**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**

**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

**Кафедра «АПУ»**

**ОТЧЁТ**

по лабораторной работе № 2

**по дисциплине «Программирование»**

**«АНАЛИЗ ЧИСЛОВЫХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ»**

|  |  |  |
| --- | --- | --- |
| Студент гр. 4391 | **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | Зайцева О. Ю. |
| Преподаватель | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Писарев А.С. |

Санкт-Петербург

2024 г.

**Цель работы:**

Научиться анализу числовых последовательностей.

Написать программу, используя язык программирования C++.

**Ход работы:**

Была изучена задача на анализ числовых последовательностей.

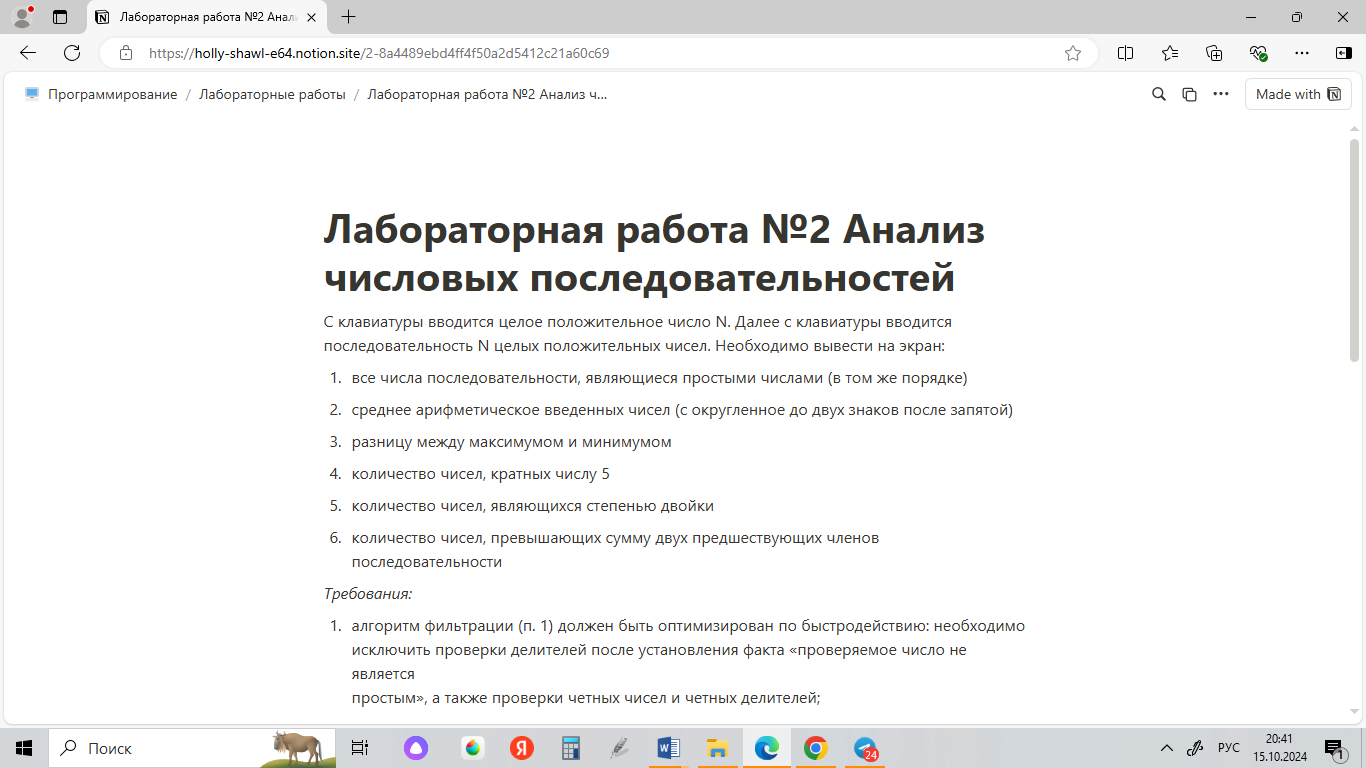


Рисунок 1 – Задача

В самом начале было выполнено подключение к библиотекам.

Библиотека <iostream> предназачена для организации ввода и вывода.

Библиотека <cmath> предназначена для выполнения простых математических операций.

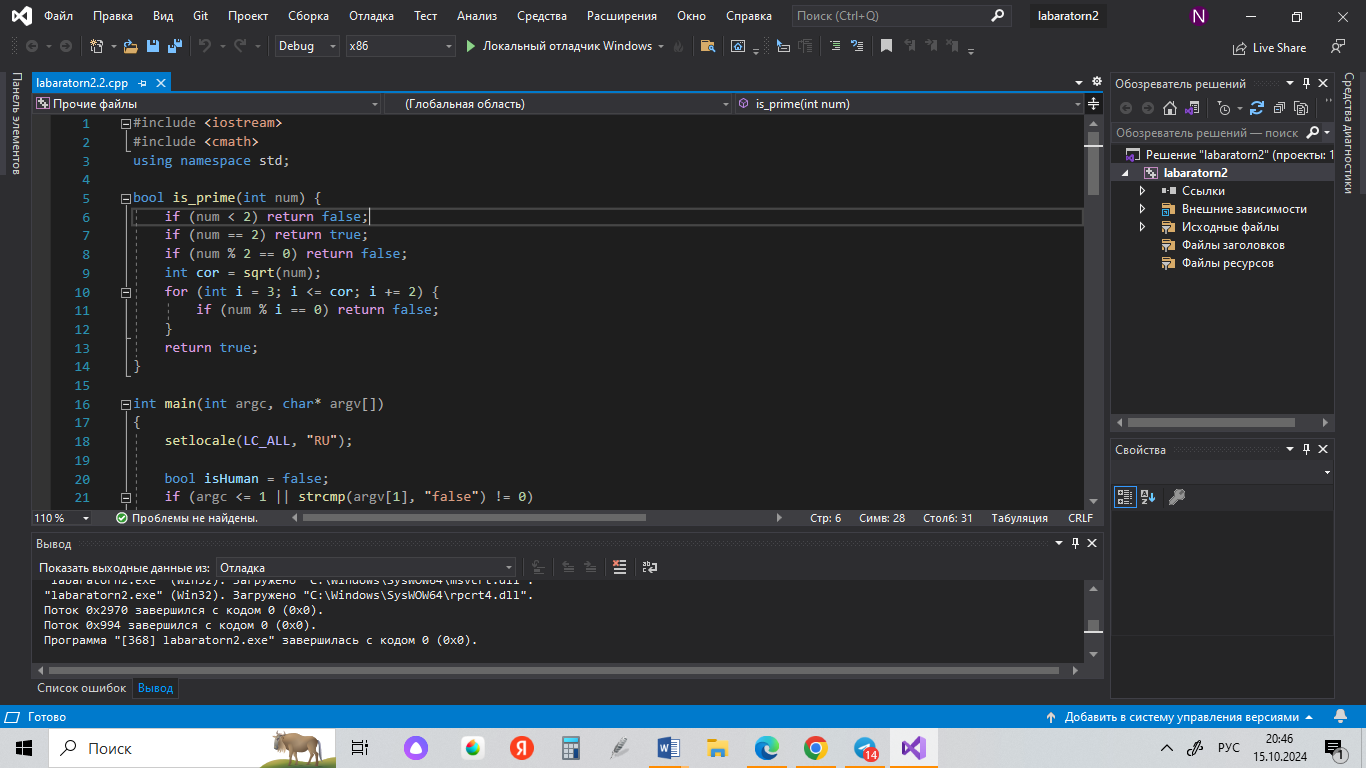


Рисунок 2 – Подключенные библиотеки

Инструкция using namespace std представляет собой команду для использования стандартного пространства имён std.

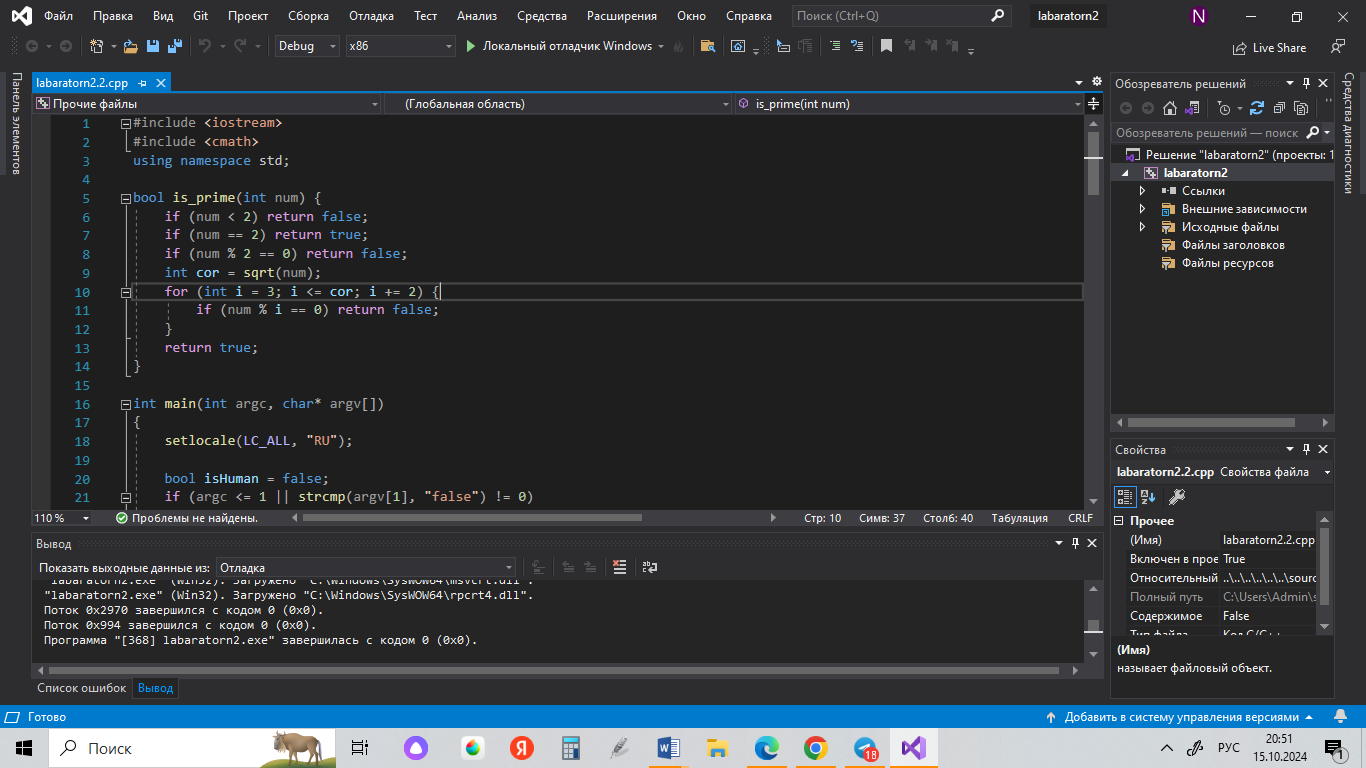


Рисунок 3 – Пространство имён

Добавлена функция bool, которая возвращает False, если объект пуст или равен нулю True в противном случае. В своей программе я использую данную функцию для проверки того, чтобы узнать является число простым или нет. Для того, чтобы число было простым оно должно быть больше или равно 2, не должно делиться на 2 без остатка, а также простое число не может иметь корней.

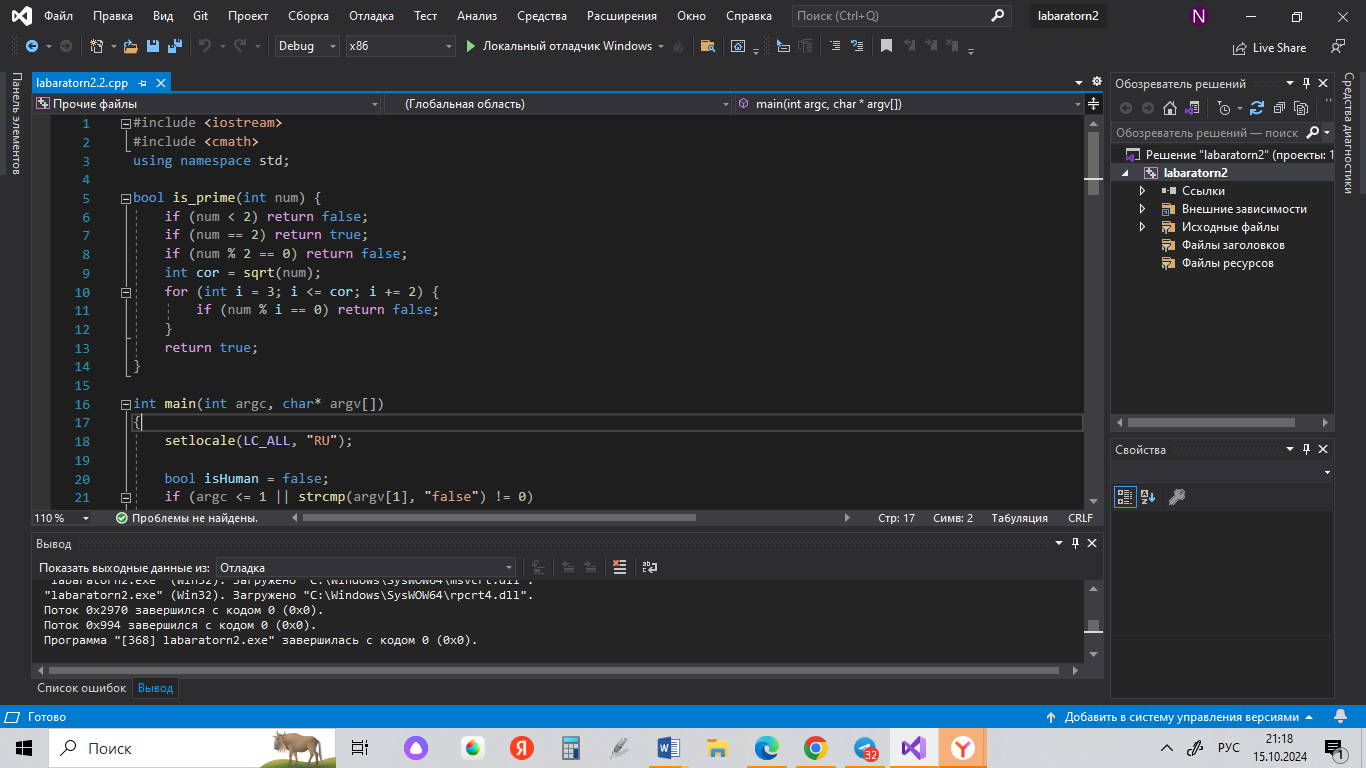


Рисунок 4 – Создание тела функции

Создано тело функции. Добавлена функция setlocale. Она позволяет устанавливать различные параметры в зависимости от географического положения пользователя, например, позволяет пользоваться тем алфавитом и временем, которые есть в этой стране. Для моей программы я указала “RU”.

Также я ещё раз использую функцию bool для проверки того, чтобы узнать человек пользуется данной программой или робот. Если программу использует человек, для него выводится полный интерфейс с подробным описанием действий. Упрощённый интерфейс краткий и ориентирован на автоматизированную проверку.

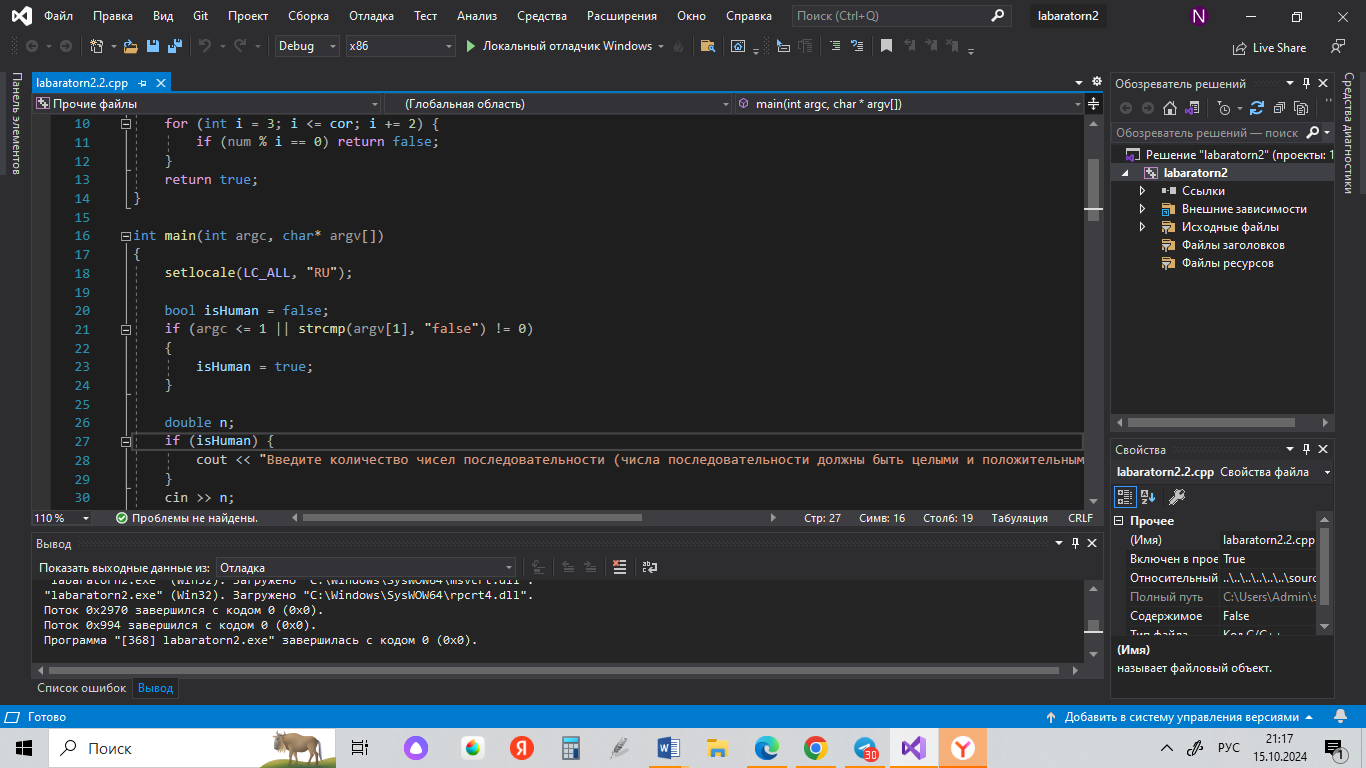


Рисунок 5 – Создание тела функции

Ввожу переменную n типа double. Для отображения данных в консоли использован оператор cout, который служит для вывода простого текста, а также переменных в консоль. Для получения данных от пользователя использую оператор cin.

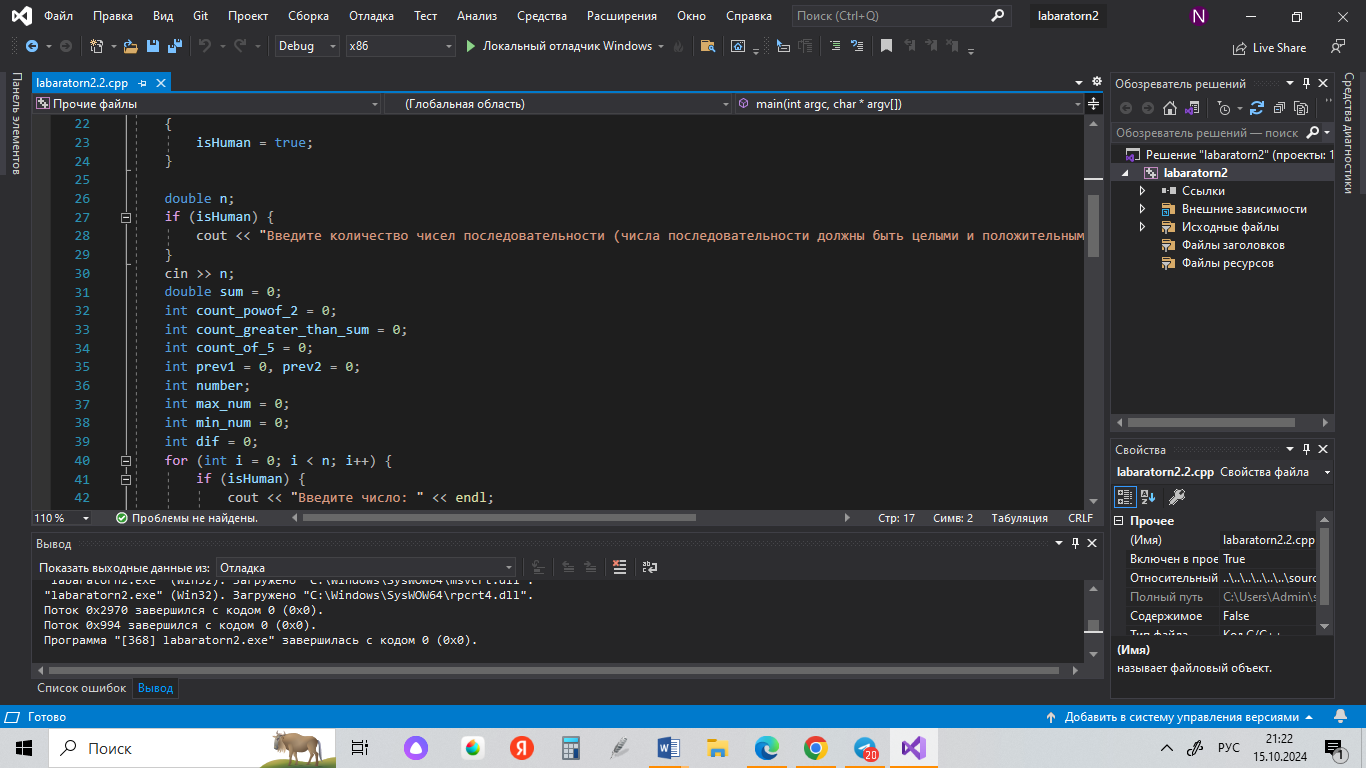


Рисунок 6 – Операторы вывода и ввода

Ввожу переменные для разных действий. Переменная sum типа double будет отвечать за суммирование всех чисел последовательности. Переменная count\_powof\_2 типа int будет отвечать за количество чисел, являющихся степенью двойки. Переменная count\_greater\_than\_sum типа int будет отвечать за количество чисел, превышающих сумму двух предшествующих членов последовательности. Переменная count\_of\_5 типа int будет отвечать за количество чисел, кратных числу 5. Переменные prev1 и prev2 типа int будут нужны при нахождении max\_num и min\_num, в них мы будем перезаписывать числа, также эти переменные нужны при проверке количества чисел, превышающих сумму двух предшествующих членов последовательности. Переменная number типа int нужна при вводе чисел человеком. Переменная max\_num и min\_num типа int нужны для записи максимума и минимума соответственно. Переменным присваиваю значение INT\_MIN и INT\_MAX соответственно. INT\_MIN минимальное значение для переменной типа int (равно -2147483647 - 1). INT\_MAX максимальное значение для переменной типа int (равно 2147483647). Переменная dif типа int будет нужна при нахождении разности между максимумом и минимумом.

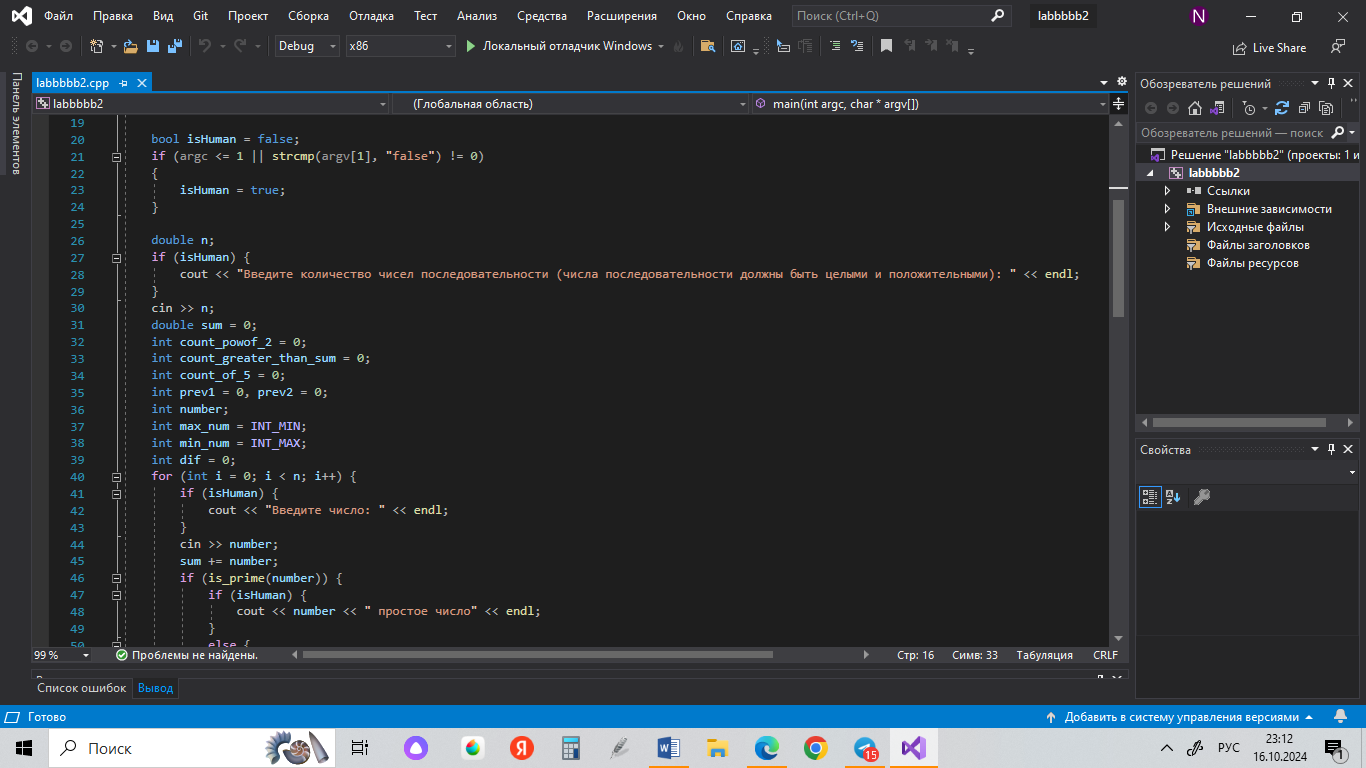


Рисунок 7 – Ввожу переменные

Прописала цикл для введения последовательности чисел.

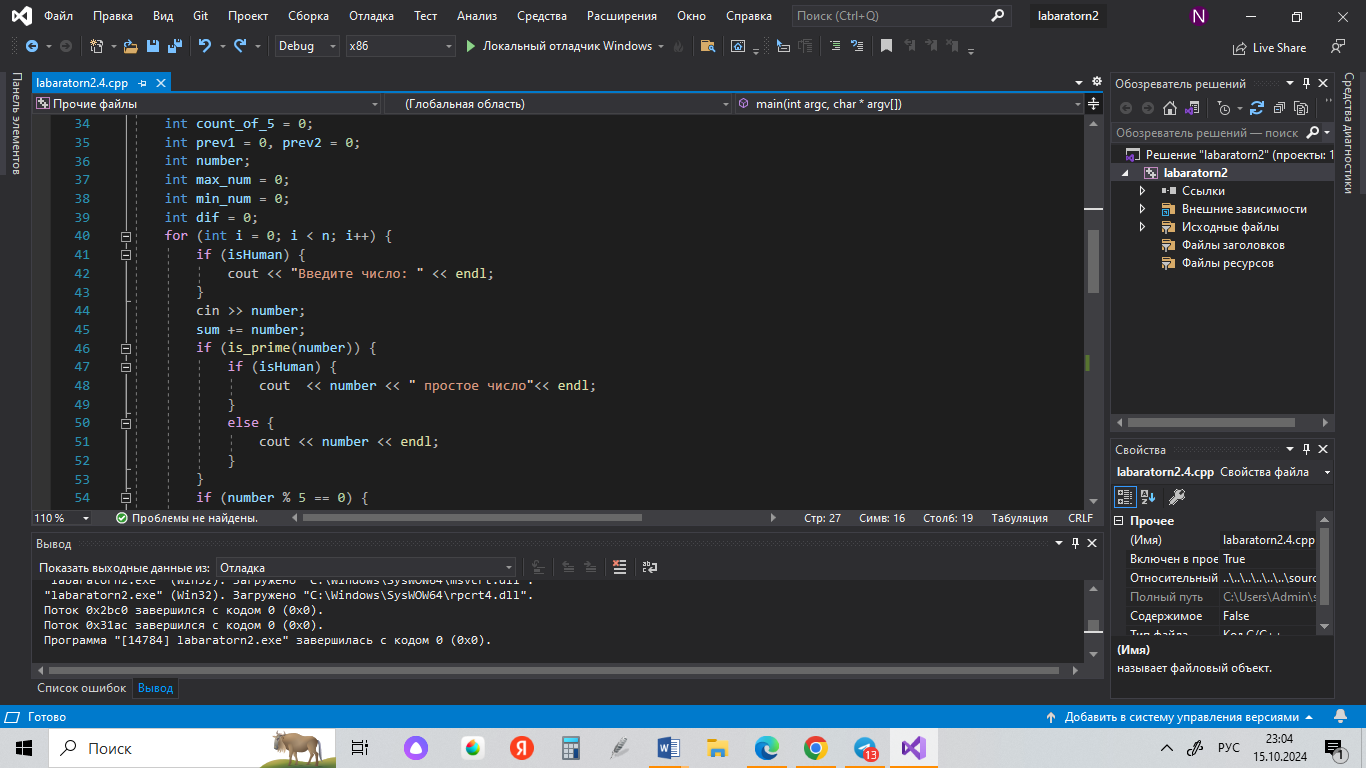


Рисунок 8 – Ввод последовательности чисел

Написала код для суммирования всех чисел последовательности.

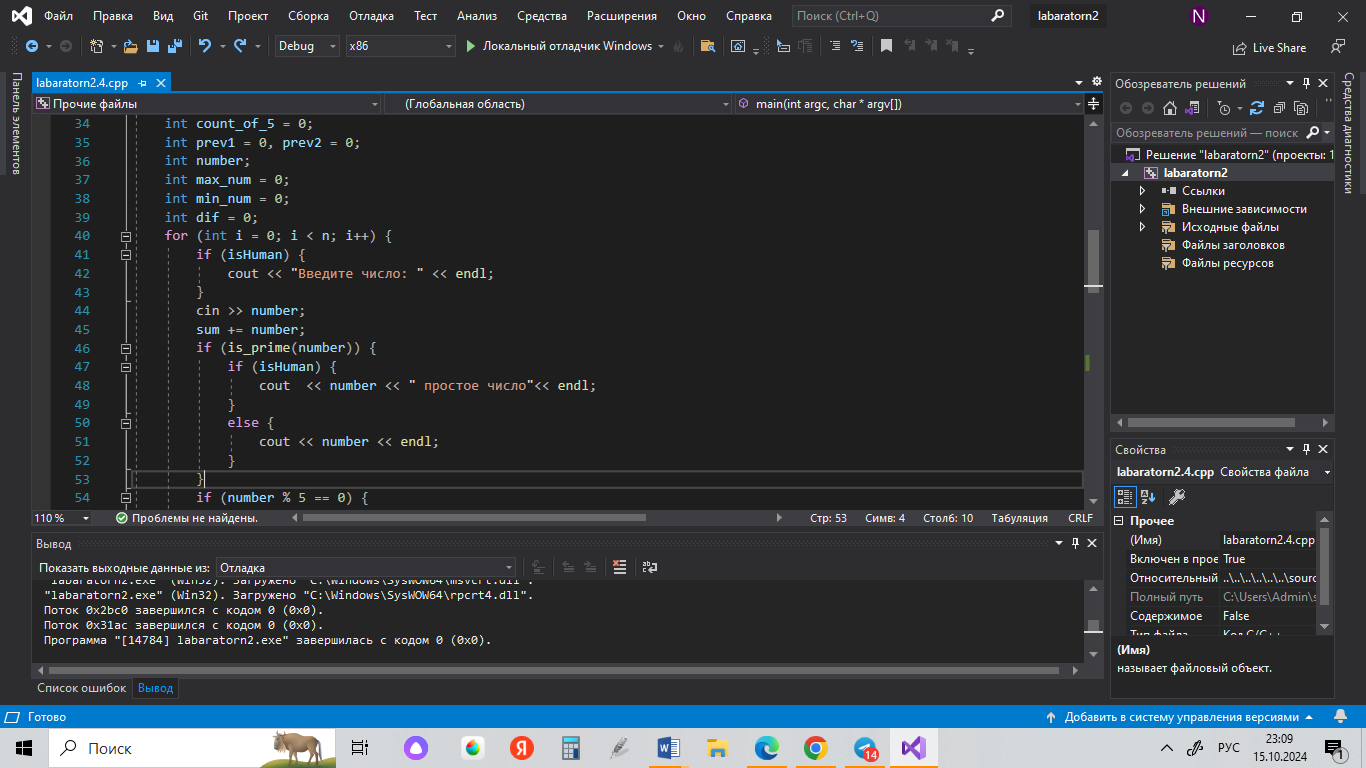


Рисунок 9 – Суммирование всех чисел последовательности

С помощью конструкции if, else вывожу на экран является ли число простым или нет. Если введённое число простое и программу запускает человек, то выводится это же число с надписью «простое число».

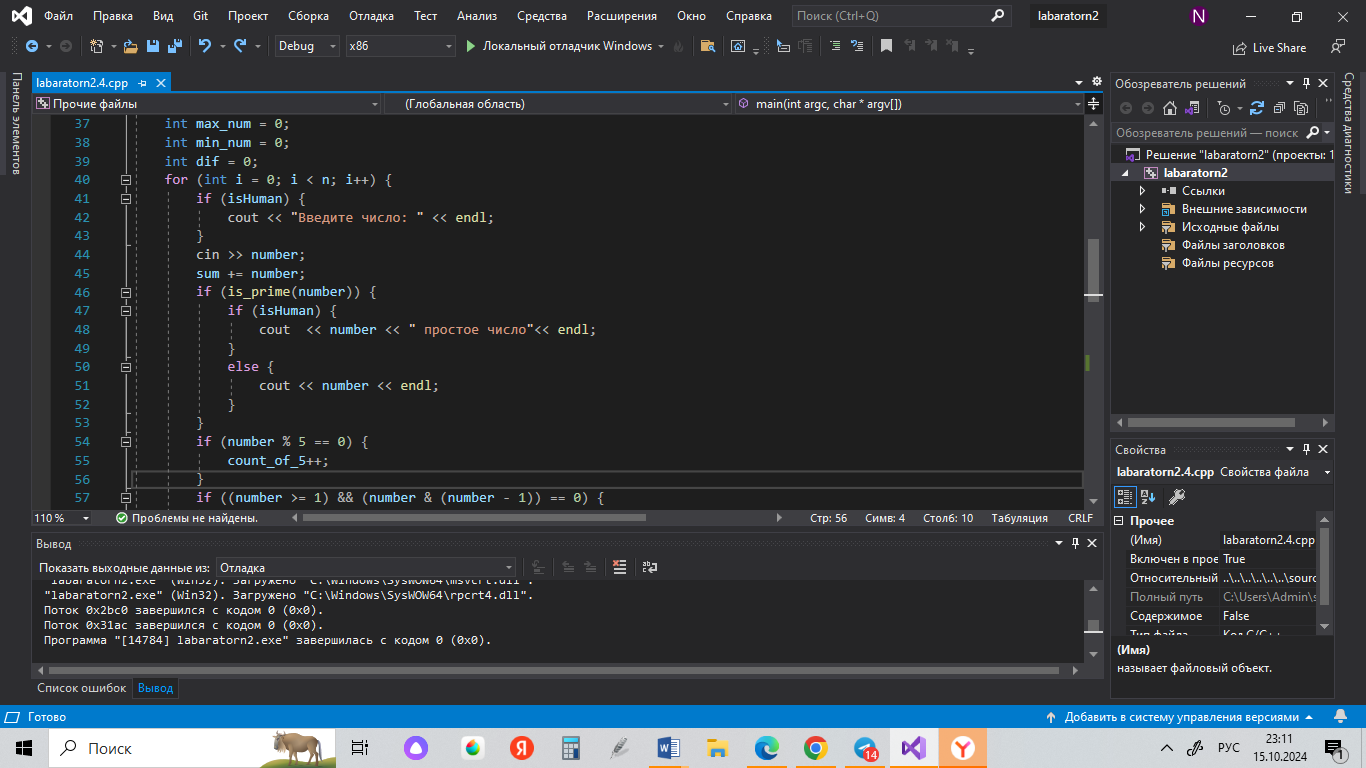


Рисунок 10 – Вывод простого числа

С помощью оператора if проверяю число на кратность числу 5.

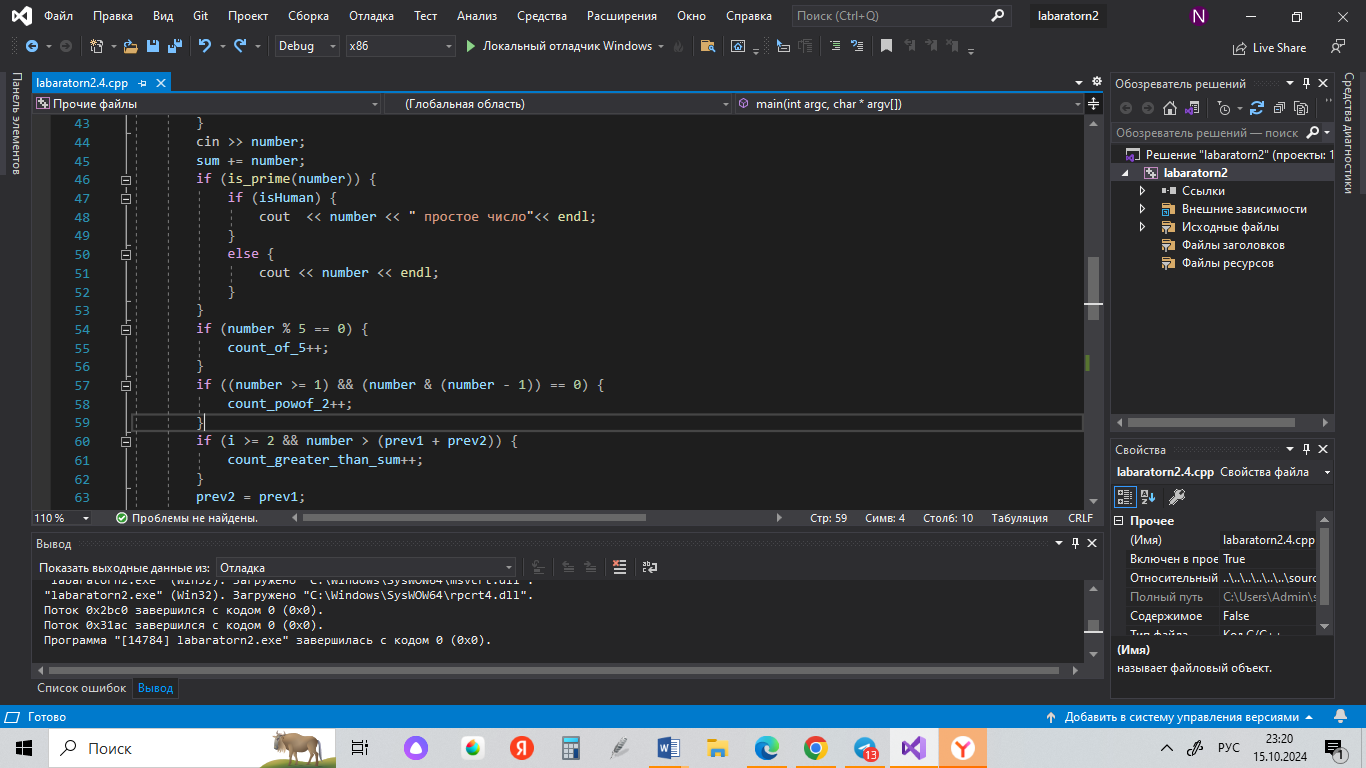


Рисунок 11 – Проверка числа на кратность числу 5

С помощью оператора if проверяю число, является ли оно степенью двойки или нет. Является число степенью двойки или нет можно проверить с помощью двоичной системы. В двоичном представлении степени 2 имеют один бит, установленный на 1, а все остальные биты равны 0 (например, 1 (0001), 2 (0010), 4 (0100), 8 (1000) и т. д.). Для числа, которое является степенью 2, (n) и (n-1) не имеют общих установленных битов. Для степени числа 2 двоичное представление равно 1000...0. Вычитание из него 1 дает 0111...1. Выполнение побитовой операции И между (n) и (n-1) дает результат 0, если (n) является степенью числа 2.

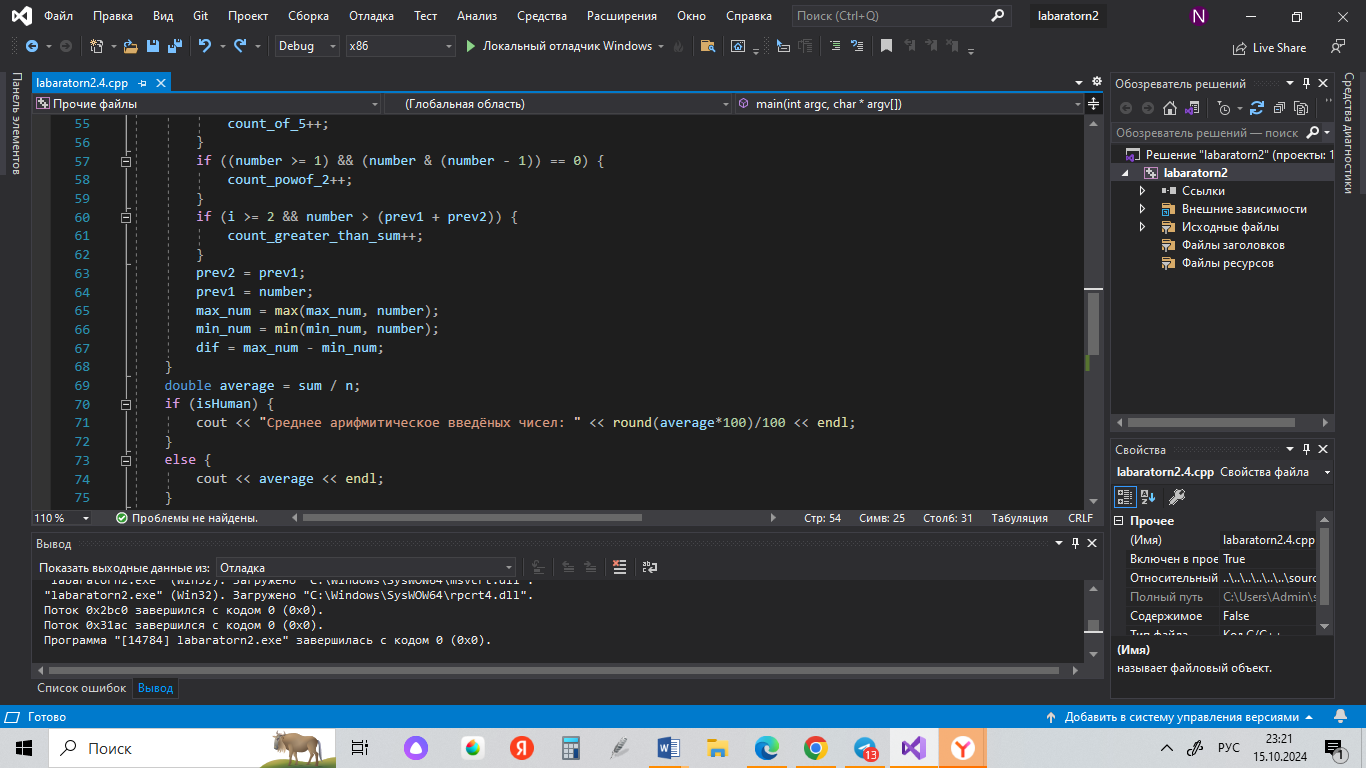


Рисунок 12 – Проверка числа, является число степенью двойки или не является

С помощью оператора if нахожу количество чисел, превышающих сумму двух предшествующих членов последовательности.

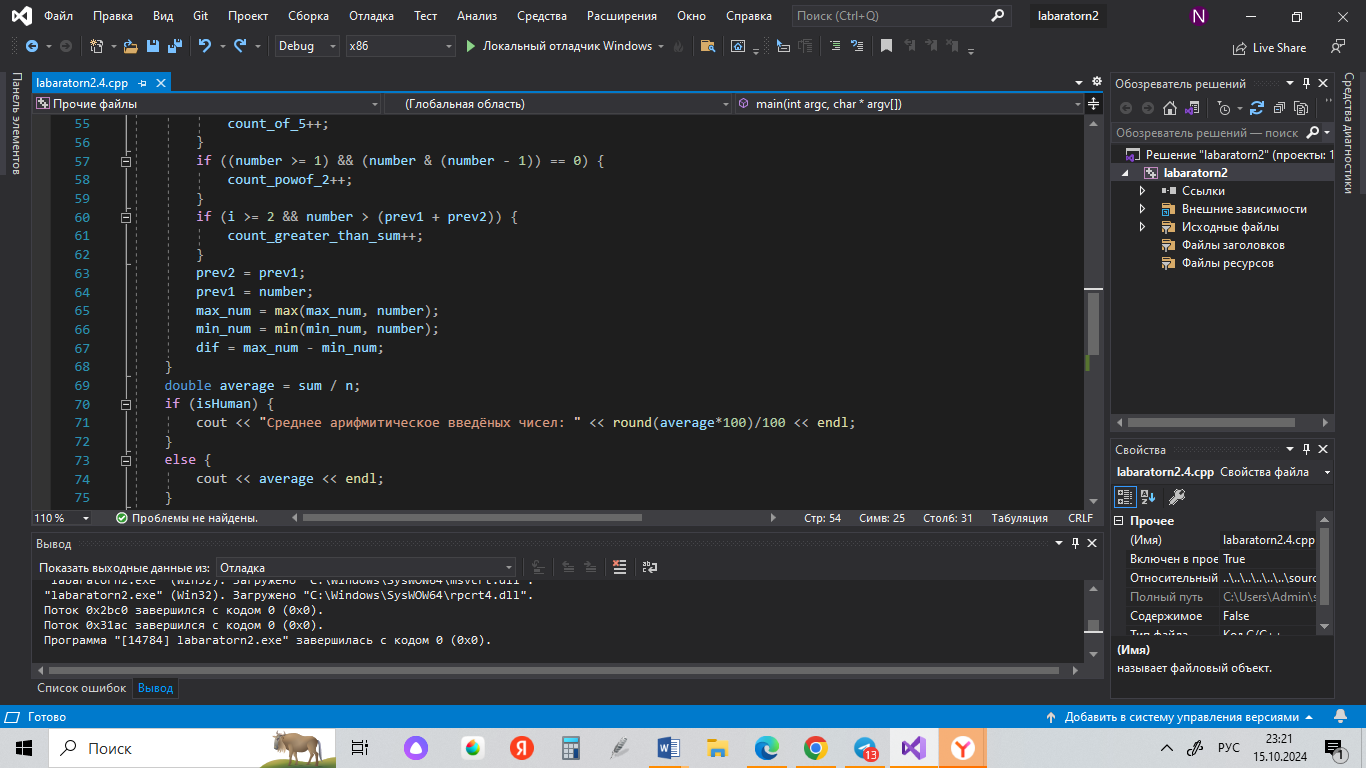


Рисунок 13 – Нахождение количества чисел, превышающих сумму двух предшествующих членов последовательности

Нахожу среди введенных чисел максимум и минимум. Далее нахожу разность между максимумом и минимумом.

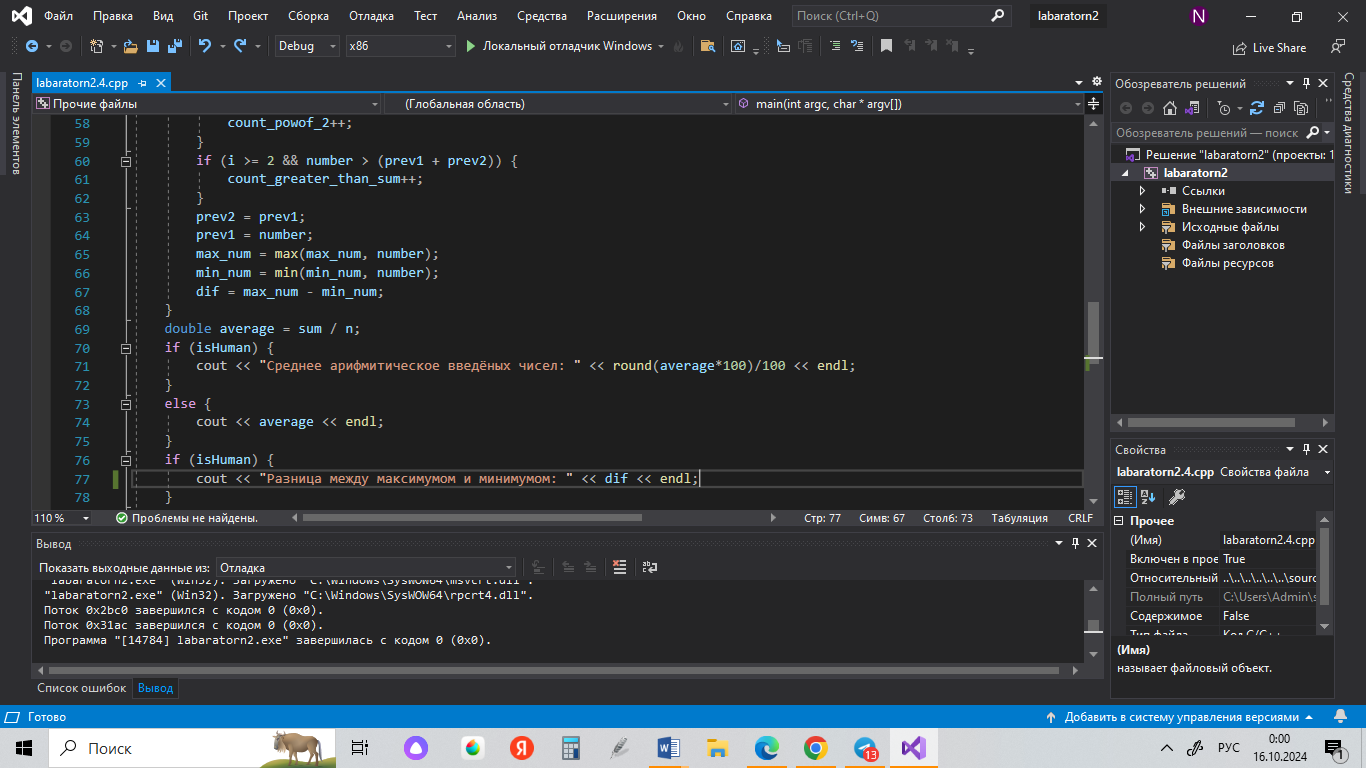


Рисунок 14 – Нахождение максимума, минимума и разности между ними

Ввожу переменную average типа double, данная переменная вычисляет среднее арифметическое введённых чисел.

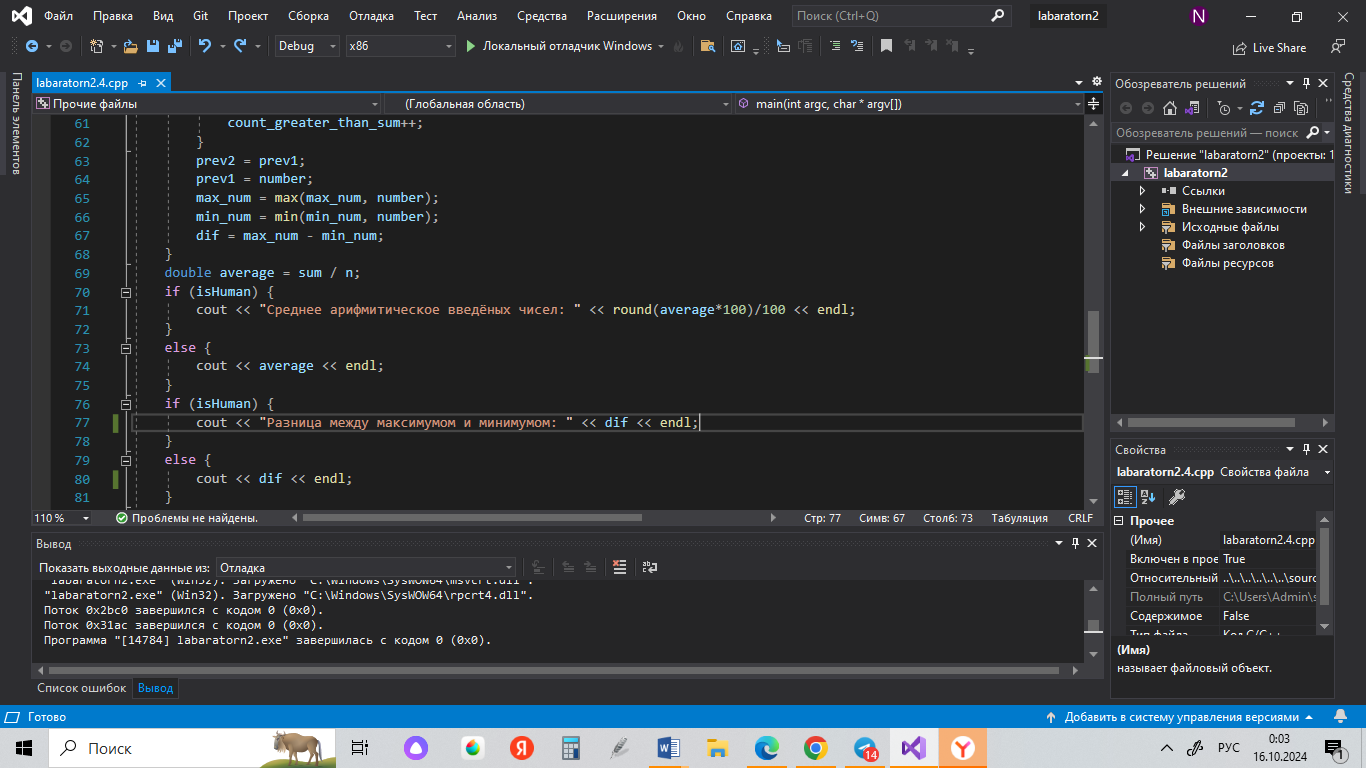


Рисунок 15 – Нахождение среднего арифметического введённых чисел

Прописан полный и упрощённый интерфейс для вывода значений. Завершаю программу.

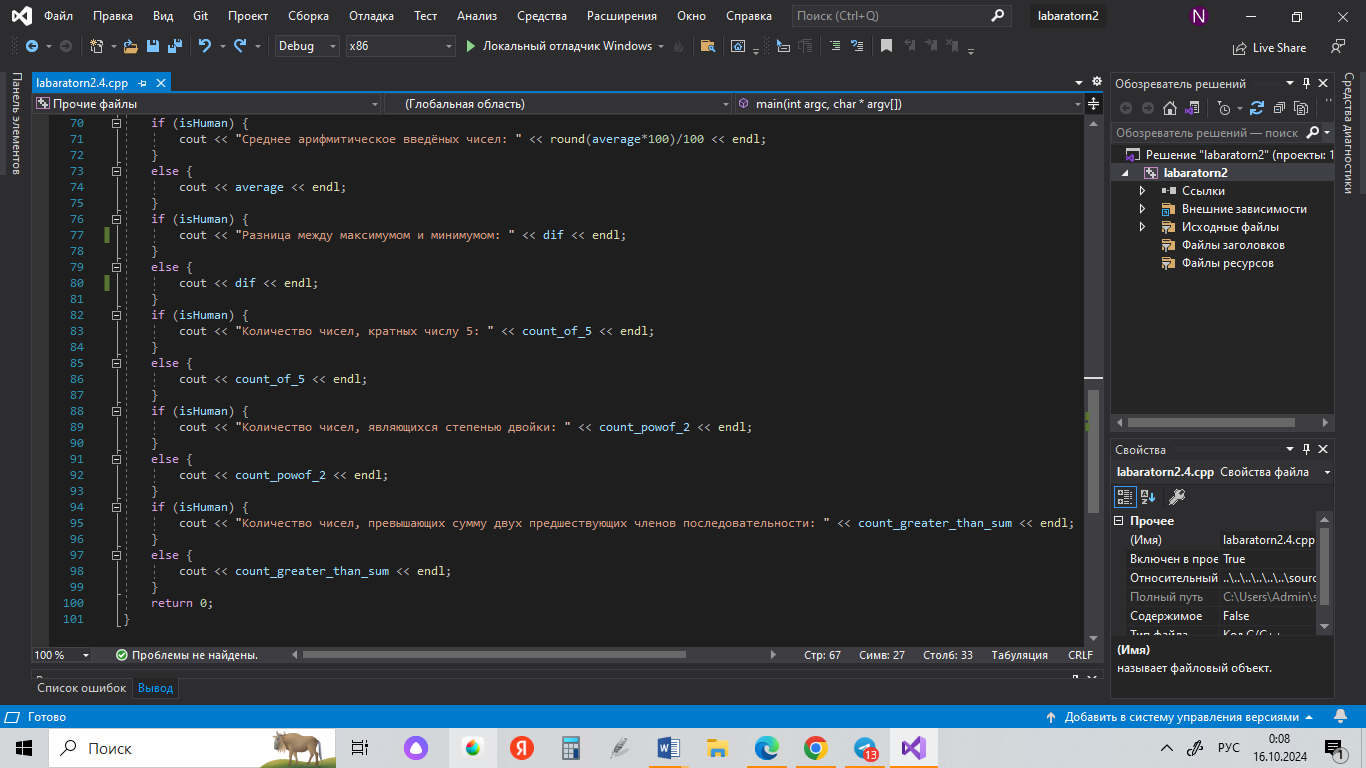


Рисунок 16 – Оформление интерфейсов и конец программы

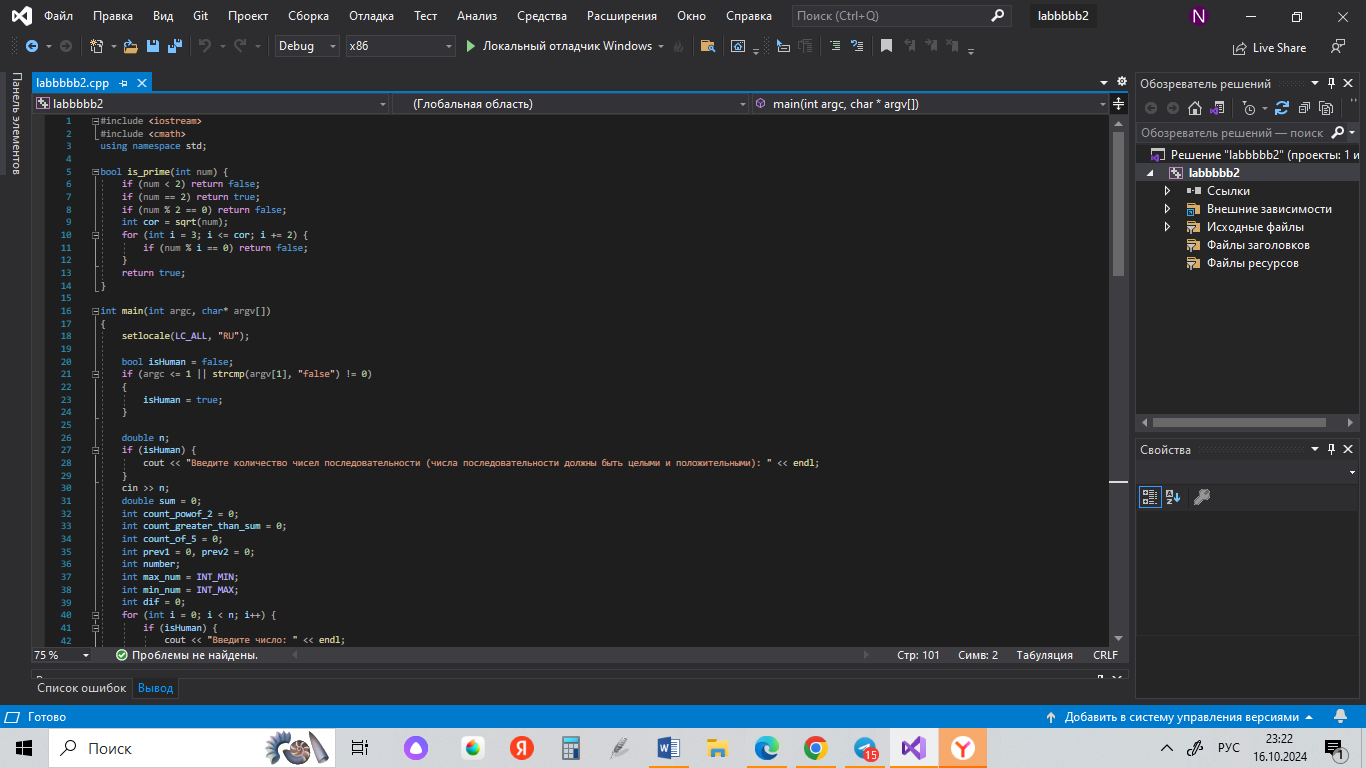


Рисунок 17 – Программа в полном интерфейсе

**Тестирование:**

Было введено целое положительное число, далее была введена последовательность целых положительных чисел.

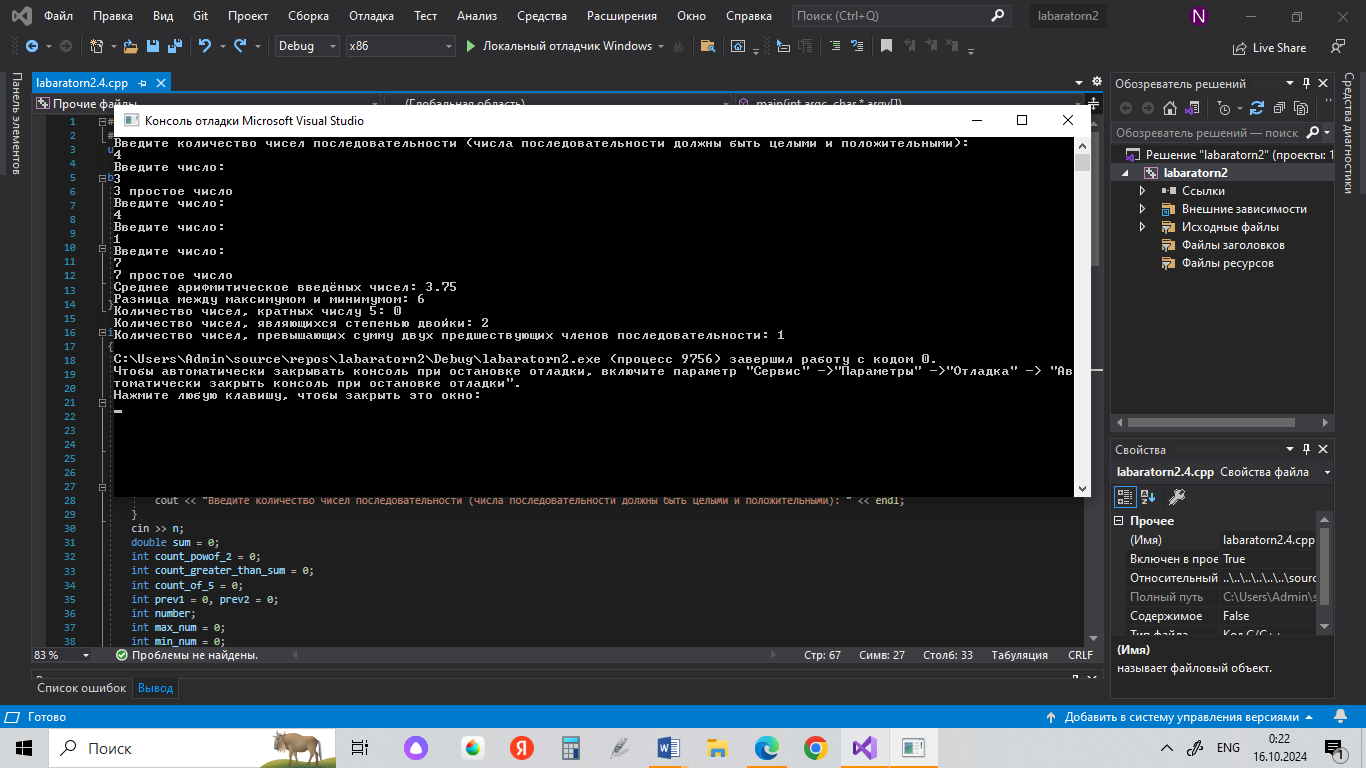


Рисунок 18 – Тестирование программы

**Вывод:**

Был получен опыт работы с анализом числовых последовательностей, а также написана программа, используя язык программирования C++.

Код программы:

#include <iostream>

#include <cmath>

using namespace std;

bool is\_prime(int num) {

if (num < 2) return false;

if (num == 2) return true;

if (num % 2 == 0) return false;

int cor = sqrt(num);

for (int i = 3; i <= cor; i += 2) {

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

}

return true;

}

int main(int argc, char\* argv[])

{

setlocale(LC\_ALL, "RU");

bool isHuman = false;

if (argc <= 1 || strcmp(argv[1], "false") != 0)

{

isHuman = true;

}

double n;

if (isHuman) {

cout << "Введите количество чисел последовательности (числа последовательности должны быть целыми и положительными): " << endl;

}

cin >> n;

double sum = 0;

int count\_powof\_2 = 0;

int count\_greater\_than\_sum = 0;

int count\_of\_5 = 0;

int prev1 = 0, prev2 = 0;

int number;

int max\_num = INT\_MIN;

int min\_num = INT\_MAX;

int dif = 0;

for (int i = 0; i < n; i++) {

if (isHuman) {

cout << "Введите число: " << endl;

}

cin >> number;

sum += number;

if (is\_prime(number)) {

if (isHuman) {

cout << number << " простое число" << endl;

}

else {

cout << number << endl;

}

}

if (number % 5 == 0) {

count\_of\_5++;

}

if ((number >= 1) && (number & (number - 1)) == 0) {

count\_powof\_2++;

}

if (i >= 2 && number > (prev1 + prev2)) {

count\_greater\_than\_sum++;

}

prev2 = prev1;

prev1 = number;

max\_num = max(max\_num, number);

min\_num = min(min\_num, number);

dif = max\_num - min\_num;

}

double average = sum / n;

if (isHuman) {

cout << "Среднее арифмитическое введёных чисел: " << round(average \* 100) / 100 << endl;

}

else {

cout << average << endl;

}

if (isHuman) {

cout << "Разница между максимумом и минимумом: " << dif << endl;

}

else {

cout << dif << endl;

}

if (isHuman) {

cout << "Количество чисел, кратных числу 5: " << count\_of\_5 << endl;

}

else {

cout << count\_of\_5 << endl;

}

if (isHuman) {

cout << "Количество чисел, являющихся степенью двойки: " << count\_powof\_2 << endl;

}

else {

cout << count\_powof\_2 << endl;

}

if (isHuman) {

cout << "Количество чисел, превышающих сумму двух предшествующих членов последовательности: " << count\_greater\_than\_sum << endl;

}

else {

cout << count\_greater\_than\_sum << endl;

}

return 0;

}