Министерство образования Республики Беларусь

Учреждение образования

"Брестский государственный технический университет"

Кафедра ИИТ

Лабораторная работа №10

За 2ой семестр

По дисциплине: "ОАиП"

Тема: «Пользовательские функции»

Выполнил:

Студент 1-го курса ПО-4

(подгруппа 2)

Левоцкий Н.Д.

Проверила:

Хацкевич М.В.

Брест 2020

**Цель работы:** Изучить основные принципы написания пользовательских функций, ознакомиться с возможностями передачи данных в функции и получения результата по итогам работы функции. Реализовать собственные функции для обработки данных составных и простых типов.

**Задание 1.**

Вариант 19

Описать функцию Norm1(A, M, N) вещественного типа, вычисляющую норму вещественной матрицы A размера M × N:

Norm1(A, M, N) = max {|A1,J| + |A2,J| + … + |AM,J|},

где максимум берется по всем J от 1 до N. Для данной матрицы A размера

M × N найти Norm1(A, K, N), K = 1, …, M.

Код программы:

#include<iostream>

using namespace std;

float Norm1(float\*\* A, int N, int M);

void FillArray(float\*\* A, int N, int M);

void PrintArray(float\*\* A, int N, int M);

int main()

{

setlocale(LC\_ALL, "rus");

int M, N;

cout << "Введите количество строк массива: ";

cin >> N;

cout << "Введите количество столбцов массива: ";

cin >> M;

float\*\* A = new float\* [N];

for (int i = 0; i < N; i++)

A[i] = new float[M];

FillArray(A, N, M);

PrintArray(A, N, M);

cout << "Результат функции: " << Norm1(A, M, N) << endl;

for (int i = 0; i < N; i++)

delete[] A[i];

delete A;

system("color F0");

}

void FillArray(float\*\* A, int N, int M)

{

for (int i = 0; i < N; i++)

{

for (int j = 0; j < M; j++)

{

cout << "[" << i << "]" << "[" << j <<"]";

cin >> A[i][j];

}

}

}

void PrintArray(float\*\* A,int N, int M)

{

cout << "Матрица: " << endl;

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

for (int j = 0; j < M; j++)

cout << "\t" << A[i][j];

cout << endl;

}

}

float Norm1(float\*\* A, int N, int M)

{

float max = 0;

float tmp = 0;

for (int i = 0; i < M; i++)

{

tmp = 0;

for (int j = 0; j < N; j++)

{

tmp += fabs(A[i][j]);

}

if (tmp > max)

{

max = tmp;

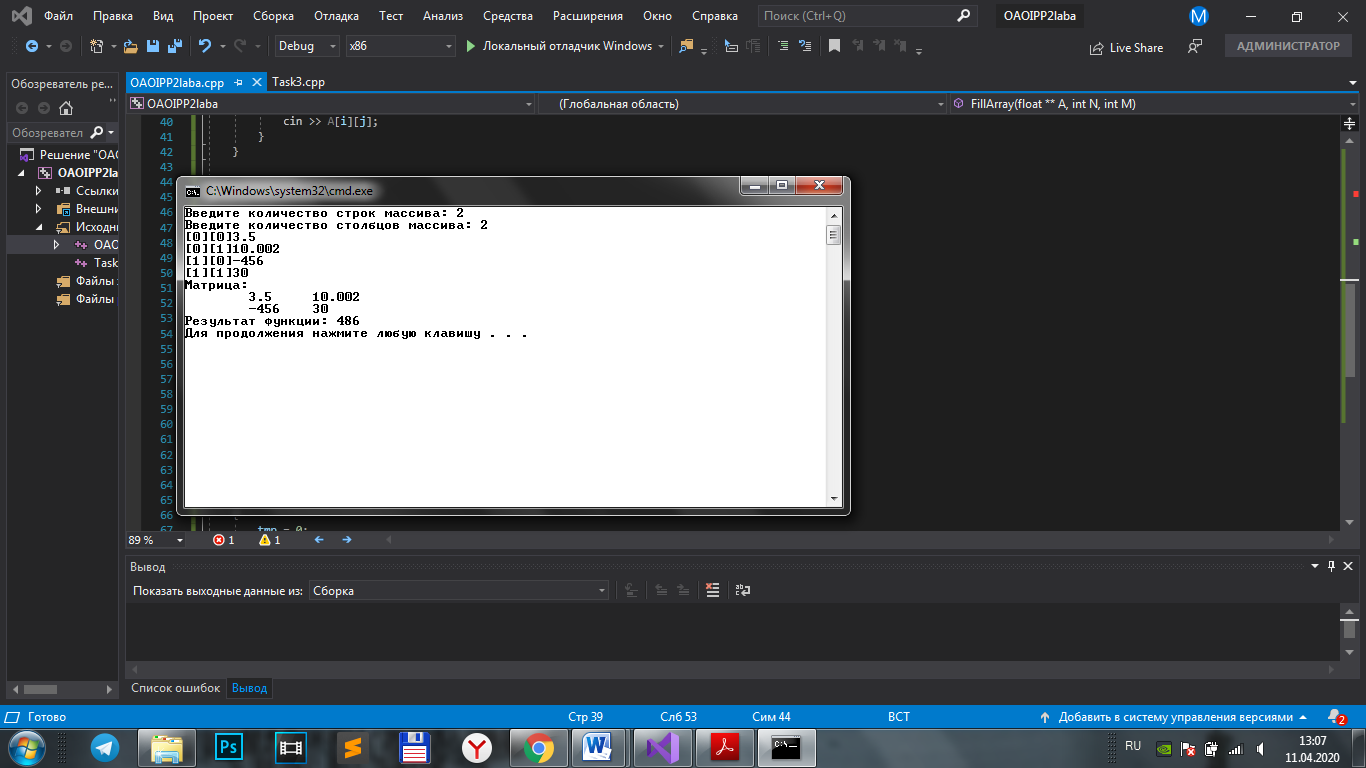
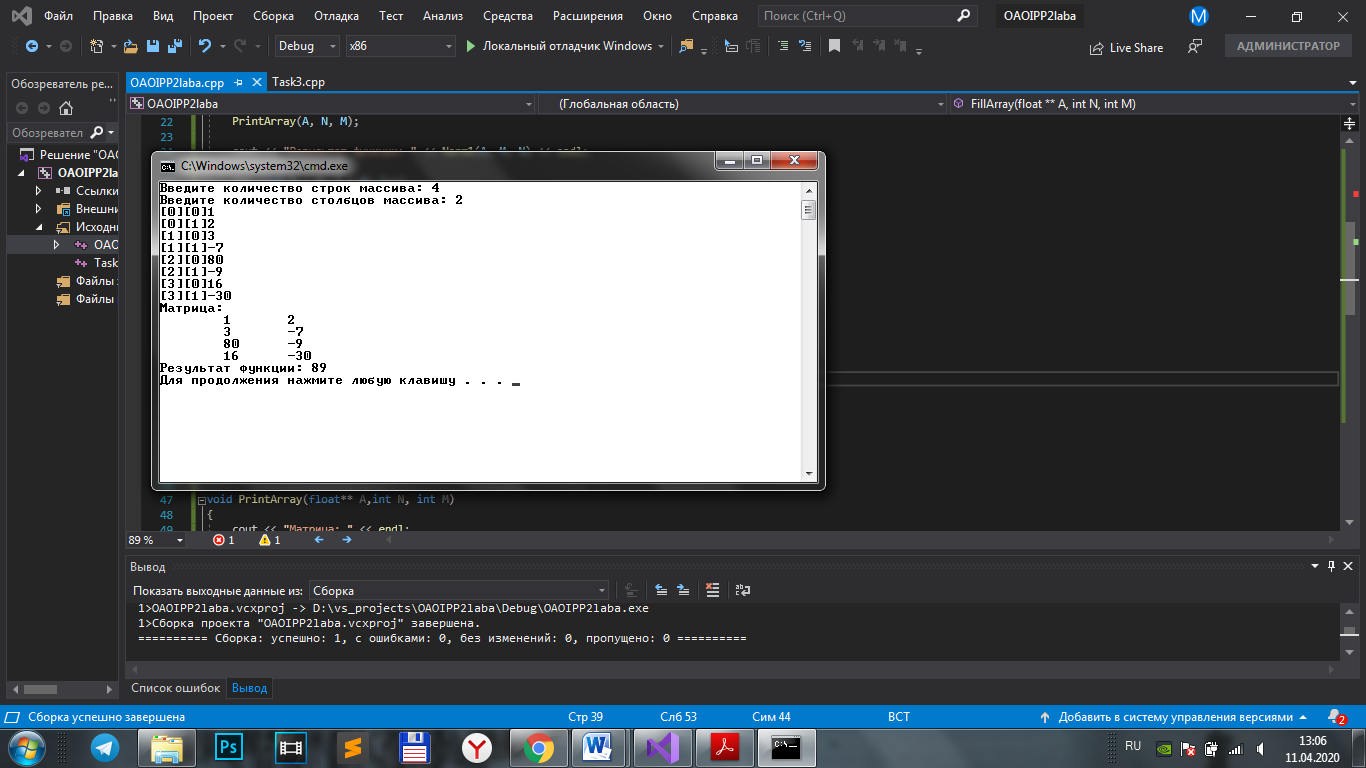
}

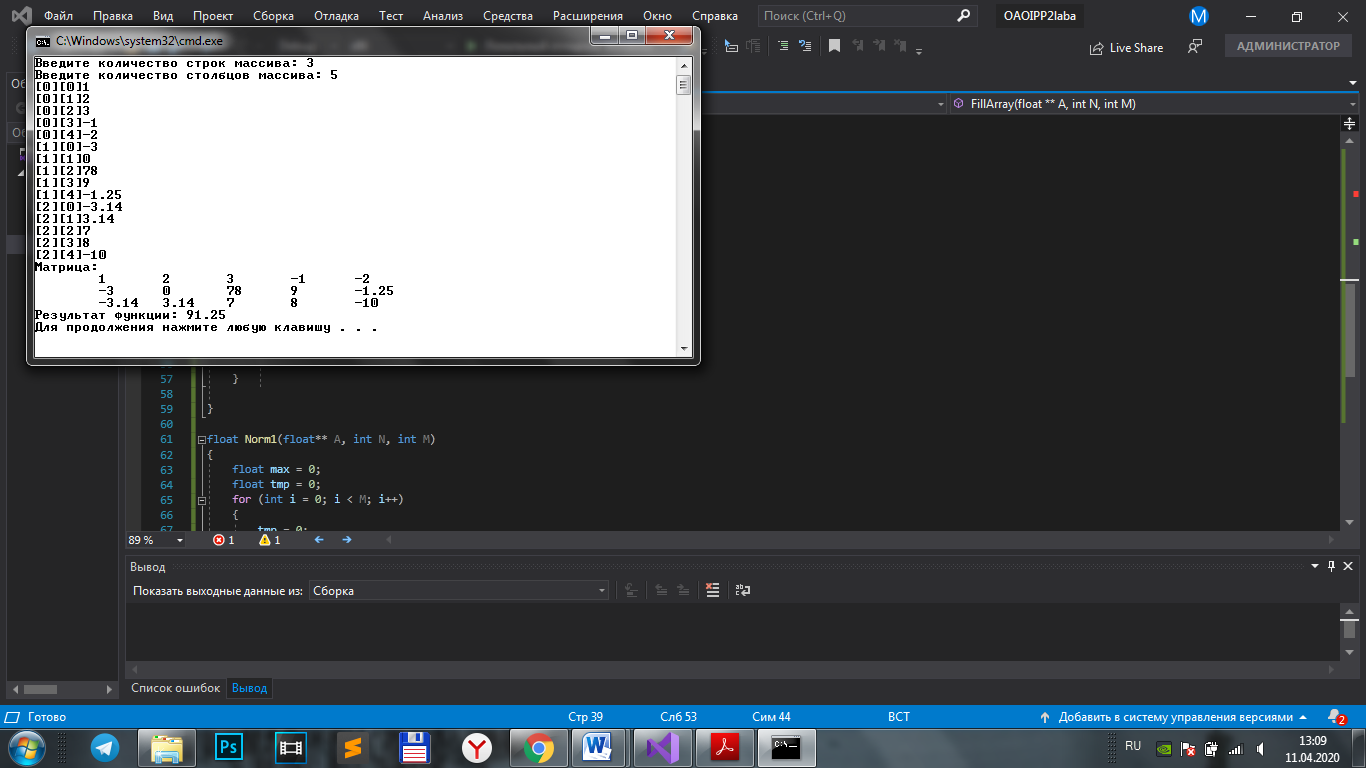
}

return max;

}

Пример (матрицы со всеми возможными соотношениями размеров):





**Задание 2.**

Вариант 18

18.Описать функцию HexToDec(S) целого типа, определяющую целое неотрицательное число по его строковому представлению S в 16-ричной системе счисления. Параметр S имеет строковый тип, состоит из символов «0»–«9», «A»–«F» и не содержит ведущих нулей (за исключением значения «0»). Используя эту функцию, вывести пять чисел, для которых даны их 16-ричные представления.

Код программы:

#include<iostream>

#include<string.h>

using namespace std;

//c использованием аски

int HexToDec(char num[]);

int main()

{

const int N = 20;

char num[N];

gets\_s(num);

cout <<"(16)"<< num <<"==>"<<"(10)"<< HexToDec(num) << endl;

return 0;

}

int HexToDec(char num[])

{

int length = strlen(num);

int base = 1;

int temp = 0;

for (int i = length - 1; i >= 0; i--)

{

if (num[i] >= '0' && num[i] <= '9')

{

temp += (num[i] - 48) \* base;

base = base \* 16;

}

else if (num[i] >= 'A' && num[i] <= 'F')

{

temp += (num[i] - 55) \* base; /\*-55 потому что 65 это представление “А” в десятичной сс, а тк в 16ричной сс “A”(и остальные до “F”) больше 10, поэтому от 65 отнимаем 10\*/

base = base \* 16;

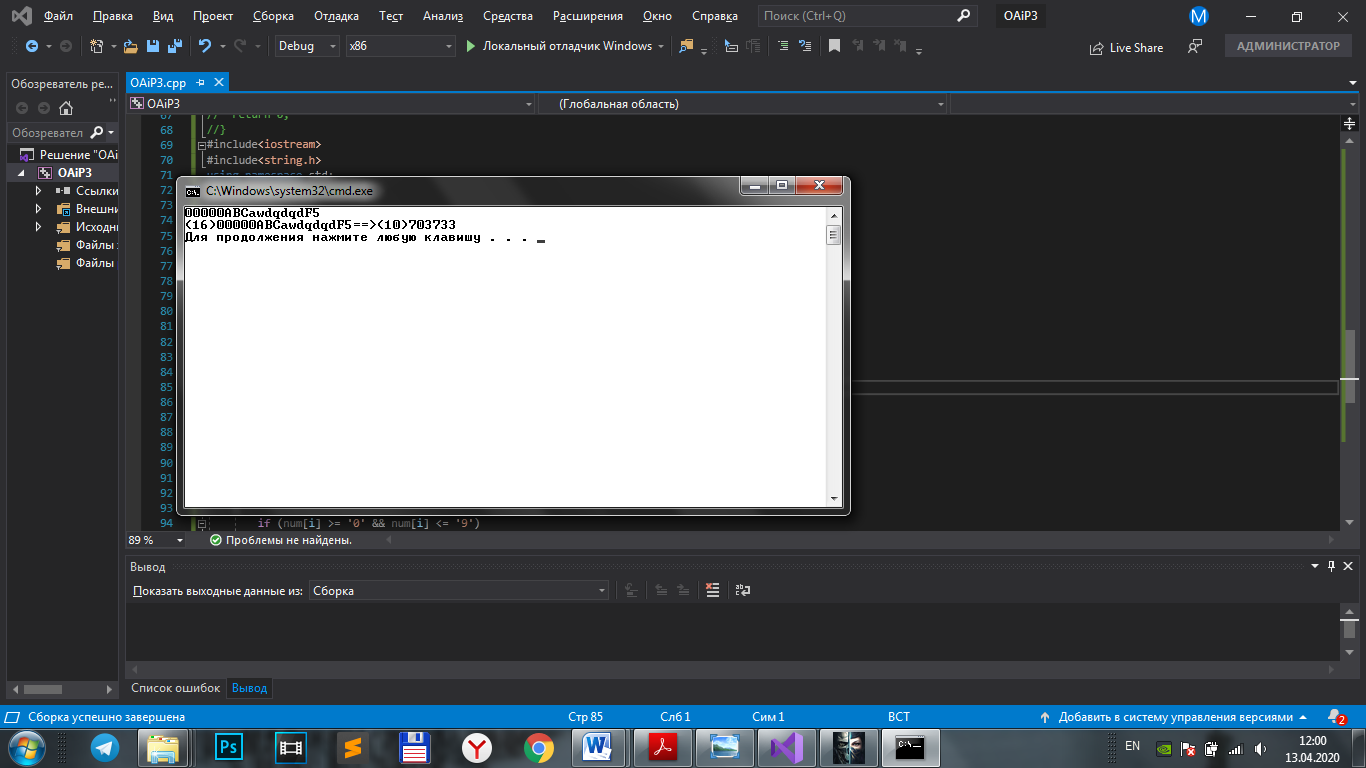
}

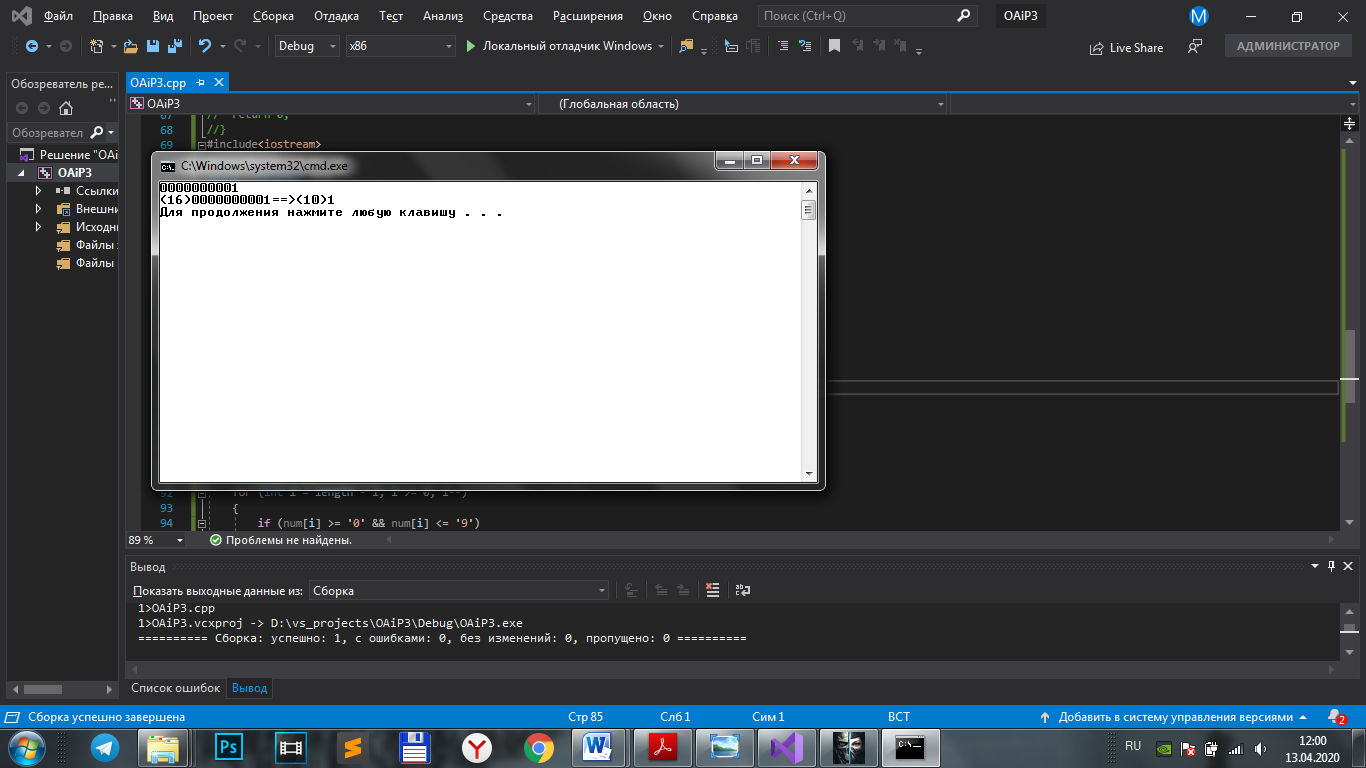
}

return temp;

}

Примеры:





И тд

**Задание 3.**

Вариант 4.

Описать рекурсивную функцию Fib1(N) целого типа, вычисляющую

N-й элемент последовательности чисел Фибоначчи (N — целое число):

F1 = F2 = 1, FK = FK–2 + FK–1, K = 3, 4, … .

С помощью этой функции найти пять чисел Фибоначчи с данными номерами, и вывести эти числа вместе с количеством рекурсивных вызовов функции Fib1, потребовавшихся для их нахождения.

Код программы:

#include <iostream>

#include <stdio.h>

using namespace std;

int Fib1(int N);

int main()

{

setlocale(LC\_ALL, "rus");

int counter = 0;

int N;

cin >> N;

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

{

Fib1(i);

counter++;

}

cout<<N<<" член последовательности равен: "<< Fib1(N)<< endl;

cout << "Количество вызовов " << counter<<endl;

system("color F0");

return 0;

}

int Fib1(int N)

{

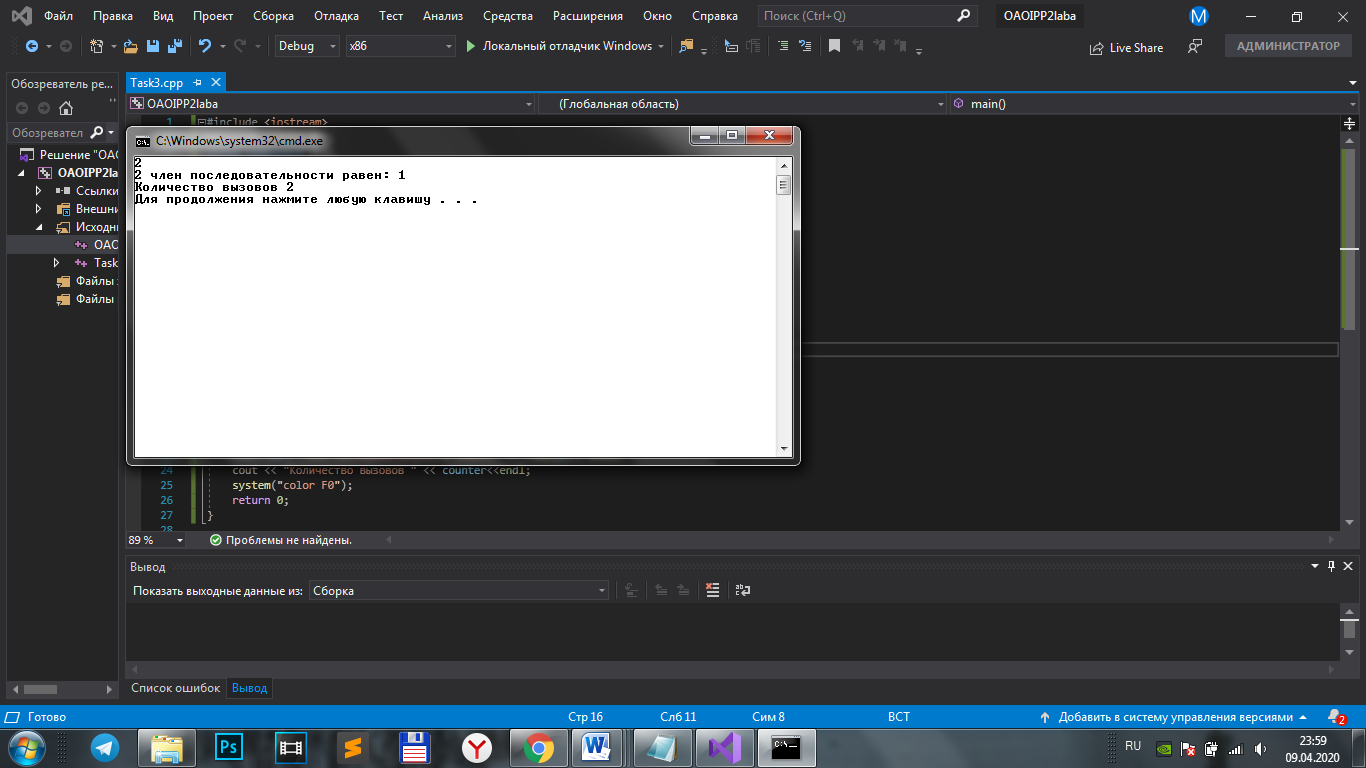
if (N == 1 || N == 2)

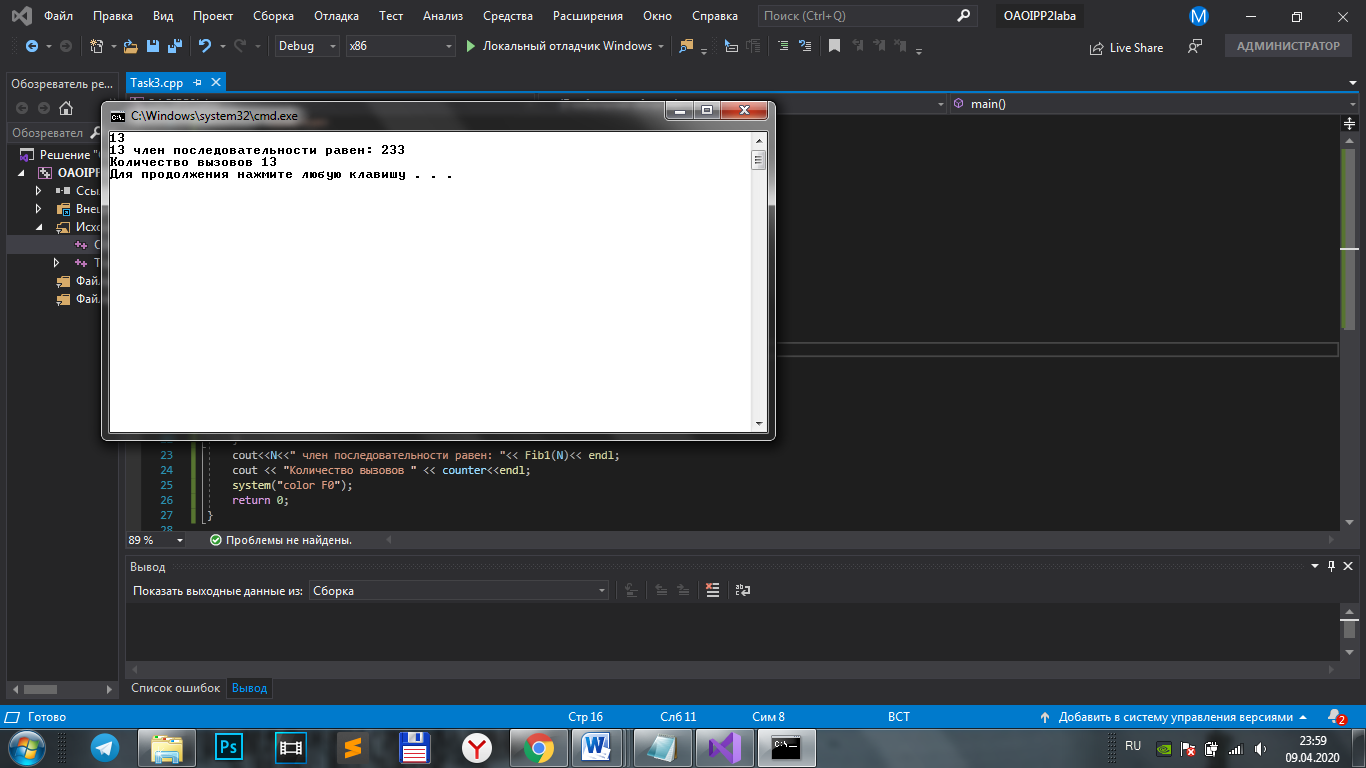
return 1;

return Fib1(N - 1) + Fib1(N - 2);

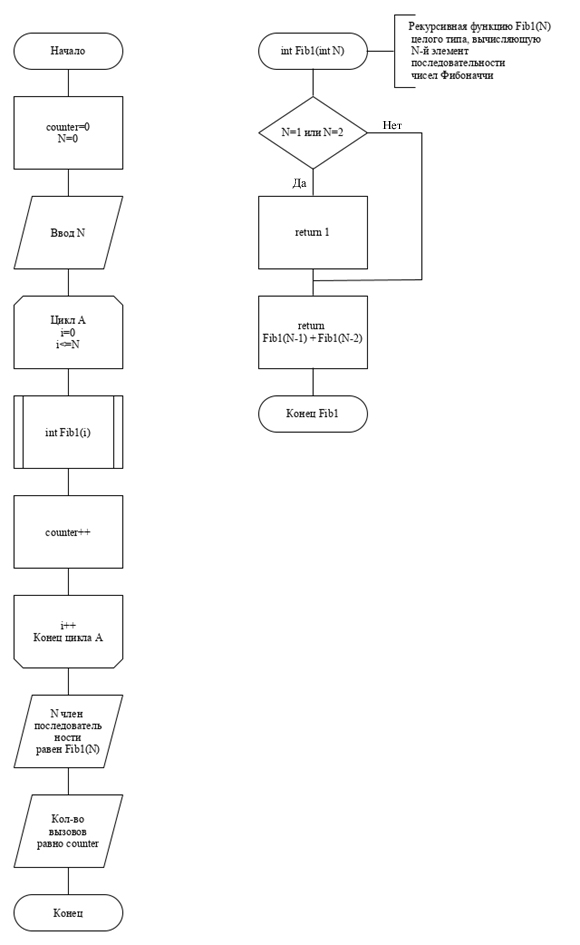
}

Примеры:





Блок-схема к заданию 3 на следующей странице



**Вывод:** Я изучил основные принципы написания пользовательских функций, ознакомился с возможностями передачи данных в функции и получения результата по итогам работы функции. Реализовад собственные функции для обработки данных составных и простых типов.