

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

Часть 1. (выполняется на языке C.C++)

Задание к лабораторной работе:

- 1) реализовать алгоритм преобразования информации с использованием логических выражений в соответствии с вариантом задания из ЛАБОРАТОРНОГО ПРАКТИКУМА «Программирование задач по обеспечению информационной безопасности» (варианты, помеченные знаком «*», имеют повышенную сложность);
- 2) выполнить проверку на допустимость значений используемых переменных;
- 3) при описании переменных применить регистровый класс памяти.

Вариант 1.

Задание: Реализовать механизм шифрования и расшифрования символьной информации с применением операции инверсии \sim .

Ход работы:

- 1) Считать строку из консоли в переменную типа string
- 2) Реализовать проверку на ввод пустой строки.
- 3) Создать функции шифрования и расшифрования, с возвращаемым значением типа char.
- 4) С помощью цикла зашифровать каждый символ строки и вывести на консоль.
- 5) С помощью цикла расшифровать каждый символ строки и вывести на консоль

Листинг кода:

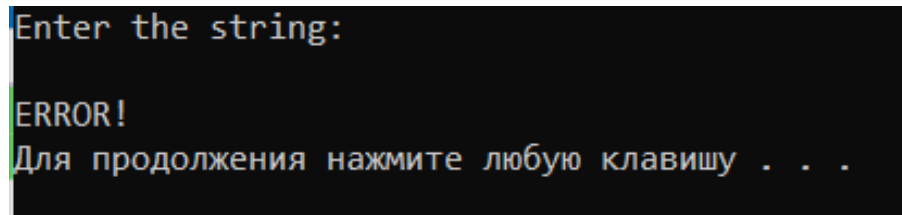
```
#include <iostream>
#include <string>
using namespace std;
//подпрограммы шифрования и расшифрования
char fun_cipher(char simbol) { return ~simbol; }
char fun_decipher(char simbol) { return ~simbol; }
int main()
{
    register string str;
    // вывод строки на консоль
    cout << "Enter the string: " << endl;
```

```

// считывание строки из консоли в переменную str
getline(cin, str);
// проверка на ввод пустой строки
bool flag = 0;
if (str == "") { flag = 1; }
if (flag == 1) {
    cout << "ERROR!" << endl;
}
else {
    // перевод курсора на новую строку (endl) и вывод на //консоль "cipher:"
    cout << endl << "cipher: ";
    // в цикле каждый символ переменной str шифруется
    // и выводится на консоль
    for (register int i = 0; i < str.length(); i++) {
        str[i] = fun_cipher(str[i]);
        cout << str[i];
    }
    cout << endl << "decode cipher: ";
    for (register int i = 0; i < str.length(); i++) {
        str[i] = fun_decipher(str[i]);
        cout << str[i];
    }
    cout << endl;
}
// Чтобы программа не завершалась сразу, а ждала нажатия // какойлибо клавиши
system("pause");
return 0;
}

```

Тесты:

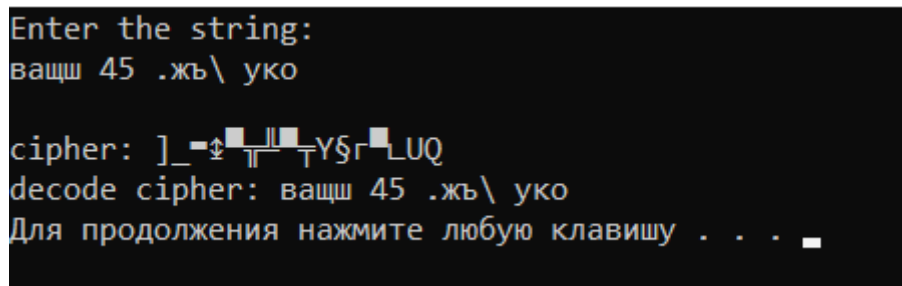


```

Enter the string:
ERROR!
Для продолжения нажмите любую клавишу . . .

```

Рисунок 1 – Тест 1



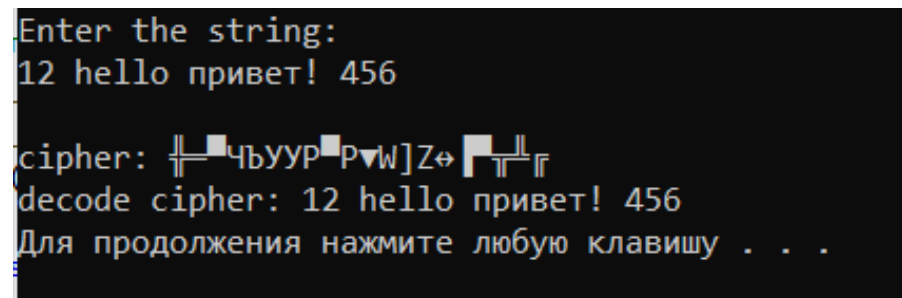
```

Enter the string:
вашш 45 .жь\ уко

cipher: ]_-$T$Y$гLUQ
decode cipher: вашш 45 .жь\ уко
Для продолжения нажмите любую клавишу . . .

```

Рисунок 2 – Тест 2



```

Enter the string:
12 hello привет! 456

cipher: ЧЪУУРРvW]Z$г
decode cipher: 12 hello привет! 456
Для продолжения нажмите любую клавишу . . .

```

Рисунок 3 – Тест 3

Часть 2. (выполняется на языке C#)

Задание:

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

2) Создать программу, выполняющую персональное задание в соответствии с вариантом.

1) Дано натуральное число:

- Найти произведение цифр этого числа;
- Определить, верно ли, что в данном числе нет заданной цифры А (цифра А вводится с клавиатуры).

2) Найти все трехзначные числа, которые при увеличении на 1 делятся на 2, при увеличении на 2 делятся на 3, при увеличении на 3 делятся на 4, а при увеличении на 4 делятся на 5.

Вариант 2.

Ход работы задания 1:

- 1) Считать имя и фамилию из текст-букса в переменную типа string
- 2) Привести переменную к массиву типа char
- 3) Если имя и/или фамилия не введены, вывести сообщение об ошибке
- 4) С помощью цикла проходить по каждому символу, вывести его юникод
- 5) Проверить длиннее имя или фамилия, чтобы верно вывести хвост и сложить символы, вывести юникод и результаты суммы

Листинг кода консольного проекта:

```
using System;

namespace ConsoleApp1
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Enter your name: ");
            string name = Console.ReadLine();
            Console.WriteLine("Enter your surname: ");
            string fullName = Console.ReadLine();

            char[] charName = name.ToCharArray();
            char[] charFullName = fullName.ToCharArray();
        }
    }
}
```

```

int[] unicodeName = new int[name.Length];
int[] unicodeFullname = new int[fullName.Length];

if (fullName.Length == 0 || name.Length == 0)
{
    Console.WriteLine("ERROR!");
}
else
{
    if (fullName.Length >= name.Length)
    {
        int[] text = new int[fullName.Length];
        for (int i = 0; i < name.Length; i++)
        {
            int n = charName[i] + charFullName[i];
            unicodeName[i] = (int)charName[i];
            unicodeFullname[i] = (int)charFullName[i];
            text[i] = n;
        }
        Console.WriteLine("\nTail: ");
        for (int i = name.Length; i < fullName.Length; i++)
        {
            Console.Write(fullName[i]);
            unicodeFullname[i] = (int)charFullName[i];
            text[i] = charFullName[i];
        }

        Console.WriteLine("\nUnicode Surname: ");
        for (int i = 0; i < fullName.Length; i++)
        {
            Console.Write(fullName[i] + " = " + (int)unicodeFullname[i] + "\n");
        }

        Console.WriteLine("\nUnicode name: ");
        for (int i = 0; i < name.Length; i++)
        {
            Console.Write(name[i] + " = " + (int)unicodeName[i] + "\n");
        }

        Console.WriteLine(" \nResult: ");
        for (int i = 0; i < fullName.Length; i++)
        {
            Console.Write((char)text[i]);
        }
    }
    else
    {
        char[] text = new char[name.Length];
        for (int i = 0; i < fullName.Length; i++)
        {
            int n = charName[i] + charFullName[i];
            unicodeName[i] = (int)charName[i];
            unicodeFullname[i] = (int)charFullName[i];
            text[i] = (char)n;
        }
        Console.WriteLine("\nTail: ");
        for (int i = fullName.Length; i < name.Length; i++)
        {
            Console.Write(name[i]);
            unicodeName[i] = (int)charName[i];
            text[i] = charName[i];
        }

        Console.WriteLine("Unicode Surname: ");
    }
}

```



```
Enter your name:
daria
Enter your surname:

ERROR!
```

Рисунок 5 – Тест 2

```
Enter your name:
elizabetha
Enter your surname:
kim

Tail:
zaveta
Unicode Surname:
k = 107
i = 105
m = 109

Unicode name:
e = 101
l = 108
i = 105
z = 122
a = 97
v = 118
e = 101
t = 116
a = 97

Result:
?00
```

Рисунок 6 – Тест 3

Листинг кода Windows Forms проекта:

```
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 WindowsFormsApp1_imya_familiya
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
    }
}
```

```

    }

    private void label1_Click(object sender, EventArgs e)
    {

    }

    private void textBox1_TextChanged(object sender, EventArgs e)
    {

    }

    private void button1_Click(object sender, EventArgs e)
    {
        string name = textBox1.Text;
        string fullname = textBox2.Text;
        char[] char_name = name.ToCharArray();
        char[] char_fullname = fullname.ToCharArray();
        textBox3.Text = "";
        textBox4.Text = "";
        textBox5.Text = "";
        if ((name == null) || (fullname == null) || (name == "") || (fullname == "")) {
textBox4.Text += "ERROR!"; }
        else
        {
            for (int i = 0; i < char_name.Length; i++)
            {
                int b = char_name[i];
                textBox4.Text += char_name[i] + " = " + b + "\r\n ";
                b = 0;
            }
            for (int i = 0; i < char_fullname.Length; i++)
            {
                int b = char_fullname[i];
                textBox4.Text += char_fullname[i] + " = " + b + "\r\n ";
                b = 0;
            }
            if (char_name.Length < char_fullname.Length)
            {
                for (int i = 0; i < char_name.Length; i++)
                {
                    int a = char_name[i] + char_fullname[i];
                    textBox5.Text += char_name[i] + " + " + char_fullname[i] + " = " +
(char)a + "\r\n ";
                    textBox4.Text += char_name[i] + " + " + char_fullname[i] + " = " + a +
"\r\n ";
                    a = 0;
                }
                for (int i = char_name.Length; i < char_fullname.Length; i++)
                {
                    textBox3.Text += char_fullname[i];
                }
                textBox3.Text += "\r\n";
            }
            else
            {
                for (int i = 0; i < char_fullname.Length; i++)
                {
                    int a = char_name[i] + char_fullname[i];
                    textBox5.Text += char_name[i] + " + " + char_fullname[i] + " = " +
(char)a + "\r\n ";
                    textBox4.Text += char_name[i] + " + " + char_fullname[i] + " = " + a +
"\r\n ";
                    a = 0;
                }
            }
        }
    }
}

```

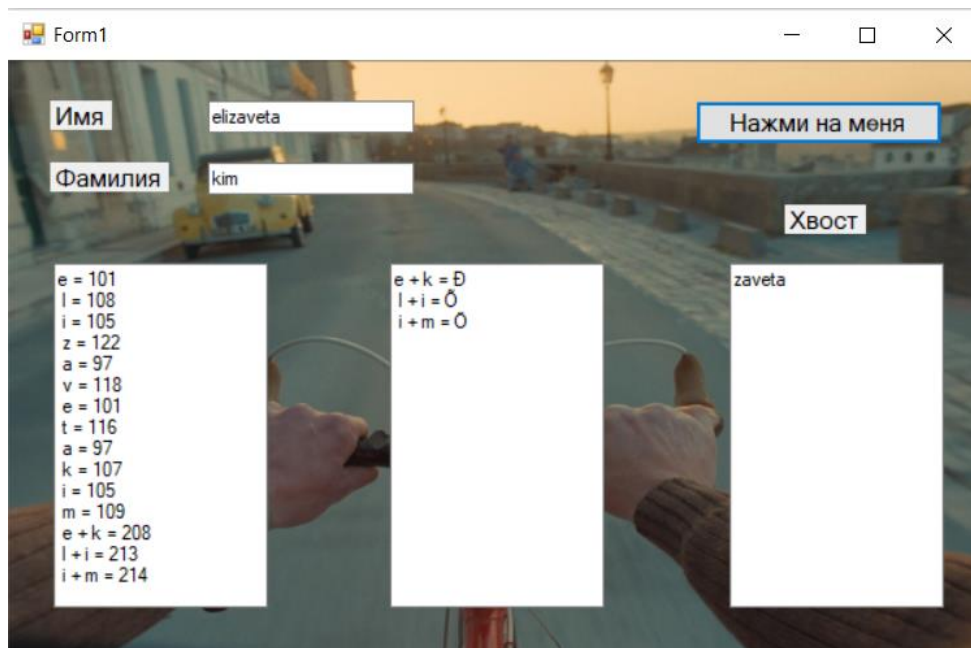



Рисунок 9 – Тест 3

Ход работы задания 2:

- 1) Считываем число, используя деление на 10 с остатком считаем произведение цифр. С помощью метода Contains(), выясняем есть ли нужная цифра в числе.
- 2) Считываем A. В цикле проверяем все трехзначные числа на то, что при увеличении на число оно делится на другое без остатка.

Листинг кода:

```
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 WinFormsApp1
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            string nat = textBox1.Text;
            string a = textBox2.Text;
            textBox3.Text = "";
            //textBox4.Text = "";
        }
    }
}
```

```

bool proverka = true;
if (nat == null || nat == "") { proverka = false; }
for (int i = 0; i < nat.Length; i++)
{
    if ((nat[i] ^ '0') > 9) { proverka = false; }
}
if((a == null) || (a == "") || (a.Length > 1)) { proverka = false; }
if (a.Length == 1 && ( a[0] ^ '0' )> 9 ) { proverka = false; }
if (proverka == false)
{
    textBox3.Text += "ERROR" + "\r\n";
}
else
{
    int nat_1 = Convert.ToInt32(nat);
    int mult = 1;
    while (nat_1 != 0)
    {
        mult *= nat_1 % 10;
        nat_1 /= 10;
    }
    textBox3.Text += "Произведенице цифр = " + mult.ToString() + "\r\n";

    bool flag = nat.Contains(a);
    if (flag == true)
    {
        textBox3.Text += "В числе " + nat + " ЕСТЬ цифра " + a + "\r\n";
    }
    if (flag != true)
    {
        textBox3.Text += "В числе " + nat + " НЕТ цифры " + a + "\r\n";
    }
}
}

private void label1_Click(object sender, EventArgs e)
{
}

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

private void textBox3_TextChanged(object sender, EventArgs e)
{
}

private void textBox2_TextChanged(object sender, EventArgs e)
{
}

private void button2_Click(object sender, EventArgs e)
{
    textBox4.Text = "";
    textBox4.Text += "При увеличении на 1 делятся на 2: ";
    for (int i = 100; i <= 999; i++)
    {
        if ((i + 1) % 2 == 0)
        {
            textBox4.Text += i + " ";
        }
    }
}

```

```

    }
}
textBox4.Text += "\r\n \r\n \r\nПри увеличении на 2 делятся на 3: ";
for (int i = 100; i <= 999; i++)
{
    if ((i + 2) % 3 == 0)
    {
        textBox4.Text += i + " ";
    }
}
textBox4.Text += "\r\n \r\n \r\nПри увеличении на 3 делятся на 4: ";
for (int i = 100; i <= 999; i++)
{
    if ((i + 3) % 4 == 0)
    {
        textBox4.Text += i + " ";
    }
}
textBox4.Text += "\r\n \r\n \r\nПри увеличении на 4 делятся на 5: ";
for (int i = 100; i <= 999; i++)
{
    if ((i + 4) % 5 == 0)
    {
        textBox4.Text += i + " ";
    }
}
}

private void textBox4_TextChanged(object sender, EventArgs e)
{
}

private void Form1_Load(object sender, EventArgs e)
{
}
}
}

```

Тесты:

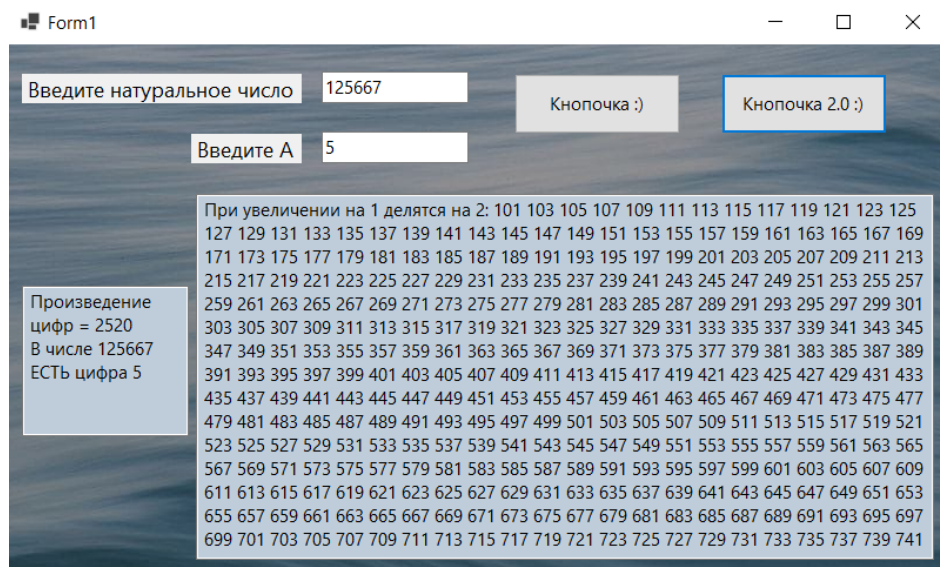


Рисунок 10 – Тест 1

Form1

Введите натуральное число 123ф Кнопочка :) Кнопочка 2.0 :)

Введите A

ERROR

699 701 703 705 707 709 711 713 715 717 719 721 723 725 727 729 731 733 735 737 739 741
 743 745 747 749 751 753 755 757 759 761 763 765 767 769 771 773 775 777 779 781 783 785
 787 789 791 793 795 797 799 801 803 805 807 809 811 813 815 817 819 821 823 825 827 829
 831 833 835 837 839 841 843 845 847 849 851 853 855 857 859 861 863 865 867 869 871 873
 875 877 879 881 883 885 887 889 891 893 895 897 899 901 903 905 907 909 911 913 915 917
 919 921 923 925 927 929 931 933 935 937 939 941 943 945 947 949 951 953 955 957 959 961
 963 965 967 969 971 973 975 977 979 981 983 985 987 989 991 993 995 997 999

При увеличении на 2 делятся на 3: 100 103 106 109 112 115 118 121 124 127 130 133 136
 139 142 145 148 151 154 157 160 163 166 169 172 175 178 181 184 187 190 193 196 199 202
 205 208 211 214 217 220 223 226 229 232 235 238 241 244 247 250 253 256 259 262 265 268
 271 274 277 280 283 286 289 292 295 298 301 304 307 310 313 316 319 322 325 328 331 334
 337 340 343 346 349 352 355 358 361 364 367 370 373 376 379 382 385 388 391 394 397 400
 403 406 409 412 415 418 421 424 427 430 433 436 439 442 445 448 451 454 457 460 463 466

Рисунок 11 – Тест 2

Form1

Введите натуральное число 666 Кнопочка :) Кнопочка 2.0 :)

Введите A 8

Произведение цифр = 216
В числе 666 НЕТ цифры 8

При увеличении на 3 делятся на 4: 101 105 109 113 117 121 125 129 133 137 141 145 149
 153 157 161 165 169 173 177 181 185 189 193 197 201 205 209 213 217 221 225 229 233 237
 241 245 249 253 257 261 265 269 273 277 281 285 289 293 297 301 305 309 313 317 321 325
 329 333 337 341 345 349 353 357 361 365 369 373 377 381 385 389 393 397 401 405 409 413
 417 421 425 429 433 437 441 445 449 453 457 461 465 469 473 477 481 485 489 493 497 501
 505 509 513 517 521 525 529 533 537 541 545 549 553 557 561 565 569 573 577 581 585 589
 593 597 601 605 609 613 617 621 625 629 633 637 641 645 649 653 657 661 665 669 673 677
 681 685 689 693 697 701 705 709 713 717 721 725 729 733 737 741 745 749 753 757 761 765
 769 773 777 781 785 789 793 797 801 805 809 813 817 821 825 829 833 837 841 845 849 853
 857 861 865 869 873 877 881 885 889 893 897 901 905 909 913 917 921 925 929 933 937 941
 945 949 953 957 961 965 969 973 977 981 985 989 993 997

При увеличении на 4 делятся на 5: 101 106 111 116 121 126 131 136 141 146 151 156 161
 166 171 176 181 186 191 196 201 206 211 216 221 226 231 236 241 246 251 256 261 266 271

Рисунок 12 – Тест 3

Form1

Введите натуральное число 666 Кнопочка :) Кнопочка 2.0 :)

Введите A 85

ERROR

681 685 689 693 697 701 705 709 713 717 721 725 729 733 737 741 745 749 753 757 761 765
 769 773 777 781 785 789 793 797 801 805 809 813 817 821 825 829 833 837 841 845 849 853
 857 861 865 869 873 877 881 885 889 893 897 901 905 909 913 917 921 925 929 933 937 941
 945 949 953 957 961 965 969 973 977 981 985 989 993 997

При увеличении на 4 делятся на 5: 101 106 111 116 121 126 131 136 141 146 151 156 161
 166 171 176 181 186 191 196 201 206 211 216 221 226 231 236 241 246 251 256 261 266 271
 276 281 286 291 296 301 306 311 316 321 326 331 336 341 346 351 356 361 366 371 376 381
 386 391 396 401 406 411 416 421 426 431 436 441 446 451 456 461 466 471 476 481 486 491
 496 501 506 511 516 521 526 531 536 541 546 551 556 561 566 571 576 581 586 591 596 601
 606 611 616 621 626 631 636 641 646 651 656 661 666 671 676 681 686 691 696 701 706 711
 716 721 726 731 736 741 746 751 756 761 766 771 776 781 786 791 796 801 806 811 816 821
 826 831 836 841 846 851 856 861 866 871 876 881 886 891 896 901 906 911 916 921 926 931
 936 941 946 951 956 961 966 971 976 981 986 991 996

Рисунок 13 – Тест 4

Вывод: Получены навыки работы с основными инструментами среды программирования Microsoft Visual Studio 2019. Созданы программы с использованием Windows Forms на языке C#.