Министерство образования и науки Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего профессионального образования

«Уральский федеральный университет имени первого Президента России Б.Н. Ельцина»

Уральский энергетический институт

Кафедра автоматизированных электрических систем

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №1**

**«ПЕРВАЯ ПРОГРАММА: КОНСОЛЬНОЕ ПРИЛОЖЕНИЕ»**

Не исправила @i<j@ в проге с отладкой

Не вставила блоксхему в конце

Студентка Ханзафарова А.У.

В списке по журналу №18

Группа ЭН-360001

Преподаватель Семененко С.И.

Оглавление

[Цель работы: 3](#_Toc531345170)

[Задание 1. Запуск среды. 3](#_Toc531345171)

[Задание 2. Создание первого проекта. 3](#_Toc531345172)

[Задание 3. Модификация программы приветствия 3](#_Toc531345173)

[Задание 4. Пошаговое выполнение простого цикла. 4](#_Toc531345174)

[Задание 5. Исследование двойных циклов 6](#_Toc531345175)

[Задание 6. Числовые матрицы. 8](#_Toc531345176)

Блок схема программы [11](#_Toc531345177)

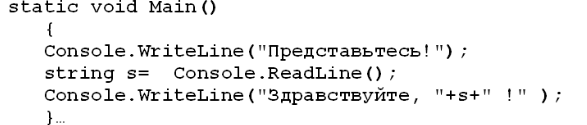
## Цель работы:

Изучение основных инструментов среды, подготовка и запуск простейшей программы, практическое освоение приемов обработки массивов и матриц с использованием возможностей отладочного режима.

Задание 1. Запуск среды.

Задание 2. Создание первого проекта.

Задание 3. Модификация программы приветствия



По индивидуальному заданию Разработайте программу аутентификации, используя в качестве пароля сумму гласных букв в имени и фамилии

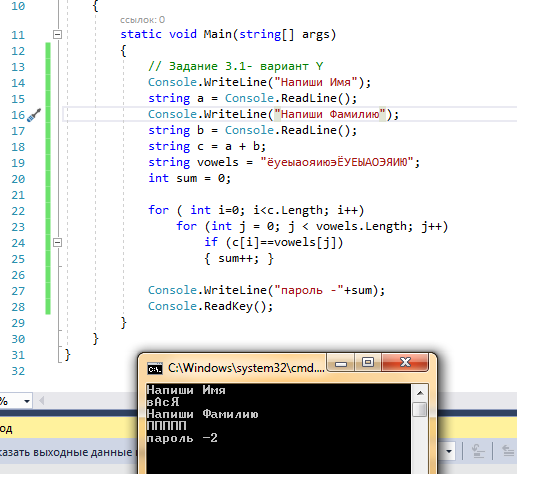
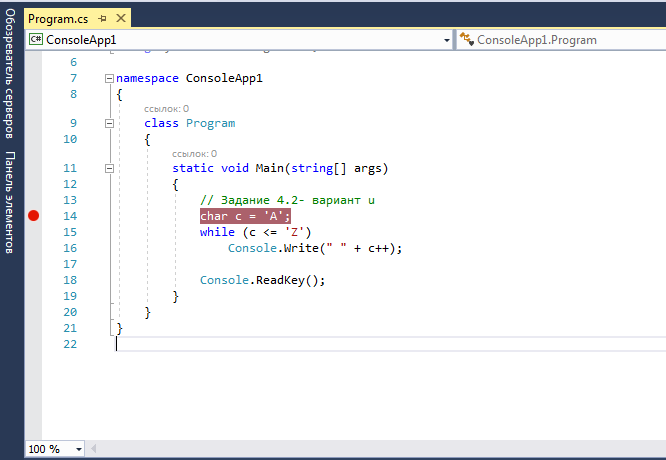


Рисунок 1. Результаты выполнения модифицированной программы приветствия

Задание 4. Пошаговое выполнение простого цикла.



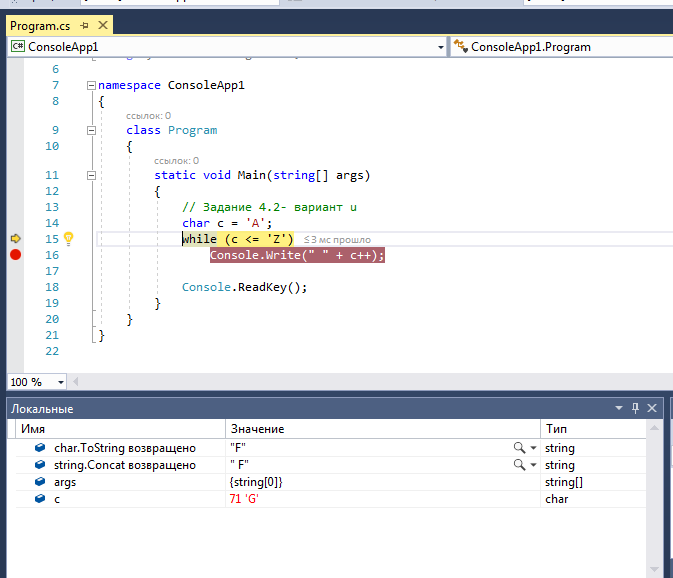


Рисунок 2. Задание точки останова функции в цикле while

Прошагав 6 шагов программы нажатием клавиши F11, получили следующий результат в консольном окне и в окне «Локальные»:

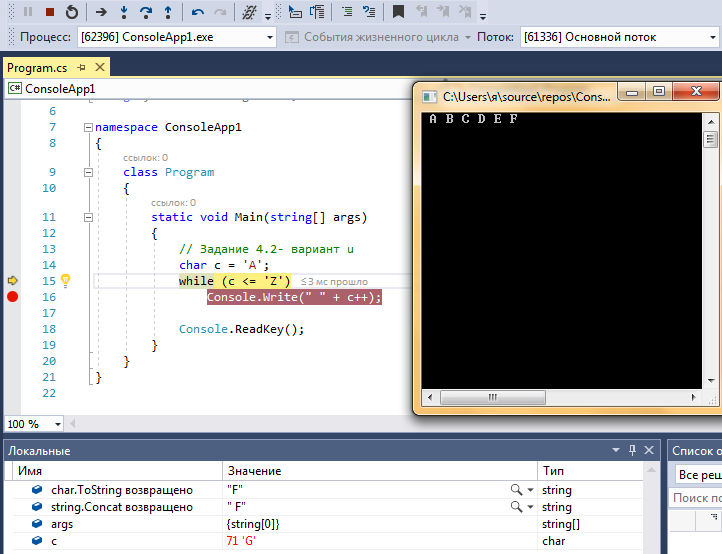


Рисунок 3. Результат пошагового прохождения цикла while

Как видно на рисунке 3, в окне «Локальные» указано значение переменной с «G», которое следующим выведется на экран. Нам нужно, чтобы это была «u». Для этого изменим значение переменной в окне «Локальные» на нужную (см. рисунок 4).

Рисунок 4. Изменение значения переменной

Так как значение 100 для «j» находится за пределами заданного условия (поскольку «Z» имеет значение 99), переменная с не может принимать значение «j». Программа завершает цикл и переходит на следующую строку.

Рисунок 5. Строка программы, выполняемая в данный момент

Чтобы этого не происходило,остановим отладку и изменим условие цикла для получения необходимого результата:

Рисунок 5. Код программы с внесенными изменениями

Прошагав 6 шагов, на 7 шаге изменяем значение «G» на «d» в окне «Локальные», делаем еще один шаг и видим требуемый результат в консольном окне.

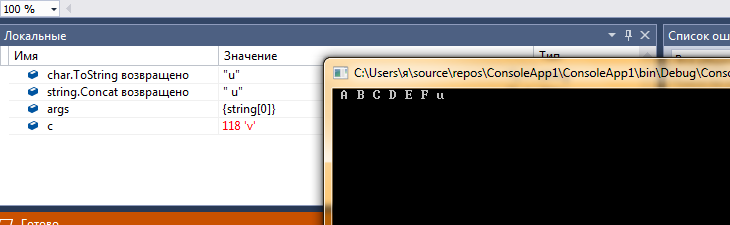


Рисунок 6. Требуемый результат выполнения программы в консольном окне

Стоит отметить, что после изменения значение переменной с в окне «Локальные» следующим значением, выведенным на экран, будет следующее за заданным нами значением «j» (см. рисунок 7)

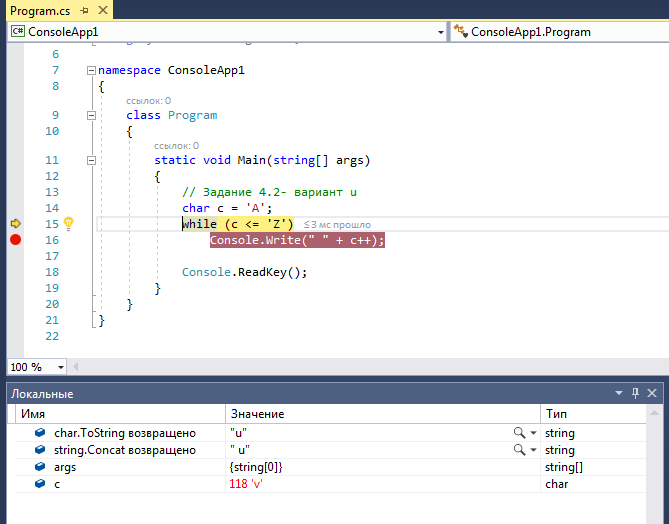


Рисунок 7. Значение переменной с, которое выведется на экран на следующем шаге цикла

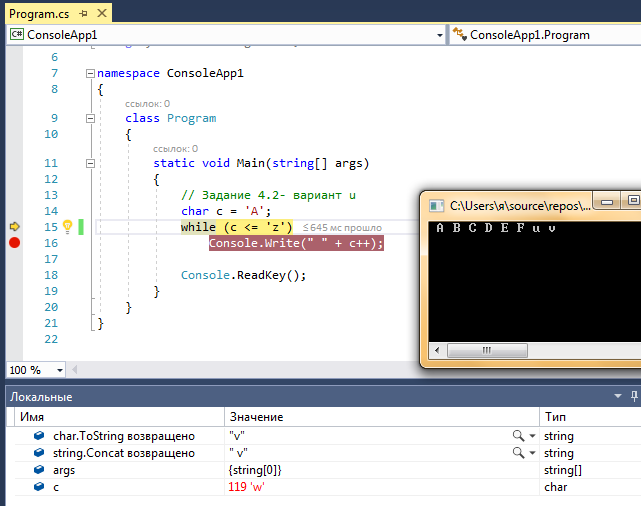


Рисунок 8. Результат выполнения программы на следующем шаге цикла

Итоговый текст программы:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication2

{

class Program

{

static void Main(string[] args)

{

char c = 'A';//Ввод переменной с со значением буквы А

while (c <= 'z')//цикл "пока": пока переменная с при пошаговом выполнении не примет значение Z

Console.Write(" " + c++);// (тело цикла) Последовательный вывод на экран значений переменной с от A до Z

Console.ReadKey();//Введение необходимости нажатия клавиши пользователем для завершения процесса отладки программы

}

}

}

Задание 5. Исследование двойных циклов

При пошаговом прохождении цикла for (для переменной i) c еще одним вложенным циклом for (для переменной j) видим, что изначально переменная i имеет значение 0. Затем на первом шаге первого цикла for (для i) после вывода на экран первого значения i начинается пошаговое выполнение вложенного цикла for (для j), который выводит на экран значения j=0, j=2, j=4, j=6, j=8, прибавляя на каждом шаге 2. Видно, что при следующем шаге значение j равно 10, и вложенный цикл завершается без вывода этого значения в консольное окно, переходя на строку 18. После обработчик переходит вновь в начало цикла for (для i) – строку 13. Соответственно, находясь в строке 18, мы видим первые две строки выполнения программы, и точку останова необходимо поставить именно здесь.

Результат выполнения отладки с точкой останова (помечена красным маркером слева) в строке 18, а также текст программы представлены на рисунке 9.



Рисунок 9. Результат выполнения программы с двойным циклом с точкой останова

Задание 6. Числовые матрицы.

Исходный код программы:

int[,] mas = new int[10, 10];

Random Gen = new Random();

for (int i = 0; i < mas.GetLength(0); i++)

for (int j = 0; j < mas.GetLength(1); j++)

mas[i, j] = Gen.Next(0, 100);

Console.WriteLine("\t\t\t\tЭлементы массива");

int k = 0;

**foreach** (int j in mas)

{

Console.Write("{0,8}", j);

k++;

if (k % 10 == 0) Console.WriteLine();

}

int сумма = 0;

**foreach** (int j in mas) сумма += j;

Console.WriteLine("Сумма элементов Массива = {0}", сумма);

Результатом выполнения программы будет массив случайных чисел (матрица размерность 10х10) от 0 до 100. После желтой строки стоит вложенный цикл, суммирующий все элементы массива. Конечный результат представлен на рисунке 10.

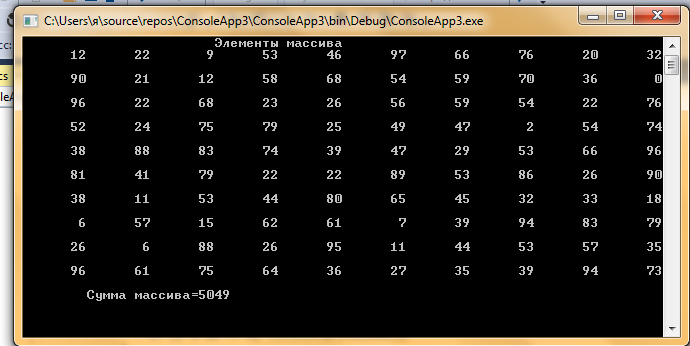


Рисунок 10. Результат выполнения программы с числовыми матрицами с двойными циклами for

Измененный фрагмент кода программы с циклом while:

// 6.1 замена foreach на -while

int[,] mas = new int[10, 10];

Random Gen = new Random();

for (int i = 0; i < mas.GetLength(0); i++)

for (int j = 0; j < mas.GetLength(0); j++)

mas[i, j] = Gen.Next(0, 100);

Console.WriteLine("\t\t\tЭлементы массива");

int k = 0;

int sum = 0;

while (k < 100)

{

Console.Write("{0,8}", mas[k / 10, k % 10]);

sum += mas[k / 10, k % 10];

k++;

if (k % 10 == 0) Console.WriteLine();

}

Console.WriteLine("\tCумма массива={0}", sum);

Результатом выполнения данного фрагмента кода также будет матрица 10х10, содержащая элементы, равные случайным числам от 0 до 100 (см. рисунок 11).

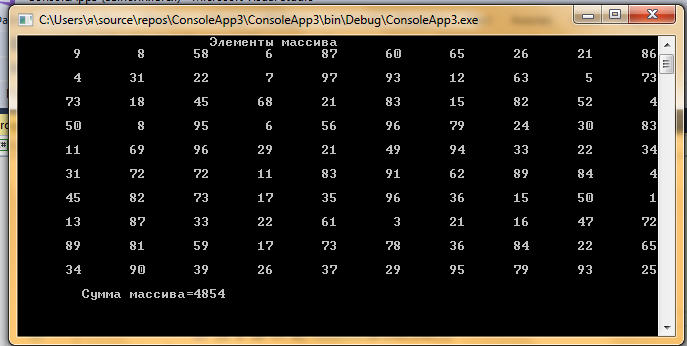


Рисунок 11. Результат выполнения программы с числовыми матрицами с двойными циклами for и while

По индивидуальному заданию изменим конец исходного кода: вместо суммы всех элементов массива напишем фрагмент программы для вывода на экран результата умножения матрицы на саму себя.

Измененный фрагмент кода

Console.WriteLine();

Console.WriteLine("\t\t\tПеремножение матрицы ");

int[,] squer = new int[10, 10];

for (int i = 0; i < 10; i++)

{

for (int j = 0; j < 10; j++)

{

for (k = 0; k < 10; k++)

{

squer[i, j] += mas[i, k] \* mas[k, j];

}

Console.Write("{0,8}", squer[i, j]);

}

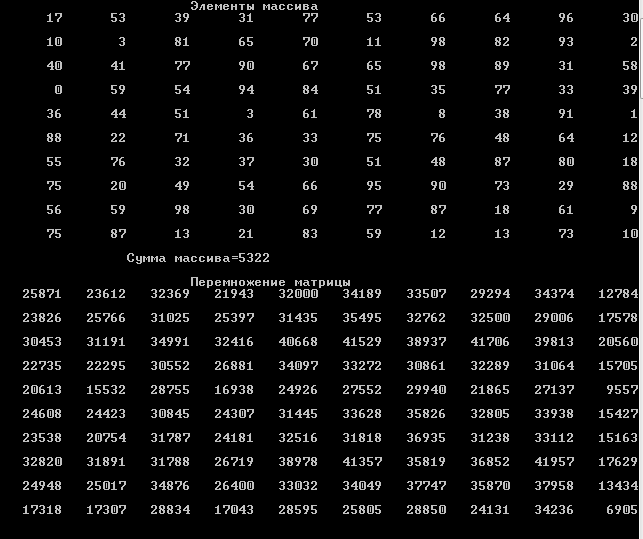
if (k % 10 == 0) Console.WriteLine();

}

Console.ReadKey();

}

}



}

Рисунок 12. Результат выполнения итогового кода программы с числовыми матрицами с двойными циклами for и while и перемножения матрицы на себя

**Блок-схема программы.**

## 

Начало

int[,] mas = new int[10, 10];

Random Gen = new Random();

int l = 0;

l < mas.GetLength(0);

l++;

int j = 0;

j < mas.GetLength(0);

j++;

mas[l, j] = Gen.Next(0, 100)

int j = 0

int l = 0

Console.WriteLine("\t\t\t\tЭлементы массива");

int сумма = 0

double z = 0

Console.Write("{0,8}", mas[i, j]);

int k = 0

k ++

сумма += mas[m, n];

Console.WriteLine()

Да

Нет

if (m <= n)

Конец

Console.WriteLine("Сумма нижней треугольной = {0}", сумма);

Console.WriteLine("Колличество элементов в дивгонали = {0}", k);

Console.WriteLine("Ответ = {0}", z);

z = (double)сумма / k;

int m = 0

int n = 0

int n = 0;

n <mas.GetLength(0);

n++;

int m = 0;

m < mas.GetLength(0);

m++;