

Вариант №8

Цель: получить навыки работы с основными инструментами среды программирования Microsoft Visual Studio 2019. Создать программу с использованием Windows Forms на языках C#, C++, C.

Закрепить теоретические знания основных конструкций языка C/C++. Сформировать умения по составлению программ прикладного назначения, в частности преобразования информации с использованием логических выражений. Получить навыки в реализации простейших симметричных алгоритмов шифрования/дешифрования информации.

Часть 1.

Задание: реализовать механизм шифрования и расшифрования символьной информации мультипликативным конгруэнтным методом.

Ход работы:

Программа считывает данные из файла, который заранее указан пользователем. Данные из файла записываются в строку, которая далее преобразуется в массив символов. Далее создается отдельный элемент класса «algorithm», в котором генерируется ключевая последовательность с помощью мультипликативного конгруэнтного метода. После этого в отдельной функции класса «algorithm», происходит сложение кодов ASCII элементов символьного массива с элементами массива ключа. Перед сложением элементы обоих массивов переводят в двоичную систему счисления. После операции сложения элементов двух массивов, результат переводят из двоичной в десятичную систему счисления и каждому элементу хранящему какое-то число ставится в соответствие символ из таблицы ASCII. Таким образом получается закодированная последовательность, хранящаяся в массиве символов.

Результат работы программы – начальная строка, закодированная строка и раскодированная строка записываются в файл, который заранее указал пользователь.

Листинг C++ программы:

//Main

```
#include <iostream>

#include <string>
#include "readFile.cpp"
#include "algorithm.cpp"

int main () {
    std:: string string1 = "/Users/Kirill/Desktop/in.txt";
    std:: string string2 = "/Users/Kirill/Desktop/out4.txt";
    readFile newFile(string1,string2);
    algorithm newAlg (newFile.getSize());

    newAlg.process(newFile.getText(), newAlg.coded).
    newAlg.process(newAlg.coded, newAlg.decoded).

    newFile.saveFile(newFile.surname, newAlg.coded, newAlg.decoded).

}
```

//Algorithm

```
#include <cstdlib>
#include <iostream>
#include <string>
#include <cmath>

class algorithm {
public:
    int s=0.
    const int A = 13.
    const int C = 0.
    int* T = new int [10].
    unsigned char* coded = new unsigned char [10].
    unsigned char* decoded = new unsigned char [10].

    algorithm (int size) {
        s = size.
        T = (int*) malloc (sizeof (int) * size).
        coded = (unsigned char *) malloc (sizeof (unsigned char) * s).
        decoded= (unsigned char *) malloc (sizeof (unsigned char) * s).

        int m = pow (2.0, s).
        T [0] = 3.

        for (int i=0; i<s; i++) {
            T[i+1] = (A * T[i] + C) % m.
        }
    }

    std: string bin (int number) const {
        std: string r.
        while (number >= 1) {
            int m = number % 2.
            if (m == 0) {
                r = '0' + r.
            } else {
                r = '1' + r.
            }
        }
    }
}
```

```

        }
        number /= 2.
    }

    while (r. length () < s) {
        r = '0' + r.
    }
    return r.
}

int dec (std: string b) const {
    int r = 0.
    for (int i=0; i<s; i++) {
        if (b[i] == '1') {
            int power = s-i-1.
            r += pow (2.0, power).
        }
    }
    return r.
}

std: string sum (std: string a, std: string b) const {
    std: string r.
    for (int i=0; i<s; i++) {
        if (a[i] == b[i])
        {
            r += '0'.
        } else {
            r += '1'.
        }
    }

    while (r. length () < s) {
        r = '0' + r.
    }
    return r.
}

void process (unsigned char* str, unsigned char* result) {
    for (int i=0; i<s; i++)
    {
        result[i] = (char) dec(sum(bin(str[i]), bin(T[i+1]))).
    }
    result[s] = 0.
    std: cout<< "coded = "<< result<<std: endl.
}

};

```

// readFile

```

#include <iostream>
#include <string>
#include <cstdlib>
#include <fstream>
#include <utility>

class readFile {
public: std: string fileIn.
       std: string fileOut.
       int letters = 0.

```

```

        unsigned char* surname = new unsigned char [10].

readFile (std: string nameIn, std: string nameOut) {
    fileIn = nameIn.
    fileOut = nameOut.

}

int getSize () {
    std: ifstream infile (fileIn).
    int count = 0.
    char ch.
    if (! infile.is_open ()) {
        std: cout << "can't open file \n».
    } else {
        while (! infile.eof ()) {
            infile>>ch.
            count++.
        }
    }
    infile. close ().
    letters = count - 1.
    return letters.

};

unsigned char* getText () {
    std: ifstream infile (fileIn).
    surname = (unsigned char *) malloc (sizeof (unsigned char) * letters).

    for (int i = 0; i < letters; ++i) {
        char ch.
        infile >> ch.
        surname[i] = ch.
    }
    infile. close ();
    return surname;
}

void saveFile (unsigned char* text1, unsigned char* text2, unsigned char* text3) {
    std: ofstream outfile (fileOut).
    outfile<<text1<< std: endl.
    outfile<<text2<< std: endl.
    outfile<<text3<< std: endl.
    outfile. close ();
}

};

```

In.txt:

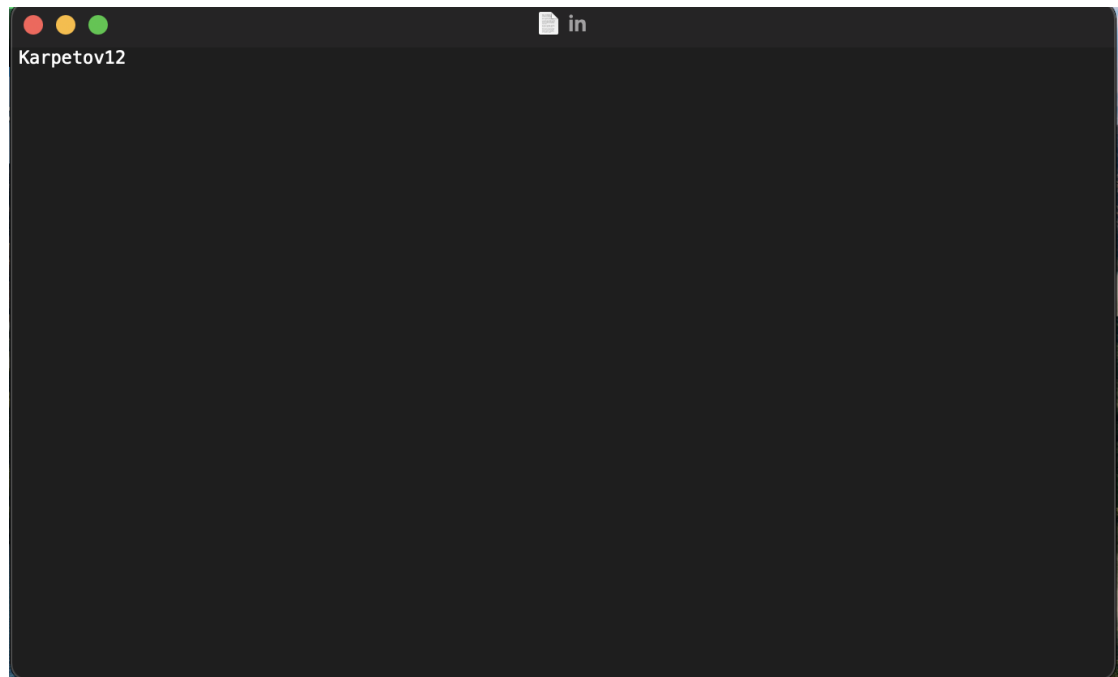


Рис. 1

Out4.txt

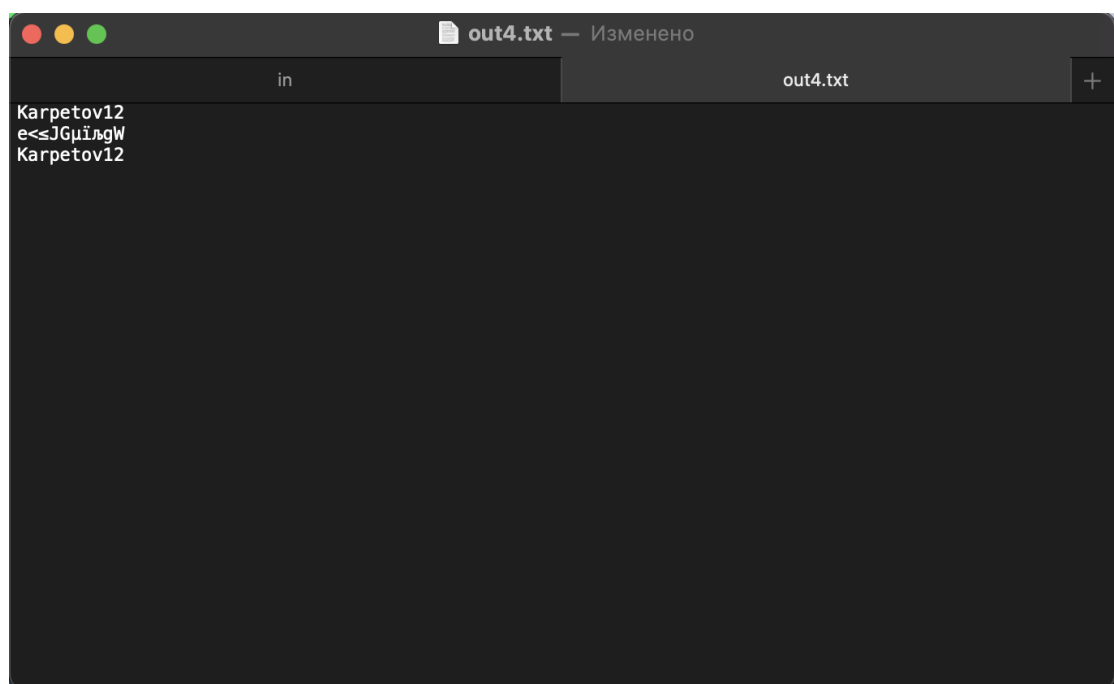


Рис. 2

Часть 2.

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

Ход работы: при запуске программы перед пользователем появляется окно с полями для ввода данных (Рис.3). Пользователю необходимо ввести свои имя и фамилию. При неправильном вводе данных, перед пользователем появится окно, сообщающее об ошибке (Рис.4), и все неправильно введенные данные будут стерты. При попытке пользователя нажать на кнопку «result», не заполнив все поля окна, результат не будет выведен. После нажатия на кнопку «result», программа произведет поразрядное сложение имени и фамилии. Результат будет выведен пользователю (Рис.5).

Листинг Windows forms программы:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace summ
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
        string name;
        string surname;

        private void textBox1_TextChanged(object sender, EventArgs e)
        {
            surname = textBox1.Text;

            for (int i = 0; i < surname.Length; i++)
            {
                if (surname[i] < 65 || surname[i] > 175)
                {
                    textBox1.Clear();
                    MessageBox.Show("enter again");
                }
            }
            byte[] asciiBytes = Encoding.ASCII.GetBytes(surname);
            string textSurname = System.Text.Encoding.ASCII.GetString(asciiBytes);
        }
    }
}
```

```

        textBoxCodeSurname.Text = String.Join(" ", Array.ConvertAll(asciiBytes,
byteValue => byteValue.ToString()));
    }

    private void textBox2_TextChanged(object sender, EventArgs e)
    {
        name = textBox2.Text;
        for (int i = 0; i < name.Length; i++)
        {
            if (name[i] < 65 || name[i] > 175 )
            {
                textBox2.Clear();
                MessageBox.Show("enter again");
            }
        }

        byte[] asciiBytesSurname = Encoding.ASCII.GetBytes(name);

        string textSurname = System.Text.Encoding.ASCII.GetString(asciiBytesSurname);
        textBoxCodeName.Text = String.Join(" ", Array.ConvertAll(asciiBytesSurname,
byteValue => byteValue.ToString()));
    }

    private void label2_Click(object sender, EventArgs e)
    {
    }

    private void Form1_Click(object sender, EventArgs e)
    {
    }

    private void button1_MouseClick(object sender, MouseEventArgs e)
    {
        var sbSurname = new StringBuilder(surname);
        var sbName = new StringBuilder(name);

        Console.WriteLine($"sbSurname={sbSurname}");
        Console.WriteLine($"sbName={sbName}");
        Console.WriteLine($"surname={surname}");
        Console.WriteLine($"name={name}");

        int size = 0;
        int sizeMin = 0;
        bool flag = false;

        if (sbName.Length <= sbSurname.Length)
        {
            size = sbSurname.Length;
            sizeMin = sbName.Length;
            flag = false;
            Console.WriteLine($"size={size}");
            Console.WriteLine($"sizeMin={sizeMin}");
        }
        else
        {
            size = sbName.Length;
            sizeMin = sbSurname.Length;
            flag = true;
        }

        int[] result = new int[size];

```

```

char[] resultChar = new char[size];
for (int i = 0; i < size; i++)
{
    if (flag == false)
    {
        result[i] = sbSurname[i];
    }
    else
    {
        result[i] = sbName[i];
    }
}

for (int i = 0; i < sizeMin; i++)
{
    result[i] = (sbSurname[i] + sbName[i]);
}

for (int i = 0; i < size; i++)
{
    resultChar[i] = (char)result[i];
}

string s = new string(resultChar);
if(!String.IsNullOrEmpty(textBox1.Text) &&
!String.IsNullOrEmpty(textBox2.Text)) {
    textBox4.Text = s ;

    String.Join("", new List<int>(result).ConvertAll(i =>
i.ToString()).ToArray());
    string resultString = string.Join(" ", result);
    textBox3.Text = resultString;
}
}
}

```

Windows forms:

Сложение

| | |
|---------------------------------------|----------------------|
| sumame | sumame code |
| <input type="text"/> | <input type="text"/> |
| name | name code |
| <input type="text"/> | <input type="text"/> |
| result | result code |
| <input type="text"/> | <input type="text"/> |
| <input type="button" value="result"/> | |

Рис. 3

Сложение

| | |
|---------------------------------------|---|
| sumame | sumame code |
| <input type="text" value="karpetov"/> | <input type="text" value="107 97 114 112 101 116 111 118"/> |
| name | name code |
| <input type="text"/> | <input type="text"/> |
| result | result code |
| <input type="text"/> | <input type="text"/> |
| <input type="button" value="result"/> | |

enter again

OK

Рис. 4

Сложение

| | |
|---------------------------------------|--|
| surname | surname code |
| <input type="text" value="karpetov"/> | <input type="text" value="107 97 114 112 101 116 111 118"/> |
| name | name code |
| <input type="text" value="kinill"/> | <input type="text" value="107 105 114 105 108 108"/> |
| result | result code |
| <input type="text" value="kirymlor"/> | <input type="text" value="107 105 114 121 109 124 111 118"/> |
| <input type="button" value="result"/> | |

Рис. 5

Листинг консольного проекта:

```
using System;
using System.Collections.Generic;
using System.Text;

namespace lab1
{
    class MainClass
    {
        public static void Main(string[] args)
        {

            int size = 0;
            int sizeMin = 0;
            bool flag = false;
            bool check = false;
            Console.Write("Введите surname: ");
            string surname = Console.ReadLine();
            var sbSurname = new StringBuilder(surname);

            Console.Write("Введите name: ");
            string name = Console.ReadLine();
            var sbName = new StringBuilder(name);

            for (int i = 0; i < surname.Length; i++)
            {
                if (sbSurname[i] < 65 || sbSurname[i] > 175)
                {
                    check = true;
                }
            }

            for (int i = 0; i < name.Length; i++)
            {
```

```

        if (sbName[i] < 65 || sbName[i] > 175)
        {
            check = true;
        }
    }

    if (check == false)
    {
        if (sbName.Length <= sbSurname.Length)
        {
            size = sbSurname.Length;
            sizeMin = sbName.Length;
            flag = false;

        }
        else
        {
            size = sbName.Length;
            sizeMin = sbSurname.Length;
            flag = true;
        }

        int[] result = new int[size];

        for (int i = 0; i < size; i++)
        {
            if (flag == false)
            {
                result[i] = sbSurname[i];
            }
            else
            {
                result[i] = sbName[i];
            }
        }

        for (int i = 0; i < sizeMin; i++)
        {
            result[i] = (sbSurname[i] | sbName[i]);
            // Console.WriteLine($"result[{i}]= {result[i]}");
        }

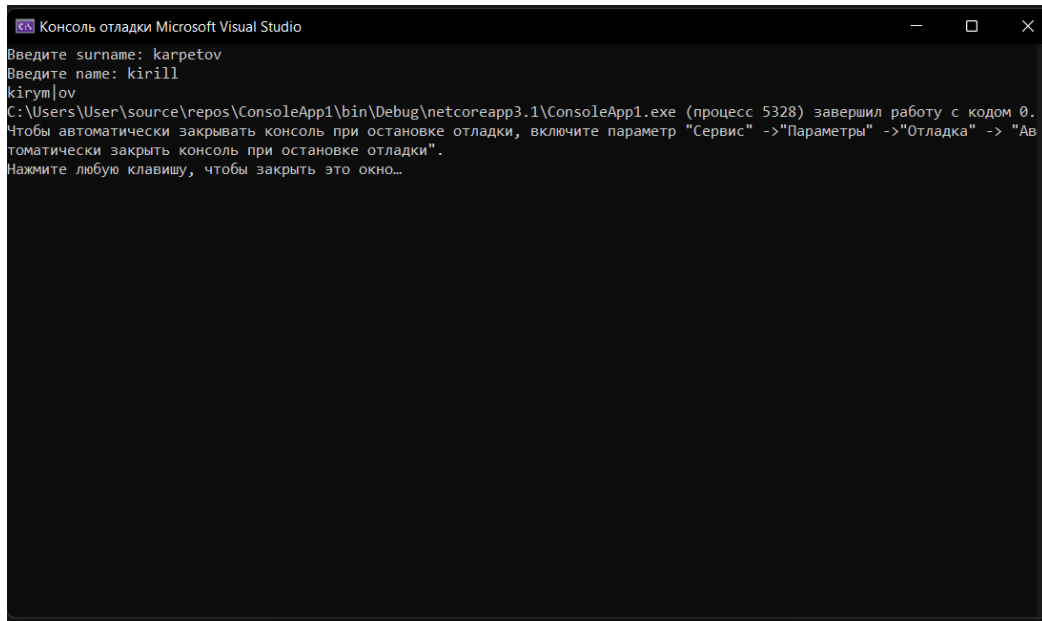
        for (int i = 0; i < size; i++)
            Console.Write((char)result[i]);
    }
    else
    {
        Console.Write("error! try again");
    }
}

}

}

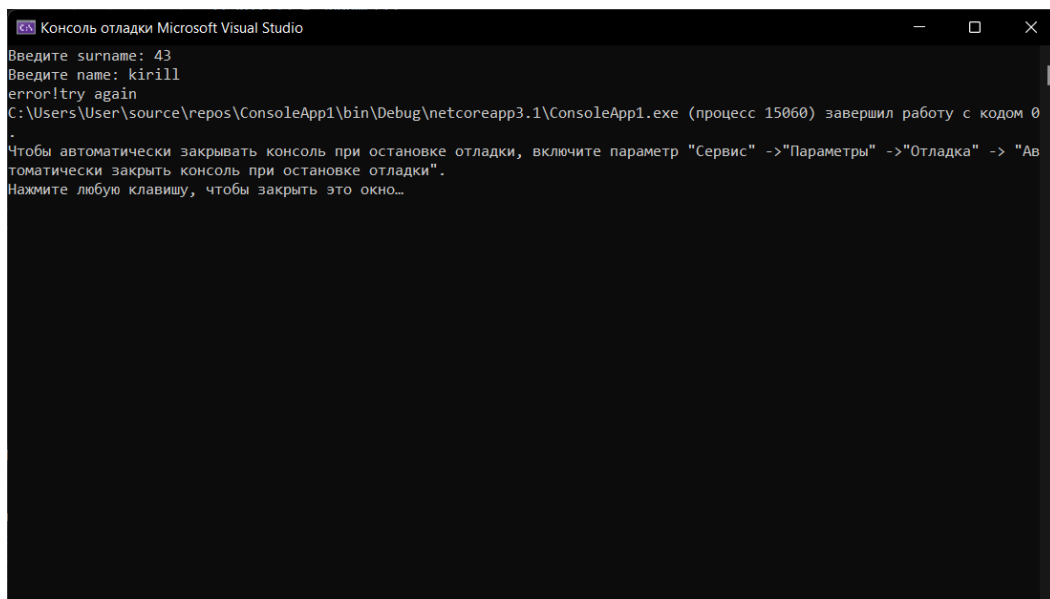
```

Выполнение консольного приложения :



```
Консоль отладки Microsoft Visual Studio
Введите surname: karpetov
Введите name: kirill
kirill\ov
C:\Users\User\source\repos\ConsoleApp1\bin\Debug\netcoreapp3.1\ConsoleApp1.exe (процесс 5328) завершил работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно...
```

Рис. 6



```
Консоль отладки Microsoft Visual Studio
Введите surname: 43
Введите name: kirill
error!try again
C:\Users\User\source\repos\ConsoleApp1\bin\Debug\netcoreapp3.1\ConsoleApp1.exe (процесс 15060) завершил работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно...
```

Рис. 7

Часть 2.2

Задание:

Дано натуральное число:

- найти количество цифр данного числа, больших А (А вводится с клавиатуры);

- определить, верно ли, что данное число принадлежит промежутку от А до В и кратно 3, 4 и 5 (А и В вводятся с клавиатуры).

Найти сумму всех целых чисел из промежутка от А до В, кратных 13 и 5 (А и В вводятся с клавиатуры).

Ход работы: при запуске программы перед пользователем появляется окно, в котором ему необходимо ввести числа (Рис.8). При некорректном вводе данных поле с данными будет очищено, а перед пользователем появится окно с предупреждением (Рис.9). При попытке пользователя нажать на кнопку «ок», не заполнив все поля окна, результат не будет выведен (Рис.10). После корректного заполнения полей программа проверит все необходимы по заданию условия и выведет результат каждого задания в отдельном поле (Рис.11).

Листинг C # программы:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace lab1_2
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void textBox1_TextChanged(object sender, EventArgs e)
        {
            string strNum = (textBox1.Text);
            for (int i = 0; i < strNum.Length; i++)
            {
                if (strNum[i] < 48 || strNum[i] > 57)
                {
                    textBox1.Clear();
                    MessageBox.Show("enter only number");
                }
            }
        }
    }
}
```

```

private void button1_Click(object sender, EventArgs e)
{
}

private void textBox2_TextChanged(object sender, EventArgs e)
{
    string strA1 = (textBox2.Text);
    for (int i = 0; i < strA1.Length; i++)
    {
        if (strA1[i] < 48 || strA1[i] > 57)
        {
            textBox2.Clear();
            MessageBox.Show("enter only number");
        }
    }
}

private void textBox3_TextChanged(object sender, EventArgs e)
{
    string strB1 = (textBox3.Text);
    for (int i = 0; i < strB1.Length; i++)
    {
        if (strB1[i] < 48 || strB1[i] > 57)
        {
            textBox3.Clear();
            MessageBox.Show("enter only number");
        }
    }
}

private void textBox4_TextChanged(object sender, EventArgs e)
{
    string strA2 = (textBox4.Text);
    for (int i = 0; i < strA2.Length; i++)
    {
        if (strA2[i] < 48 || strA2[i] > 57)
        {
            textBox4.Clear();
            MessageBox.Show("enter only number");
        }
    }
}

private void textBox5_TextChanged(object sender, EventArgs e)
{
    string strB2 = (textBox5.Text);
    for (int i = 0; i < strB2.Length; i++)
    {
        if (strB2[i] < 48 || strB2[i] > 57)
        {
            textBox5.Clear();
            MessageBox.Show("enter only number");
        }
    }
}

private void button1_MouseClick(object sender, MouseEventArgs e)
{
    if (!String.IsNullOrEmpty(textBox1.Text) &&
        !String.IsNullOrEmpty(textBox2.Text) && !String.IsNullOrEmpty(textBox3.Text) &&
        !String.IsNullOrEmpty(textBox5.Text) && !String.IsNullOrEmpty(textBox4.Text))
    {
        int tmp = 0;
        int countB = 0;
        int num = int.Parse(textBox1.Text);
    }
}

```

```

int a1 = int.Parse(textBox2.Text);
int b1 = int.Parse(textBox3.Text);
int a2 = int.Parse(textBox4.Text);
int b2 = int.Parse(textBox5.Text);

int count = 0;
int tmpNum = num;

do
{
    if (tmpNum % 10 > a1)
    {
        count++;
    }

}
while ((tmpNum /= 10) != 0);

string result1 = ("Кол-во цифр в числе больших чем {a1} = {count}\r\n");

if (num > a1 && num < b1)
{
    result1 += ("число {num} принадлежит промежутку от А до В\r\n");
    result1 += "\n";
    if (num % 3 == 0)
    {
        result1 += ("число {num} кратно 3\r\n");
    }
    else
    {
        result1 += ("число{num} не кратно 3\r\n");
    }

    if (num % 4 == 0)
    {
        result1 += ("число {num} кратно 4\r\n");
    }
    else
    {
        result1 += ("число {num} не кратно 4\r\n");
    }

    if (num % 5 == 0)
    {
        result1 += ("число {num} кратно 5\r\n");
    }
    else
    {
        result1 += ("число {num} не кратно 5\r\n");
    }

    if ((num % 3 == 0) && (num % 4 == 0) && (num % 5 == 0))
    {
        result1 += ("число {num} кратно 3,4 и 5\r\n");
    }
    else
    {
        result1 += ("число{num} не кратно 3,4 и 5\r\n");
    }
}

```

```

    }
    else
    {
        result1 += ("число {num} не принадлежит промежутку от А до В\r\n");
    }

    string result2 = "";
    if (num > a2 && num < b2)
    {
        result2 += ("число {num} принадлежит промежутку от А до В\r\n");
        for (int j = a2; j < b2; j++)
        {
            tmp = j;
            if ((tmp % 13 == 0) && (tmp % 5 == 0))
            {
                countB += tmp;
            }
        }
        result2 += ("сумма чисел кратных 13 и 5 = {countB}\r\n");
    }
    else
    {
        result2 += ("число {num} не принадлежит промежутку от А до В\r\n");
    }

    textBox6.Text = result1;
    textBox7.Text = result2;
}
else {
    MessageBox.Show("fill in all the lines!");
}
}

}
}

```


Windows forms:

The screenshot shows a Windows Form titled "Form1". It contains the following elements:

- A label "Enter number" above a text box.
- A label "task1" above a text box.
- A label "A" to the left of a text box.
- A label "B" to the left of a text box.
- A label "task 2" above a text box.
- A label "A" to the left of a text box.
- A label "B" to the left of a text box.
- A button labeled "ok" in the center.
- A label "result 1" above a large rectangular box.
- A label "result 2" above another large rectangular box.

Рис. 8

This screenshot shows the same form as Figure 8, but with data entered and a validation dialog box displayed:

- The "Enter number" text box contains the value "456".
- The "task1" text box is empty.
- The text box next to "A" contains the value "1".
- The text box next to "B" is empty.
- The text box next to "task 2" is empty.
- The text box next to "A" is empty.
- The text box next to "B" is empty.
- The "ok" button is still present.
- The "result 1" and "result 2" boxes are still empty.
- A small dialog box is overlaid on the form. It has a title bar with a close button (X). The text inside the dialog box says "enter only number". At the bottom of the dialog box is an "OK" button.

Рис. 9

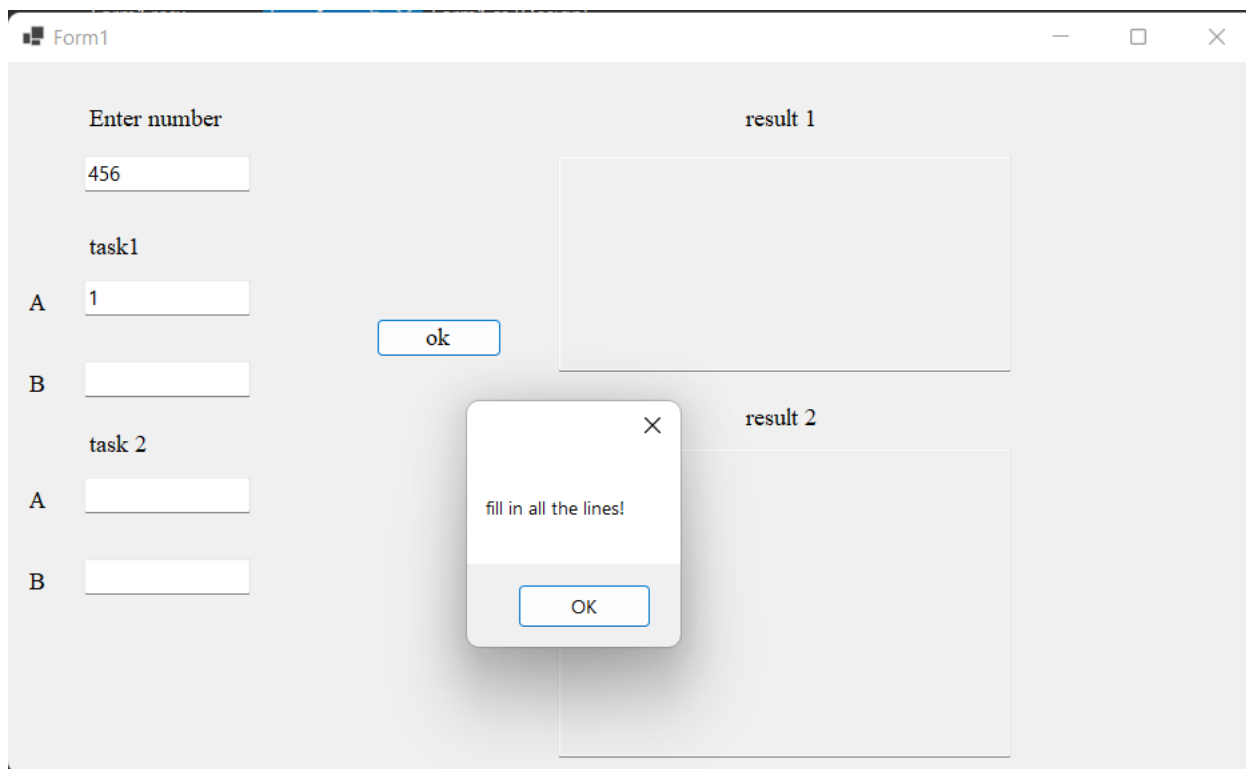


Рис. 10

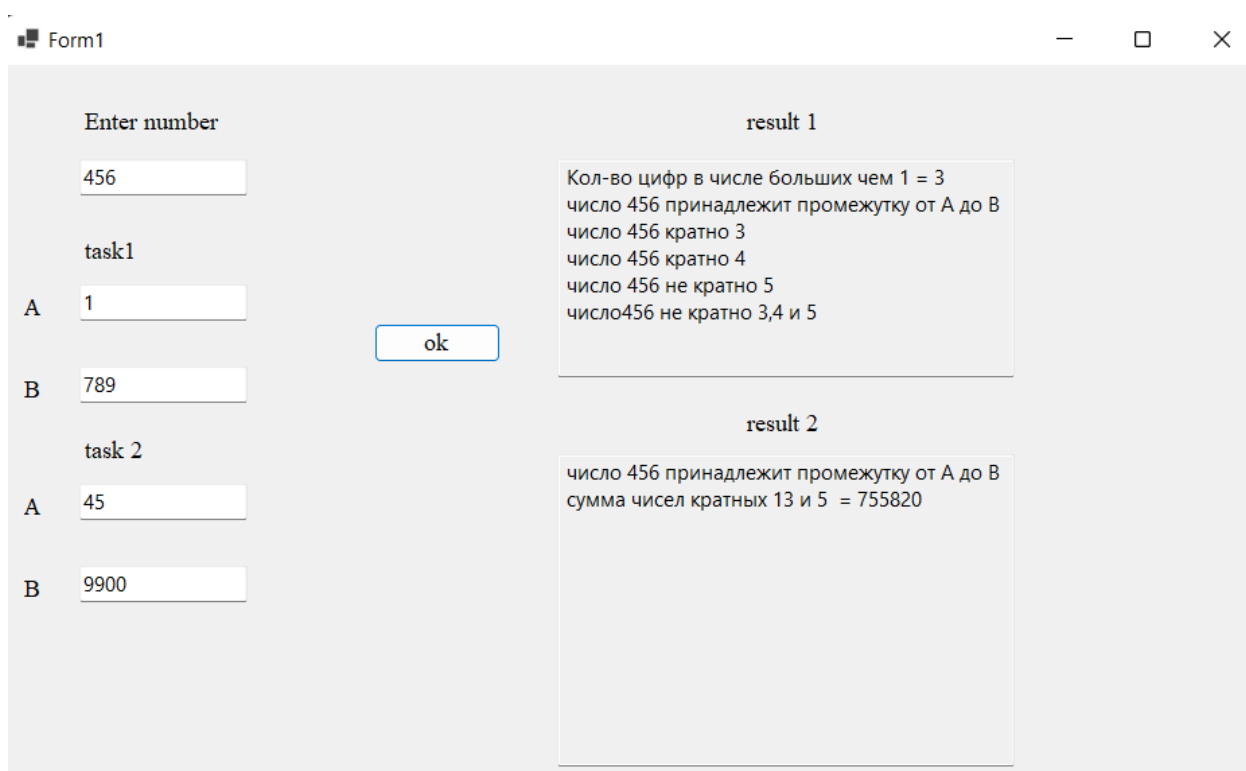


Рис. 11

Вывод:

В ходе работы были получены навыки работы с основными инструментами среды программирования Microsoft Visual Studio 2019, закреплены теоретические знания основных конструкций языка C/C++, а

также были получены навыки в реализации простейших симметричных алгоритмов шифрования/дешифрования информации.