**Національний технічний університет України**

**«Київський політехнічний інститут»**

*Факультет інформатики та обчислювальної техніки*

*Кафедра обчислювальної техніки*

**Лабораторна робота №4**

Задане натуральне число *n*. З’ясувати, чи є серед чисел *n*, *n* + 1, …, 2⋅*n* числа-близнюки, тобто прості числа, різниця між якими дорівнює двом. Вивести усі пари таких близнюків.

*Виконав:* Сулима Олександр

*Група:* ІП-53

**2015**

**ЛАБОРАТОРНА РОБОТА №4** *Сулима Олександр, ІП-53*

**І. Ціль роботи**

Задане натуральне число *n*. З’ясувати, чи є серед чисел *n*, *n* + 1, …, 2⋅*n* числа-близнюки, тобто прості числа, різниця між якими дорівнює двом. Вивести усі пари таких близнюків.

**ІІ. Постанова задачі**

Ввести ціле натуральне число n, виконати перевірку на правильність вводу, вивести усі прості числа з діапазону [n; 2n] різниця між якими дорівнює двом.

**ІІІ. Аналітичні викладки**

**Просте число**  — це [натуральне число](https://uk.wikipedia.org/wiki/%D0%9D%D0%B0%D1%82%D1%83%D1%80%D0%B0%D0%BB%D1%8C%D0%BD%D0%B5_%D1%87%D0%B8%D1%81%D0%BB%D0%BE), яке має рівно два різних натуральних [дільники](https://uk.wikipedia.org/wiki/%D0%9F%D0%BE%D0%B4%D1%96%D0%BB%D1%8C%D0%BD%D1%96%D1%81%D1%82%D1%8C) (лише [1](https://uk.wikipedia.org/wiki/1_(%D1%87%D0%B8%D1%81%D0%BB%D0%BE)) і саме число). Решту чисел, окрім одиниці, називають [складеними](https://uk.wikipedia.org/wiki/%D0%A1%D0%BA%D0%BB%D0%B0%D0%B4%D0%B5%D0%BD%D0%B5_%D1%87%D0%B8%D1%81%D0%BB%D0%BE). Таким чином, всі натуральні числа, більші від одиниці, розбивають на прості і складені. [Теорія чисел](https://uk.wikipedia.org/wiki/%D0%A2%D0%B5%D0%BE%D1%80%D1%96%D1%8F_%D1%87%D0%B8%D1%81%D0%B5%D0%BB) вивчає властивості простих чисел. В [теорії кілець](https://uk.wikipedia.org/wiki/%D0%9A%D1%96%D0%BB%D1%8C%D1%86%D0%B5_(%D0%B0%D0%BB%D0%B3%D0%B5%D0%B1%D1%80%D0%B0)) простим числам відповідають [незвідні елементи](https://uk.wikipedia.org/w/index.php?title=%D0%9D%D0%B5%D0%BF%D1%80%D0%B8%D0%B2%D0%BE%D0%B4%D0%B8%D0%BC%D0%B8%D0%B9_%D0%B5%D0%BB%D0%B5%D0%BC%D0%B5%D0%BD%D1%82&action=edit&redlink=1).

**ІV. Перевірка алгоритму**

|  |  |  |
| --- | --- | --- |
| 1. | N = -53; | Програма не буде виконуватися, виведеться повідомлення про помилку. |
| 2. | N = 100; | Програма буде виконуватися вірно і виведе прості числа, різниця між якими дорівнює 2 |

**V. Алгоритм**

**Початок**

Ввести значення n

Якщо n<=0

Вивести повідомлення, що виникла помилка

Першу координату вектора «а» поділити на першу координату вектора «b»

Вивести

повідомлення, що вектори колінеарні

**Кінець**

1 0

Якщо відповідні відношення рівні

0 1

Вивести

повідомлення, що вектори не колінеарні

**VІ. Код програми і приклад**

// Лаба\_4.cpp: определяет точку входа для консольного приложения.

//

#include "stdafx.h"

#include <iostream>

using namespace std;

int Res(int res)

{

int k, m, z;

for (int i = res; i <= (2 \* res); i++) //берет каждое число с промежутка

{

k = 0;

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

{

if (i%q == 0) k += 1;

}

if ((k == 2) || (k == 1)) //выведет только простые числа

{

for (int t = res; t <= (2 \* res); t++)

{

m = 0;

for (int r = 1; r <= t; r++)

{

if (t%r == 0) m += 1;

}

if ((m == 2) || (m == 1))

{

if ((i - t) == 2)

{

cout << i << " ^^ " << t << endl;

}

}

}

}

}

return res;

}

int main()

{

int n;

cout << "Enter n: " << endl;

cin >> n;

if (n <= 0)

{

cout << "Error!" << endl;

system("pause");

return 1;

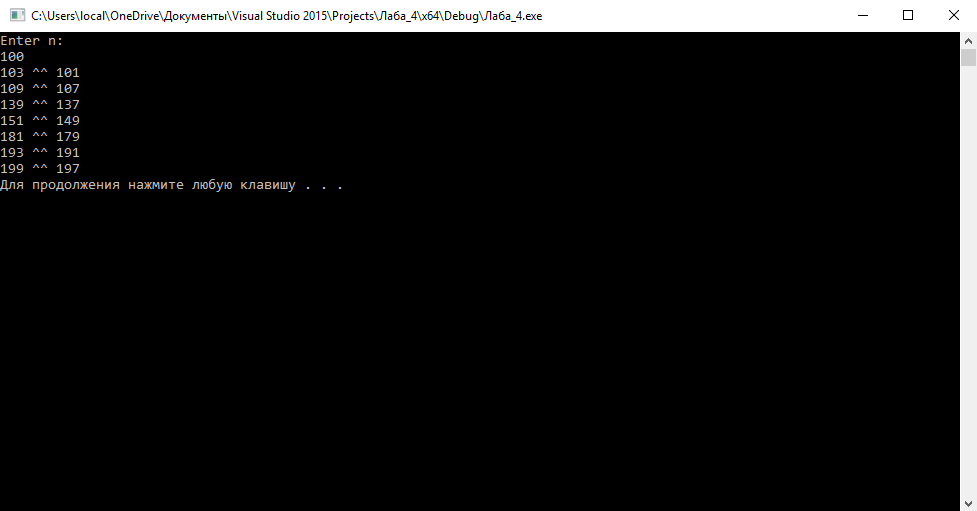
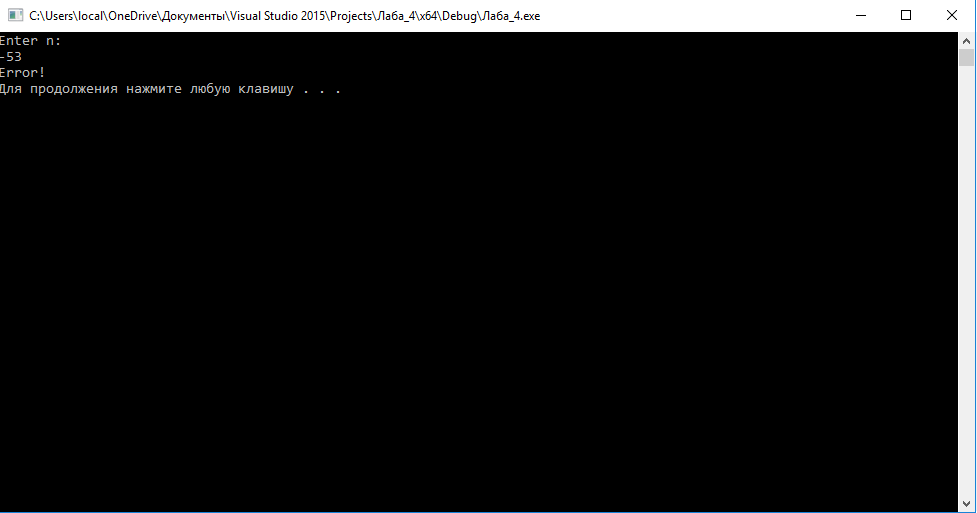
}

Res(n);

system("pause");

return 0;

}



**VІІ. Висновок**

Таким чином, для перевірки, чи є серед чисел *n*, *n* + 1, …, 2⋅*n* числа-близнюки, тобто прості числа, різниця між якими дорівнює двом, є обрахування кожного числа і підбору для нього такого числа, що задовольняє такі потреби:  
-просте число;  
-різниця між ними дорівнює «2»;  
 Якщо такі числа існують, то вони виведуться попарно. В залежності від змінної n, алгоритм має різні відповідні обчислення.