ГУАП

КАФЕДРА № 33

ОТЧЕТ  
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ассистент |  |  |  | А.И. Дубинин |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №2 |
| ИСПОЛЬЗОВАНИЕ ОБРАБОТЧИКОВ ИСКЛЮЧЕНИЙ ПРИ РАБОТЕ С МАССИВАМИ |
| по курсу: Технологии и методы программирования |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. | 3136 |  |  |  | В.А. Панчук |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2023 г.

Оглавление

[1. Цель лабораторной работы 3](#_Toc128525447)

[2. Постановка задачи 3](#_Toc128525448)

[3. Часть 1(С++) 3](#_Toc128525449)

[3.1. Листинг программы 4](#_Toc128525450)

[3.2. Пример работы программы 5](#_Toc128525451)

[4. Часть 2(С#) 6](#_Toc128525452)

[4.1. Задача 1 6](#_Toc128525453)

[4.1.1. Листинг программы 7](#_Toc128525454)

[4.1.2. Пример работы программы 8](#_Toc128525455)

[4.2. Задача 2 и 3 9](#_Toc128525456)

[4.2.1. Листинг программы 10](#_Toc128525457)

[4.2.1. Пример работы программы 12](#_Toc128525458)

# Цель лабораторной работы

Получить навыки работы обработки исключений при работе с массивами, формируемыми в различных режимах. изучить средства среды программирования Microsoft Visual Studio Community 2019 для создания статистических библиотек (файлов формата \*.lib), содержащих описание ASM-модулей с поддержкой языка С/С++/С#, и их включения в основной проект.

# Постановка задачи

В лабораторной работе, задания из первой части (с использованием языков Assembler и C++) выполнялись под вариантом 3, а из второй(С#) – под вариантом 14.

В первой части лабораторной работы нужно было реализовать алгоритм преобразования информации с использованием логических выражений на языке C++ или C. Также необходимо было применить регистровый класс памяти при описании переменных.

Вторая часть лабораторной работы включала в себя разработку трех программ на языки C#. Каждая программа должна была быть выполнена в двух вариантах: с использованием консольного интерфейса и интерфейса Windows Forms.

Первая программа должна поразрядно складывать две строки (фамилия и имя студента) с выводом промежуточных результатов.

Вторая программа должна найти все натуральные числа от 1 до 200, у которых количество делителей равняется числу, введенному с клавиатуры.

Третья программа должна найти все четырехзначные числа, в которых есть две одинаковые цифры.

Во всех трех программах входные значения должны проверяться на наличие ошибок.

# Часть 1(С++)

Задача: Реализовать процедуры умножения знакового целого числа (вводится с клавиатуры) на число, равное степени 2, с использованием побитовой операции сдвига влево <<.

Блок схема алгоритма данной задачи представлена на рис.1.

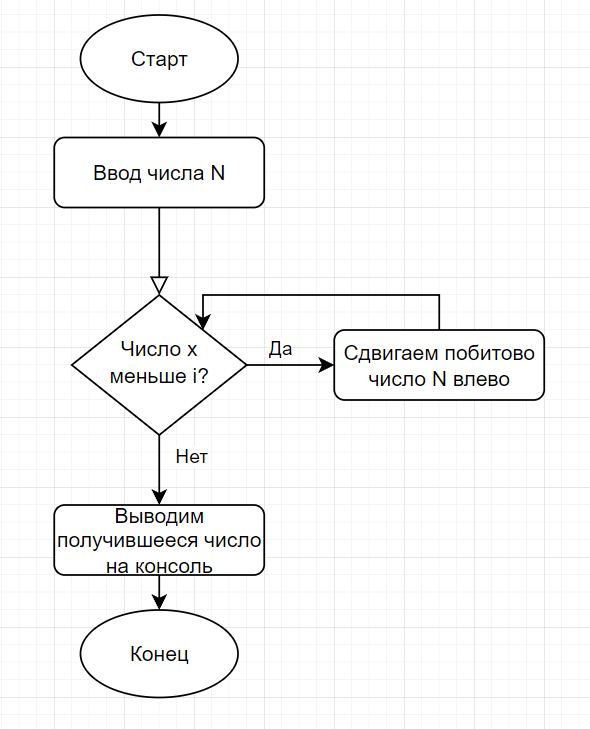


Рисунок 1− процедура умножения знакового целого числа на число, равное степени 2

## Листинг программы

#include <iostream>

using namespace std;

int main() {

const int x = 32;

register int N;

cin >> N;

for (int i = 1; i < x; i\*=2)

{

N = N << 1;

}

cout << N;

return 0;

}

## Пример работы программы

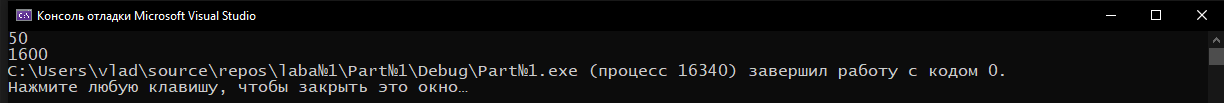
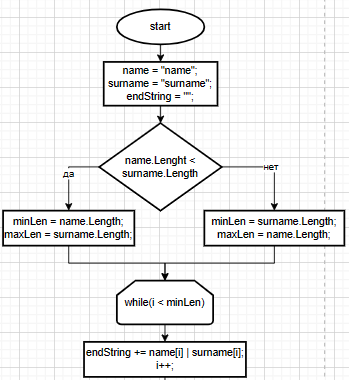


Рисунок 2 − Пример программы 1

# Часть 2(С#)

## Задача 1

Создать консольный и Windows Forms проекты, выполняющие поразрядное сложение двух строк (вашей фамилии и имени) с выводом промежуточных результатов.

Алгоритм поразрядного сложения двух строк представлен на рис 3.

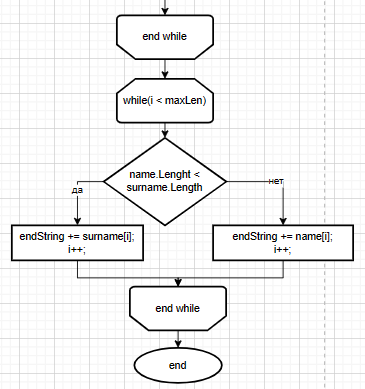


Рисунок 3 − поразрядное сложение строк

## Листинг программы

1. Консольный проект:

using System;

namespace Part\_2

{

class Program

{

static int Main()

{

String name, surname, result = "";

name = Console.ReadLine();

surname = Console.ReadLine();

foreach (var letter in name) if (!char.IsLetter(letter)) return -1;

foreach (var letter in surname) if (!char.IsLetter(letter)) return -1;

int i = 0;

for (; i < name.Length && i < surname.Length; i++)

{

int ok = name[i] + surname[i];

Console.WriteLine($"число: {ok} \tсимвол:{(char)ok}.");

result += (char)ok;

}

if (name.Length > surname.Length) result += name.Substring(i);

else result += surname.Substring(i);

Console.WriteLine(result);

return 0;

}

}

}

1. Отрывок Windows Forms проекта:

private void startButton\_Click(object sender, EventArgs e)

{

String name, surname, result = "";

name = textBox2.Text;

surname = textBox1.Text;

foreach (var letter in name) if (!char.IsLetter(letter)) return;

foreach (var letter in surname) if (!char.IsLetter(letter)) return;

int i = 0;

for (; i < name.Length && i < surname.Length; i++)

{

int ok = name[i] + surname[i];

result += (char)ok;

}

if (name.Length > surname.Length) result += name.Substring(i);

else result += surname.Substring(i);

Result.Text = result;

}

## Пример работы программы

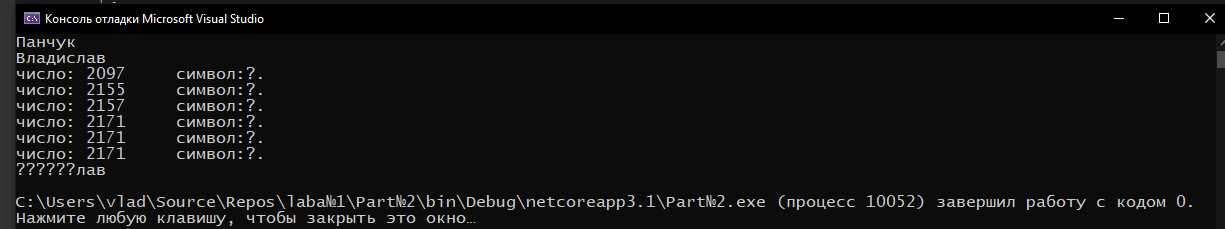


Рисунок 4 − Пример программы 2 (консоль)

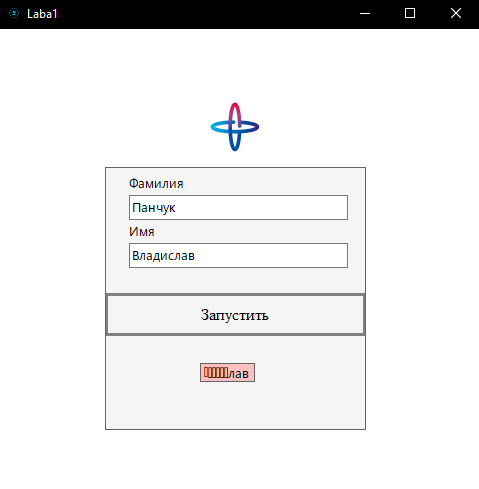


Рисунок 5 − Пример программы 2 (Windows Forms)

## Задача 2 и 3

Найти все натуральные числа из промежутка от 1 до 200, у которых количество делителей равно N (N вводить с клавиатуры).

Найти все натуральные числа А, В и С из интервала от 1 до 20, для которых выполняется равенство А + В2=С2 .

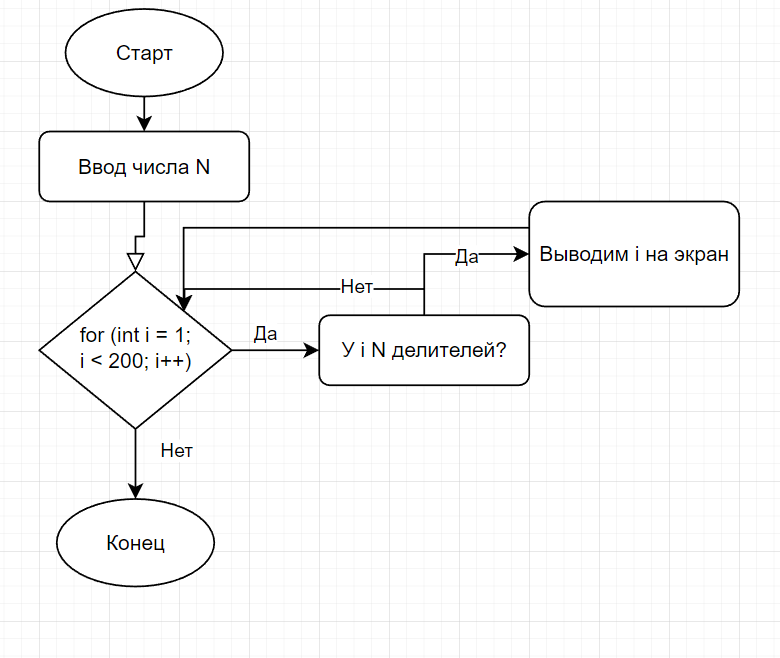
Алгоритм для выполнения данных задач представлен на рисунке 6 и 7.

Рисунок 6 – Алгоритм решения второй задачи

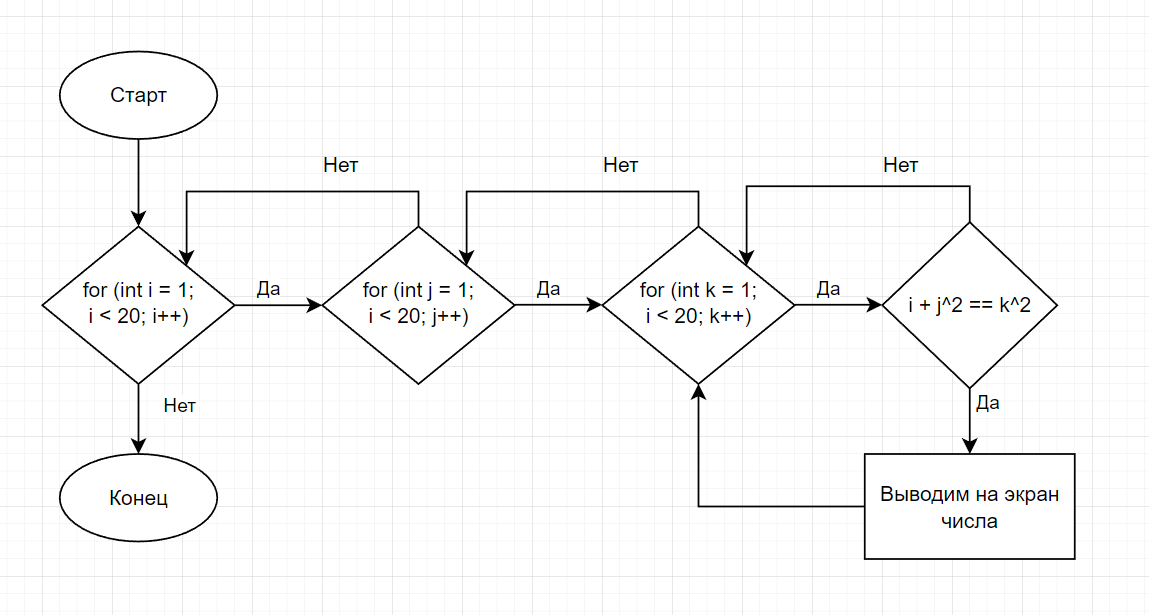


Рисунок 7 – Алгоритм решения третьей задачи

## Листинг программы

1. Консольный проект:

using System;

namespace Part\_3

{

class Program

{

static void Main(string[] args)

{

Console.Write($"Program №1:");

number1();

Console.Write($"\nProgram №2:\n");

number2();

}

static int number1()

{

int max = 200;

string userString = Console.ReadLine();

int intValue = Convert.ToInt32(userString);

if ((intValue > 50) || (intValue < 0)) return -1;

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

{

if (intValue == find\_divisor(i))

{

Console.Write($"{i}\t");

}

}

return 0;

}

private static int find\_divisor(int div)

{

if (div == 1 || div == -1) return 1;

int count = 2;

for (int i = 2; i <= (int)Math.Sqrt(div);i++)if(div % i == 0)count+= 2;

if (((int)Math.Sqrt(div) == Math.Sqrt(div)) && (div%(int)Math.Sqrt(div)== 0)) count--;

return count;

}

static void number2()

{

for (int i = 1; i < 20; i++)

{

for (int j = 1; j < 20; j++)

{

for (int k = 1; k < 20; k++)

{

if ((i + j \* j)== (k \* k))

{

Console.WriteLine($"{i} {j} {k}");

}

}

}

}

}

}

}

Отрывок Windows Forms проекта:

private void startButton\_Click(object sender, EventArgs e)

{

label2.Text = "Program №1:\n";

number1(label2, textBox1);

label3.Text = "Program №2:\n";

number2(label3);

}

static int number1(Label label, TextBox text)

{

int max = 200;

string userString = text.Text;

int intValue;

bool succes = int.TryParse(userString, out intValue);

if ((intValue > 50) || (intValue < 0) || !succes) return -1;

int count = 1;

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

{

if (intValue == find\_divisor(i))

{

if ((count++) % 10 == 0) label.Text += "\n";

label.Text += $"{i} ";

}

}

return 0;

}

private static int find\_divisor(int div)

{

if (div == 1 || div == -1) return 1;

int count = 2;

for (int i = 2; i <= (int)Math.Sqrt(div);i++)if(div % i == 0)count+= 2;

if (((int)Math.Sqrt(div) == Math.Sqrt(div)) && (div%(int)Math.Sqrt(div)== 0)) count--;

return count;

}

static void number2(Label label)

{

for (int i = 1; i < 20; i++)

{

for (int j = 1; j < 20; j++)

{

for (int k = 1; k < 20; k++)

{

if ((i + j \* j) == (k \* k))

{

label.Text += $"{i} {j} {k}\n";

}

}

}

}

}

## Пример работы программы

Рисунок 9− Пример программы 3 (Windows Forms)

Рисунок 8− Пример программы 3 (консоль)

# Вывод

Получили навыки работы с основными инструментами среды программирования Microsoft Visual Studio 2022. Создали программы на языках С#, С++, C, и научились работать с Windows Forms.