МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение

высшего профессионального образования

«Пензенский государственный технологический университет»

(ПензГТУ)

''Кафедра прикладной информатики''

ОТЧЕТ

о выполнении лабораторной работы № 5.

по дисциплине «Информационные технологии»

Выполнил студент

группы 19ИЭ1бзи Иванов Е. А.

Проверила: Голобокова Елена Михайловна

Пенза, 2019

Лабораторная работа №5.

***Разработка циклических программ с заданным числом повторений***

## 1. Цель работы

## Освоение процесса разработки циклических программ с использованием условных циклов (с неизвестным числом повторений) на языке C#.

## 2. Задание на лабораторную работу

Требуется разработать две программы (или одну с двумя циклами) с использованием операторов повтора (циклических операторов) while и do..while.

Примечание: для увеличения сложности программы были использованы не обычные функции. В последнем варианте была применена обфускация кода. Код после обфускации представлен не будет, так как разобрать код не представляется возможным.)

## Текст программы (файл Laboratory5.CS)

## using System;

## using System.Text;

## using System.Threading;

## namespace Laboratory5

## {

## class Program

## {

## public static int menu;

## static void var1(string[] args)

## {

## int n = 0;

## double presicion = 0.01;

## double stepResult = 0.02;

## double result = 0;

## while (Math.Abs(stepResult) > presicion)

## {

## stepResult = Math.Pow(-1, n) / (n + 1);

## result += stepResult;

## Console.WriteLine("Значение выражения на шаге {0} = {1}", n, stepResult);

## Console.WriteLine("Значение суммы ряда на шаге {0} = {1}", n, result);

## n++;

## }

## Console.WriteLine("Сумма ряда={0}", result);

## Console.ReadKey(true);

## }

## static void var8(string[] args)

## {

## Console.Clear();

## Console.Title = "Лабораторная работа 5: вариант 8";

## Console.WriteLine("данный вариант предназначен для вычисления суммы ряда с пределами 1-infinite. Шаг = 0,1");

## Console.WriteLine("(1) цикл while");

## Console.WriteLine("(2) цикл do-while");

## Console.WriteLine("(0) exit");

## try

## {

## menu = int.Parse(Console.ReadLine());

## }

## catch

## {

## Console.WriteLine("что то не так, попробуйте еще раз");

## Console.ReadKey();

## Main(args);

## }

## switch (menu)

## {

## case 1:

## {

## Console.Clear();

## int n = 1;

## int min = 0 - 1;

## double presicion = 0;

## double stepResult = 0.01;

## double result = 0;

## while (Math.Abs(stepResult) > presicion)

## {

## stepResult = Math.Pow(min, n) \* Math.Pow(n, 3) / Math.Pow(3, n);

## result += stepResult;

## Console.WriteLine("Значение выражения на шаге {0} = {1}", n, stepResult);

## Console.WriteLine("Значение суммы ряда на шаге {0} = {1}", n, result);

## n++;

## }

## result = Math.Round(result, 2); //по заданию округлять нужно до одного знака после запятой, но там получится 0, потому вот вам два знака.

## Console.WriteLine("Сумма ряда={0}", result);

## Console.ReadKey();

## Main(args);

## }

## break;

## case 2:

## {

## Console.Clear();

## int n = 1;

## int min = 0 - 1;

## double presicion = 0;

## double stepResult = 0.01;

## double result = 0;

## do

## {

## stepResult = Math.Pow(min, n) \* Math.Pow(n, 3) / Math.Pow(3, n);

## result += stepResult;

## Console.WriteLine("Значение выражения на шаге {0} = {1}", n, stepResult);

## Console.WriteLine("Значение суммы ряда на шаге {0} = {1}", n, result);

## n++;

## } while (Math.Abs(stepResult) > presicion);

## result = Math.Round(result, 2); //по заданию округлять нужно до одного знака после запятой, но там получится 0, потому вот вам два знака.

## Console.WriteLine("Сумма ряда={0}", result);

## Console.ReadKey();

## Main(args);

## }

## break;

## case 0:

## {

## Main(args);

## }

## break;

## }

## }

## static void var16(string[] args)

## {

## Console.Clear();

## Console.Title = "Лабораторная работа 5: вариант 16";

## Console.WriteLine("данный вариант предназначен для вычисления суммы ряда с пределами 1-infinite. Шаг = 0,1");

## int n = 1;

## int min = 0 - 1;

## double presicion = 0;

## double stepResult = 0.01;

## double result = 0;

## int factorial=0,TmpFactorial=1;

## while (Math.Abs(stepResult) > presicion)

## {

## TmpFactorial = TmpFactorial \* (n + 1);

## factorial += TmpFactorial;

## stepResult = Math.Pow(min, n) / (Math.Pow(2, n) \* factorial);

## result += stepResult;

## Console.WriteLine("Значение выражения на шаге {0} = {1}", n, stepResult);

## Console.WriteLine("Значение суммы ряда на шаге {0} = {1}", n, result);

## n++;

## }

## result = Math.Round(result, 2); //по заданию округлять нужно до одного знака после запятой, но там получится 0, потому вот вам два знака.

## Console.WriteLine("Сумма ряда={0}", result);

## Console.ReadKey();

## Main(args);

## }

## static string[] colors = Enum.GetNames(typeof(ConsoleColor));

## static int indexTx = 0;

## static int indexBg = 15;

## static void CallBack(object state)

## {

## Encoding enc = Encoding.GetEncoding(20127);

## Encoding altEnc = Encoding.GetEncoding("ASCII");

## Console.OutputEncoding = Encoding.Unicode;

## Random rndNum = new Random(int.Parse(Guid.NewGuid().ToString().Substring(0, 8), System.Globalization.NumberStyles.HexNumber));

## int rnd1 = rndNum.Next(0, 15);

## indexTx = rnd1;

## int rnd2 = rndNum.Next(0, 15);

## indexBg = rnd2;

## indexBg = rnd2;

## indexTx = (indexTx == colors.Length) ? 0 : indexTx;

## indexBg = (indexBg == colors.Length) ? 0 : indexBg;

## Console.BackgroundColor = (ConsoleColor)Enum.Parse(typeof(ConsoleColor), colors[indexBg]);

## Console.ForegroundColor = (ConsoleColor)Enum.Parse(typeof(ConsoleColor), colors[indexTx]);

## //Очистка экрана

## Console.Clear();

## /\*тут сюрприз\*/

## }

## static void autor(string[] args)

## {

## Console.WriteLine("Работу выполнил студент группы 19ИЭ1БЗИ");

## Console.WriteLine("");

## Console.WriteLine("Иванов Евгений Андреевич");

## Console.WriteLine("https://github.com/NecrosPrime/--sharp");

## Console.WriteLine("https://vk.com/0x000000000f");

## Console.WriteLine("нажмите любую кнопку для открытия пасхалочки...");

## Console.ReadKey(true);

## Timer myTimer = new Timer(new TimerCallback(CallBack), null, 0, 400); //таймер для пасхалочки

## Console.ReadLine();

## }

## static void Main(string[] args)

## {

## Console.Clear();

## Console.ForegroundColor = ConsoleColor.Cyan;

## Console.Title = "Лабораторная работа 5";

## Console.WriteLine("Главное меню:");

## Console.WriteLine("1 Вариант 1");

## Console.WriteLine("8 Вариант 8");

## Console.WriteLine("-1 Об Авторе");

## Console.WriteLine("0 выход");

## try

## {

## menu = int.Parse(Console.ReadLine());

## }

## catch

## {

## Console.WriteLine("что то не так, попробуйте еще раз");

## Console.ReadKey();

## Main(args);

## }

## switch (menu)

## {

## case 1:

## var1(args);

## break;

## case 8:

## var8(args);

## break;

## case -1:

## autor(args);

## break;

## case 0:

## Environment.Exit(0);

## break;

## }

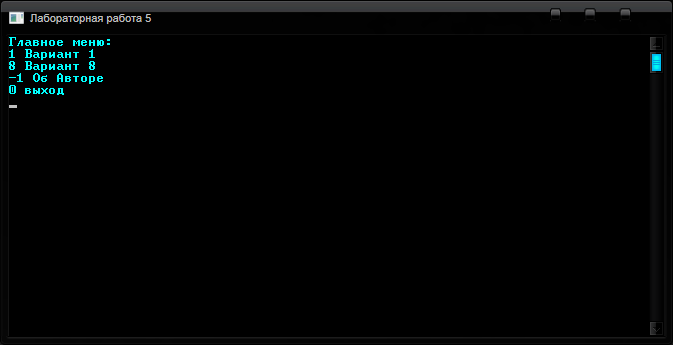
## }

## }

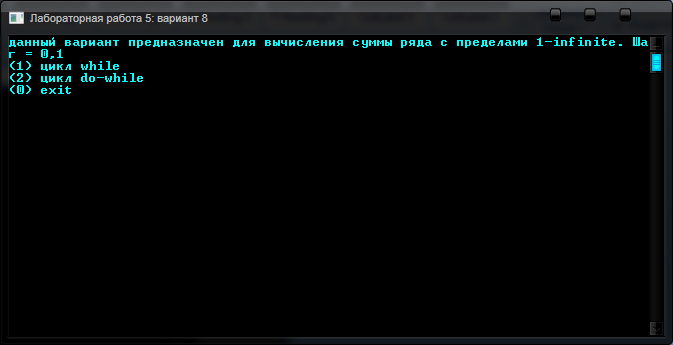
## }

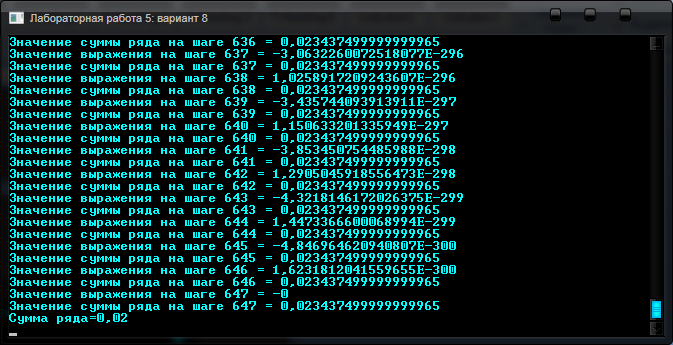
## Примеры работы программы

1. Главное меню. Мне надоел белый цвет, взял cyan

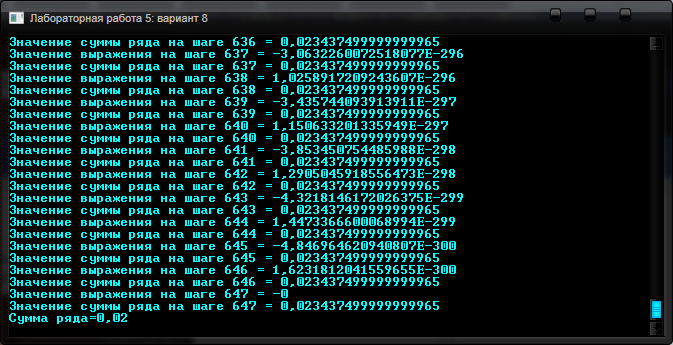


1. Вариант 8



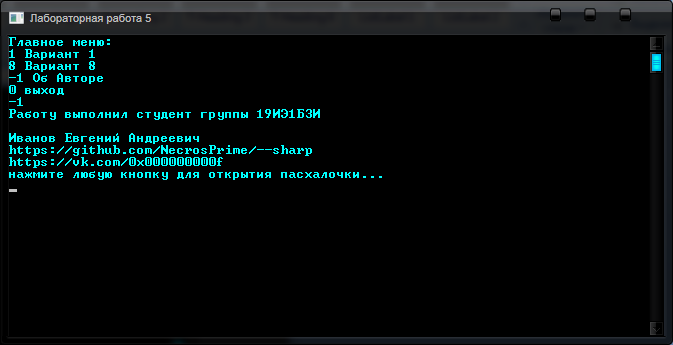


while работает отлично. Было совершено 647 шагов, после чего он выдал результат, который мы округлили согласно заданию.



а это уже do-while. результат тот же.

об авторе



Во всех тестовых случаях программа работает корректно.

**Дополнительные вопросы:**

**1. Каково назначение условных операторов повтора (циклов)? Выполнение кода несколько раз**

**2. Какие требования предъявляются к выражениям, управляющим повторениями? тип - int, границы, и тд**

**3. В чем отличия операторов повтора while и do..while? исполнение хотя бы один раз.**

**4. В каких случаях предпочтительнее использовать для организации циклов оператор повтора for? Что записывается в заголовке этого оператора?**

**тут уже как кому захочется, кому как удобнее. for (int i = 0; i > 0; i++)**

**{**

**body of program;**

**}**

**например**

**5. Какие правила пунктуации необходимо соблюдать при записи операторов? точку с запятой? это единственный символ пунктуации 0\_о**

**6. Что такое вложенные циклы? Какие дополнительные условия необходимо соблюдать при организации вложенных циклов? вложенный цикл - это цикл внутри цикла. при каждом выполнении внешнего цикла, выполняется внутренний некоторое количество раз.**

примечание автора: Данная работа выполнена с учетом на профессионала, а потому задание было мною же усложнено. Сам исходный код добавляю на свой гитхаб, а так же отсылаю вам рабочую программу и исходный код в архиве.