**Лабораторная работа №17**

**Форматированный вывод. Ввод-вывод в файлы**

**Цели:**

1) Научиться выполнять форматированный вывод в окно консоли.

2) Научиться работать с текстовыми файлами.

**Приложение Lab17\_01. Вывод таблицы умножения**

***Задание:***

Добавить в класс **Helper** процедуру **MultiplicationTable(),** выводящую в окно таблицу умножения чисел от 1 до 9. Сопроводить разрабатываемую процедуру соответствующими XML-комментариями.

***Константы:***

Отсутствуют.

***Переменные:***

i, j

***Исходные данные:***

nLub - номер работы

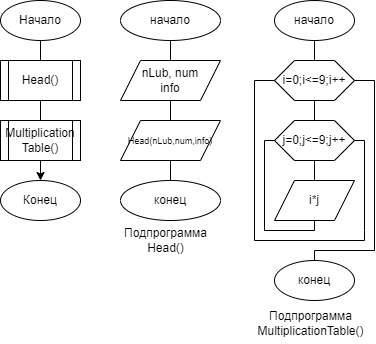
num – номер приложения

info – информация о задании

***Результат:***

В окне консоли выводится таблица умножения.

**Блок-схемы программы:**

****

***Код приложения:***

**Листинг 1.1 – Код подпрограммы MultipicationTable().**

/// <summary>

/// Вывод таблицы умножения до 9

/// </summary>

public static void MultiplicationTable()

{

for (int i = 1; i <= 9; i++)

{

for (int j = 1; j <= 9; j++)

{

Console.Write("{0,3}", i \* j);

}

Console.WriteLine();

}

}

**Листинг 1.2 – Код консольного приложения Lab17\_01.**

namespace GaivoronskayEA.OP.Lab17\_01

{

internal class Program

{

static void Main(string[] args)

{

Helper.Head(17, "01", "Вывод таблицы умножения");

Helper.MultiplicationTable();

Console.ReadLine();

}

}

}

**Тестирование:**

Пример 1)

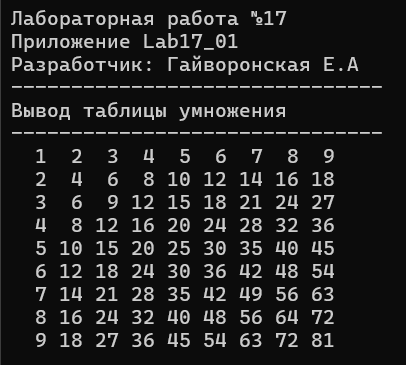


Рисунок 1 – Консольное приложение Lab17\_01

**Приложение Lab17\_02. Табулирование функции.**

***Задание:***

Добавить класс Helper процедуру TableFuncArctg(), выводящую в окно консоли таблицу значений функций f(x) = arctg(x) на отрезке [a; b] с шагом h. Сопроводить разрабатываемую процедуру соответствующими XML-комментариями.

Разработать консольное приложение, которое будет использовать процедуру TableFuncArctg().

***Константы:***

Отсутствуют

***Переменные:***

a, b, h

***Исходные данные:***

nLub - номер работы

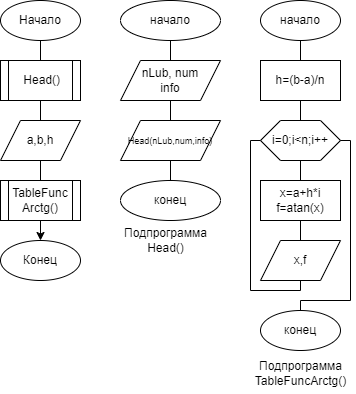
num – номер приложения

info – информация о задании

***Результат:***

В окне консоли выводится таблица значений функции.

**Блок-схема программы:**

****

***Код приложения:***

**Листинг 2.1 – Код подпрограммы TableFuncArctg().**

/// <summary>

/// Расчет функции

/// </summary>

/// <param name="a">Левая граница отрезка</param>

/// <param name="b">Правая граница отрезка</param>

/// <param name="h">шаг</param>

public static void TableFuncArctg(double a, double b, double h)

{

Console.WriteLine(" x | f(x)");

Console.WriteLine("-------------------");

double n = (b - a) / h;

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

{

double x = a + h \* i;

double f = Math.Atan(x);

Console.WriteLine("{0,8:F3} {1,8:F3}", x, f);

}

}

**Листинг 2.2 – Код консольного приложения Lab17\_02.**

namespace GaivoronskayEA.OP.Lab17\_02

{

internal class Program

{

static void Main(string[] args)

{

Helper.Head(17, "02", "Вывод таблицы значений функции");

Console.WriteLine("Введите левую границу:");

double a = double.Parse(Console.ReadLine());

Console.WriteLine("Введите правую границу:");

double b = double.Parse(Console.ReadLine());

Console.WriteLine("Введите шаг:");

double h = double.Parse(Console.ReadLine());

Helper.TableFuncArctg(a, b, h);

Console.ReadLine();

}

}

}

**Тестирование:**

Пример 1)

**Исходные данные:**

a = -0,8

b = 1,8

h = 0,8

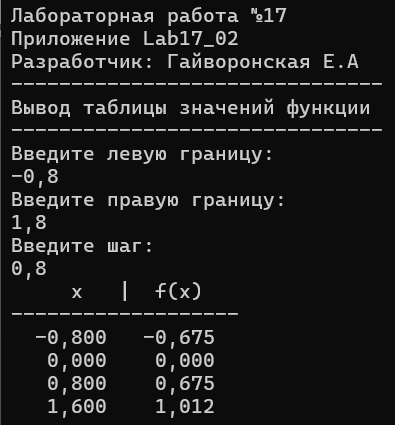


Рисунок 2 – Консольное приложение Lab17\_02

**Приложение Lab17\_03. Разработка и использование метода записи одномерного массива в текстовый файл.**

***Задание:***

Добавить в класс Helper метод WriteArrayFile(), выводящий одномерный массив в указанный текстовый файл. Метод должен принимать в качестве параметров путь к файлу и одномерный массив вещественных чисел. Сопроводить разрабатываемый метод XML-комментариями.

Разработать консольное приложение, которое, используя метод WriteArrayFile(), записывало бы в файл «H:\массив.txt», введенный в консоль, одномерный массив.

***Константы:***

path = C:\\Users\\osnov\\Desktop\\массив.txt

***Переменные:***

array, f, n

***Исходные данные:***

nLub - номер работы

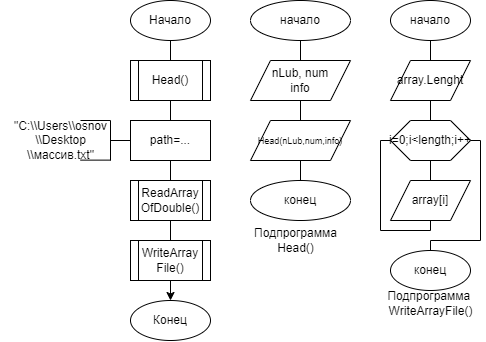
num – номер приложения

info – информация о задании

***Результат:***

В текстовом файле выводится одномерный массив введенный с клавиатуры в консоли.

**Блок-схема программы:**

******

***Код приложения:***

**Листинг 3.1 – Код подпрограммы WriteArrafyFile().**

/// <summary>

/// Выводит одномерный массив

/// в текстовый файл

/// </summary>

/// <param name="path">Путь к файлу</param>

/// <param name="array">Одномерный массив, который будет выведен</param>

public static void WriteArrayFile(string path, double [] array)

{

StreamWriter f = new StreamWriter(path, false);

f.WriteLine(array.Length);

for (int i = 0; i <array.Length; i++)

{

f.WriteLine(array[i]);

}

f.Close();

}

**Листинг 3.2 – Код консольного приложения Lab17\_03.**

namespace GaivoronskayEA.OP.Lab17\_03

{

internal class Program

{

static void Main(string[] args)

{

Helper.Head(17, "03", "Разработка и использование метода записи массива в файл");

string path = "C:\\Users\\osnov\\Desktop\\массив.txt";

double[] array = Helper.ReadArrayOfDouble();

Helper.WriteArrayFile(path, array);

Console.ReadLine();

}

}

}

**Тестирование:**

Пример 1)

**Исходные данные:**

n = 4

array = [18, 8, 19, 9]

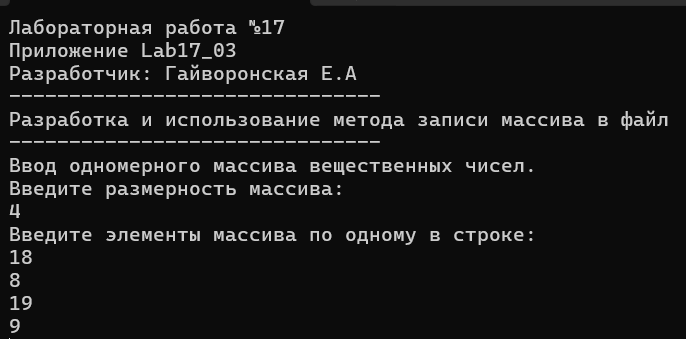


Рисунок 3 – Консольное приложение Lab17\_03

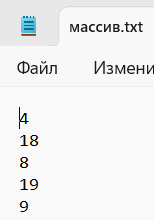


Рисунок 4 – вывод массива в текстовый файл.

**Приложение Lab17\_04. Разработка и использование метода чтения одномерного массива из текстового файла.**

***Задание:***

Реализовать в классе Helper функцию ReadArrayFile(), читающую одномерный массив из указанного текстового файла. Функция должна принимать путь к файлу в качестве параметра и возвращать в точку вызова считанный одномерный массив. Сопроводить метод-функцию XML-комментариями.

Разработать консольное приложение, которое, используя разработанный метод ReadArrayFile(), читало бы из файла «H:\массив.txt» одномерный массив и выводило его на консоль.

***Константы:***

Path = C:\\Users\\osnov\\Desktop\\массив.txt

***Переменные:***

array, f, n

***Исходные данные:***

nLub - номер работы

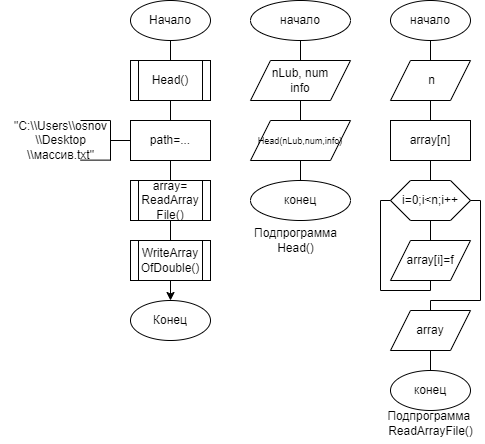
num – номер приложения

info – информация о задании

***Результат:***

В окне консоли выводятся элементы массива, считанные с текстового файла.

**Блок-схема программы:**

******

***Код приложения:***

**Листинг 4.1 – Код подпрограммы ReadArrayFile().**

/// <summary>

/// Считывает одномерный массив вещественных чисел из текстового файла.

/// </summary>

/// <param name="path">Путь к файлу</param>

/// <returns>Возвращает считанный массив</returns>

public static double[] ReadArrayFile(string path)

{

StreamReader f = new StreamReader(path);

int n = int.Parse(f.ReadLine());

double[] array = new double[n];

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

{

array[i] = double.Parse(f.ReadLine());

}

f.Close();

return array;

}

**Листинг 4.2 – Код консольного приложения Lab17\_04.**

namespace GaivoronskayEA.OP.Lab17\_04

{

internal class Program

{

static void Main(string[] args)

{

Helper.Head(17, "04", "Чтение одномерного массива из текстового файла");

string path = "C:\\Users\\osnov\\Desktop\\массив.txt";

double[] array = Helper.ReadArrayFile(path);

Helper.WriteArray(array);

Console.ReadLine();

}

}

}

**Тестирование:**

Пример 1)

**Исходные данные:**

n = 4

array = [18, 8, 19, 9]

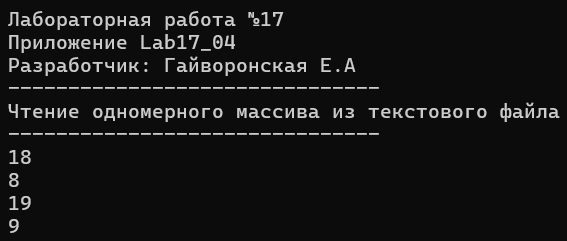


Рисунок 5 – Консольное приложение Lab17\_04