告包含实验目的、实验内容、实验过程（详细操作流程）、实验结果（程序运行结果高清截图）、实验分析总结五个部分。报告中若涉及代码程序，请在附录部分提供完整程序源码及源码托管地址(基于Highlight软件导入源码)。报告撰写完毕后请将PDF格式版本上传到坚果云作业提交系统。

1. **实验目的**

\* 掌握使用C++面向过程程序设计方法

\* 掌握C++对C语言的典型改进和扩展

\* 掌握Visual Studio建立项目步骤

\* 掌握Visual Studio程序调试方法

\* 掌握Git的基本操作原理和常见命令

1. **实验内容**

****（1）任务一：论述题****  
阐述git的基本工作过程和工作原理，基于代码托管网站gitee或GitHub给出git仓库项目的创建过程。然后结合课程PPT有关git的内容，给出git常见的操作命令的使用方法。

****（2）任务二:  程序设计题- C++素数判断****

  建立VS项目, 在源文件```main.cpp```中定义素数判别函数```isPrimeNumber()```, 在主函数中输入一个整数m, 然后输出该整数是否为素数的信息.

测试案例:

--------------------------------  
 请输入任意一个整数: 17

 整数17是素数: True  
--------------------------------

****（3）任务三: 程序设计题- C++验证哥德巴赫猜想****

****哥德巴赫猜想****: 对于任意一个不小于6的偶数,均可以将其表示为两个素数之和,例如: 6=3+3, 8=3+5, 12=5+7,...等等. 请在主函数中输入一个不小于6的整数N, 然后调用偶数解码函数DecodeEvenToPrime(int num),输出小于整数N的所有偶数的素数分解形式.

程序运行测试案例:

---------------------------------  
请输入一个整数N (N>6): 20  
偶数20的素数分解结果: 20 = 3 + 17  
偶数20的素数分解结果: 20 = 7 + 13  
偶数18的素数分解结果: 18 = 5 + 13  
偶数18的素数分解结果: 18 = 7 + 11  
偶数16的素数分解结果: 16 = 3 + 13  
偶数16的素数分解结果: 16 = 5 + 11  
偶数14的素数分解结果: 14 = 3 + 11  
偶数14的素数分解结果: 14 = 7 + 7  
偶数12的素数分解结果: 12 = 5 + 7  
偶数10的素数分解结果: 10 = 3 + 7  
偶数10的素数分解结果: 10 = 5 + 5  
偶数8的素数分解结果: 8 = 3 + 5  
偶数6的素数分解结果: 6 = 3 + 3  
---------------------------------  
输出完毕!

\* 如果未按照要求输入,例如输入2,4,或者输入字符a, '\*' 等,程序会出现什么情况? 请继续修改完善解码函数,确保程序输入输出的健壮性.

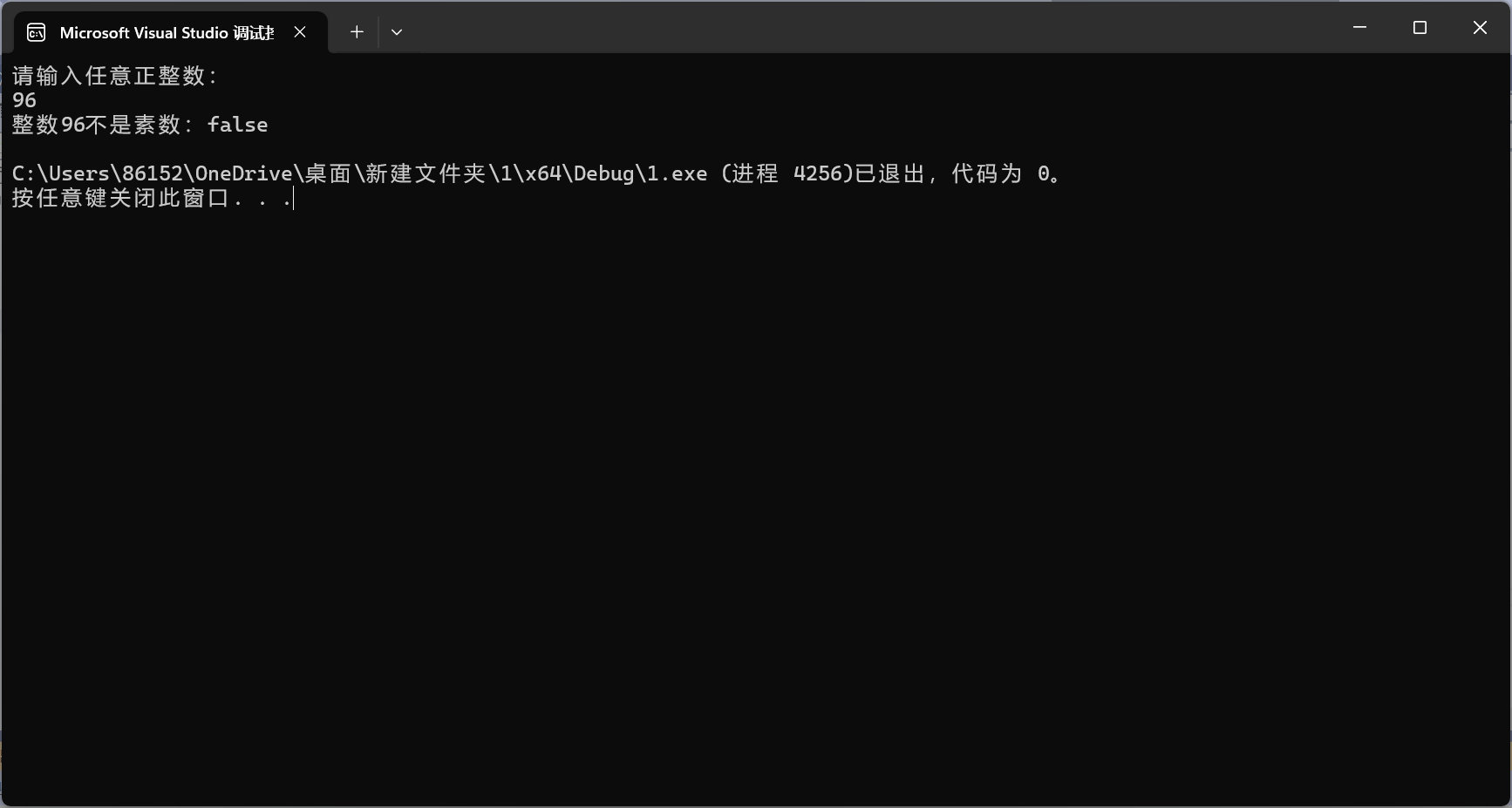
1. **实验过程**

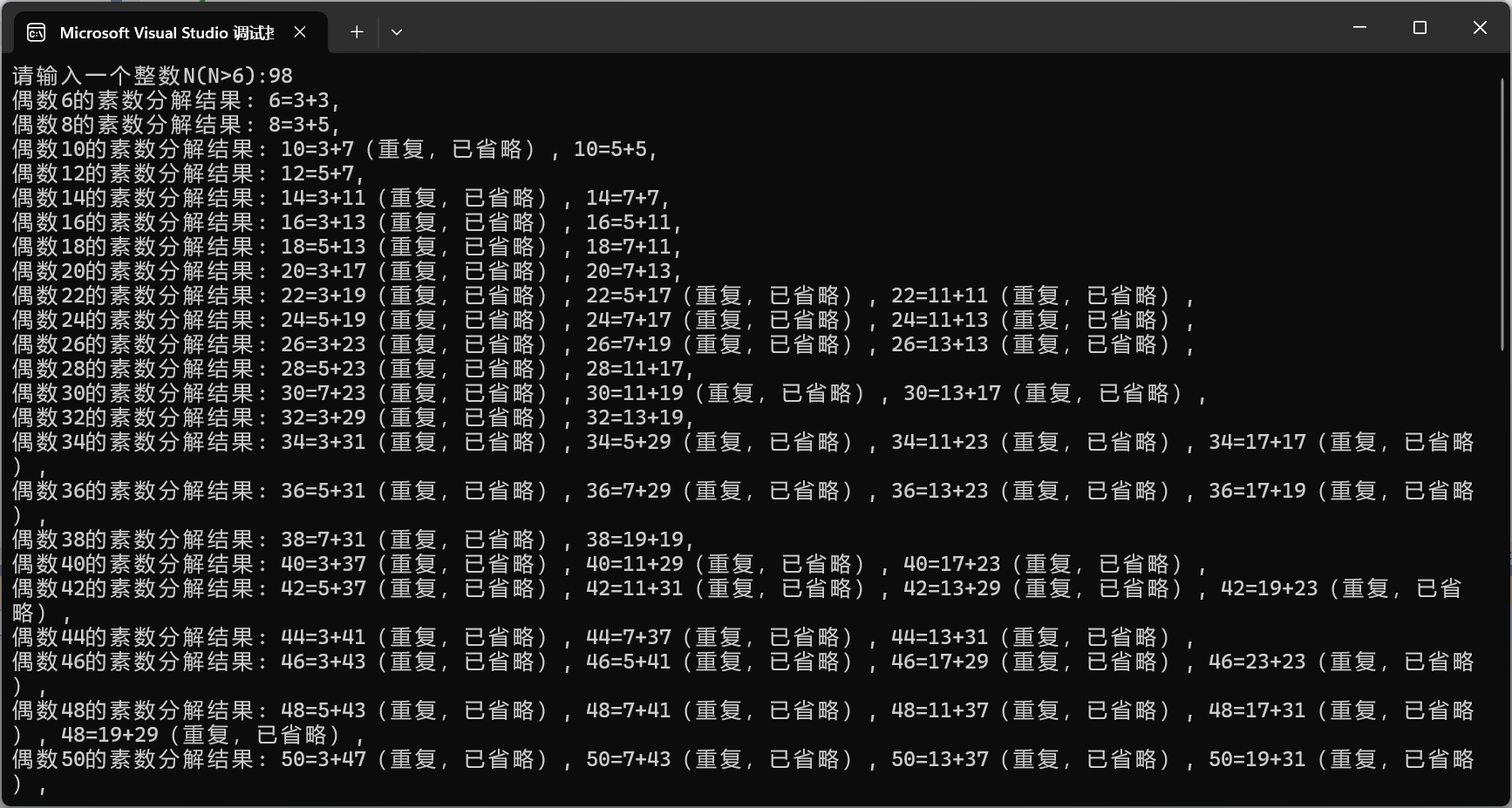
实现二：首先定义一个n,利用质数的性质，分成三种情况，首先定义小于等于一的情况，再在大于等于一的情况下，看是否有属于不是本身和一的因数，在进行定义，再根据是否是质数来输入结果。

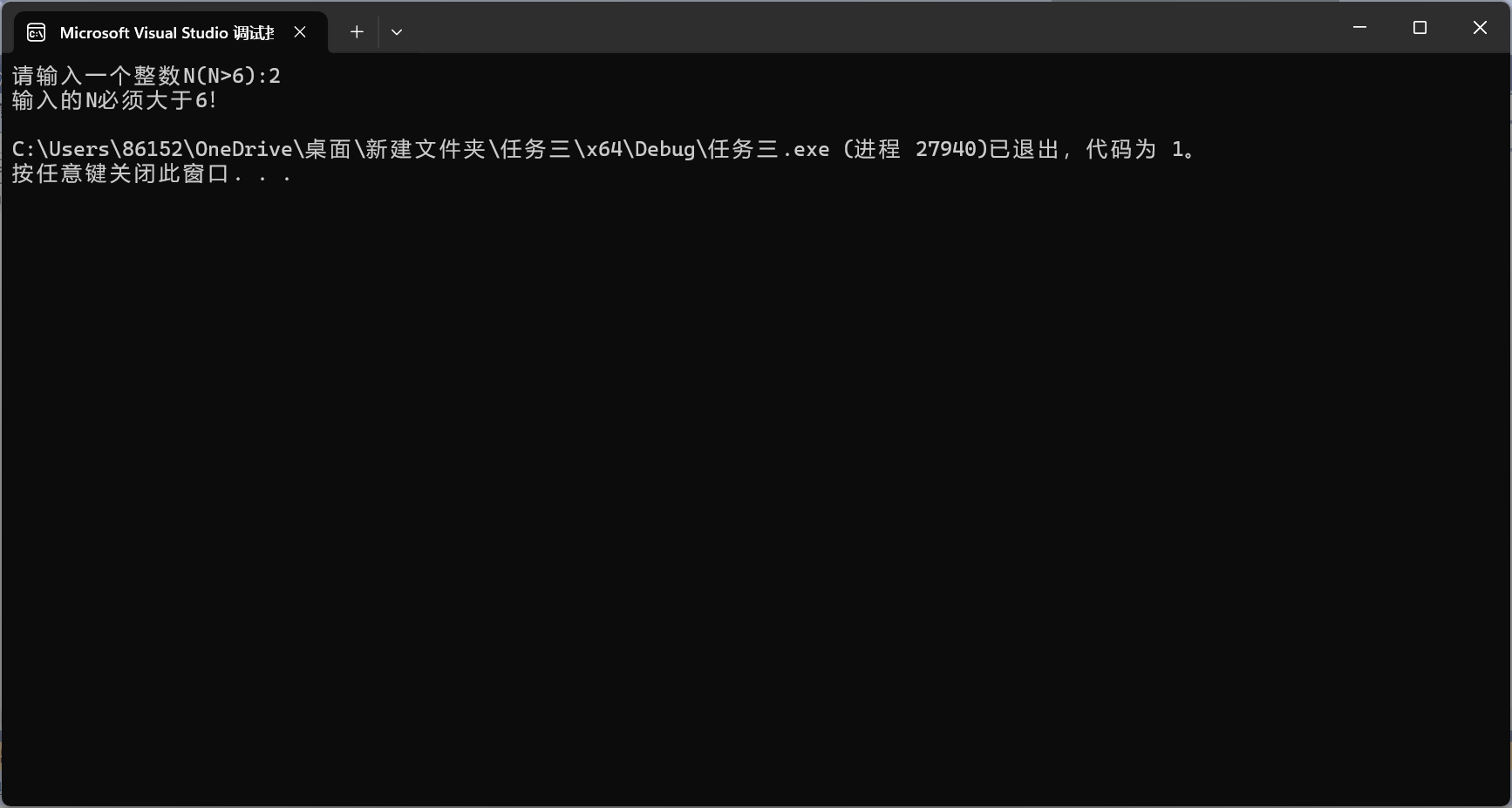
实验三：首先第一部分有实验二的代码组成，第二部分要根据每个偶数，遍历2到j/2所有的整数i,检查i和i-1是否都是素数。最后进行打印，结果分为大于6 和 小于6的情况，如果输入2或者4和a的情况，该程序会让你输入大于6的数。

1. **实验结果**

实验一：基本工作过程分为工作区，暂存区和本地仓库以及远程仓库，首先在Gitee仓库克隆代码到本地仓库，在工作区修改代码，在暂存区合并修改，提交版本到本地仓库，最后传输到远端仓库。Git常见的操作命令:git config:配置git的用户信息与全局设置 git init：初始化一个新git库 git add：将文件添加到暂存区 git commit：提交暂存区的更改到本地仓库 git push：将本地更改推送到远程仓库 git pull：从远程仓库获取更改并合并到本地分支 git status:显示当前工作目录和暂存区的状态 git log：显示提交历史记录。

实验二：

实验三：



1. **实验总结**

实验一中我学会了Gitee中的一些基本操作，让学习以及查阅代码有了一个更方便快捷的方法。实验二学会了使用c++语言来编程素数的方法，基于c语言的学习，c++的语言更加方便快捷，然后还学会了c++语言中的输入与输出功能，进一步巩固了自己的知识，实验三中进一步扩展了素数的知识，对与更深一步的C++编程有了更好的了解。

* **附录：实验源代码（基于Highlight软件粘贴带有行号的源码）**

#include<iostream>

#include<cmath>

bool isPrimeNumber(int n) {

if (n <= 1)

{

return false;

}

for (int i = 2;i <= *std*::*sqrt*(n);++i)

{

if (n % i == 0)

{

return false;

}

}

return true;

}

int main()

{

int n;

*std*::*cout* << "请输入任意正整数：" << *std*::*endl*;

*std*:: *cin* >> n;

isPrimeNumber(n);

if (isPrimeNumber(true)) {

*std*:: *cout* <<"整数"<<n<< "是素数：true" << *std*::*endl*;

}

else *std*:: *cout* << "整数"<<n<<"不是素数：false" << *std*::*endl*;

return 0;

}

实验三

#include <iostream>

#include <vector>

#include <cmath>

bool isPrime(int n) {

if (n <= 1) return false;

for (int i = 2; i <= *std*::*sqrt*(n); ++i) {

if (n % i == 0) return false;

}

return true;

}

void decodeEvenToPrime(int N) {

for (int j = 6; j < N; j += 2) {

*std*::*vector*<*std*::*pair*<int, int>> primePairs;

for (int i = 2; i <= j / 2; ++i) {

if (isPrime(i) && isPrime(j - i)) {

primePairs.*push\_back*(*std*::*make\_pair*(i, j - i));

}

}

*std*::*cout* << "偶数" << j << "的素数分解结果:";

for (const auto& pair : primePairs) {

*std*::*cout* << " " << j << "=" << pair.*first* << "+" << pair.*second*;

if (primePairs.*end*() != *std*::*find\_if*(primePairs.*begin*() + 1, primePairs.*end*(), [&](const *std*::*pair*<int, int>& p) {

return p.*first* != pair.*first* && p.*second* != pair.*second*;

})) {

*std*::*cout* << "（重复，已省略）";

}

*std*::*cout* << ",";

}

*std*::*cout* << *std*::*endl*;

}

}

int main() {

int N;

*std*::*cout* << "请输入一个整数N(N>6):";

*std*::*cin* >> N;

if (N <= 6) {

*std*::*cout* << "输入的N必须大于6！" << *std*::*endl*;

return 1;

}

decodeEvenToPrime(N);

return 0;

}