**Практическая работа No1**

Программирование линейных алгоритмов (консольное приложение)

**Цель работы:**

 Изучение структуры программы на языке C#

 Изучение ввода/вывода и форматирования данных при вводе и выводе

В результате выполнения лабораторной работы студент должен

знать:

 Структуру программы на С#

 способы ввода и вывода данных в консольном приложении

 способы форматирования данных разного типа при выводе на консоль

уметь:

 Составлять арифметические выражения

 Использовать стандартные операторы

 Программировать линейные алгоритмы

Выполнил: Жиров Егор 3ИП-1-21

Код:

using System;

namespace Osas

{

class Program

{

static void Main(string[] args)

{

Console.WriteLine("Введите значение X:");

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

double result = Math.Log(Math.Abs(Math.Cos(x))) / Math.Log(1 + Math.Pow(x, 2));

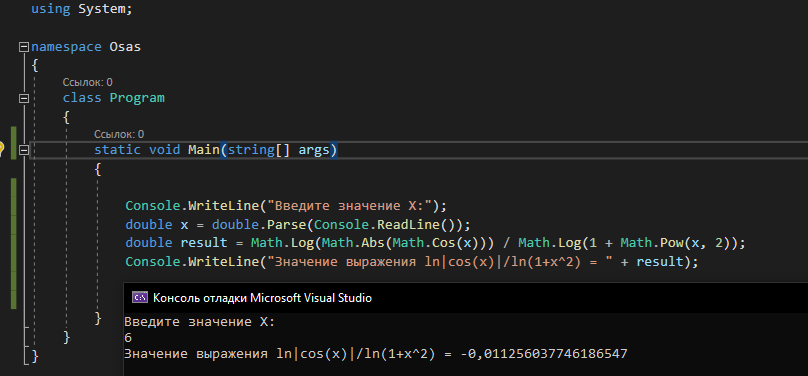
Console.WriteLine("Значение выражения ln|cos(x)|/ln(1+x^2) = " + result);

}

}

}

Скриншот:



Контрольные вопросы:

1. Использование типа данных:

Указание типа данных величины в программировании позволяет определить, какие операции можно выполнять с этой величиной, а также выделить необходимый объем памяти для её хранения.

1. Примеры констант-литералов числовых типов:

Целочисленные: 42, -10, 0x1A (шестнадцатеричная форма).

С плавающей точкой: 3.14, -0.5e-2.

1. Скалярные типы данных:

Скалярные типы данных представляют отдельные значения и включают в себя целочисленные, вещественные, символьные и логические типы.

1. Примеры служебных слов C#:

using: Используется для добавления пространства имен.

class: Определяет класс.

public, private: Модификаторы доступа.

static: Определяет статический элемент.

void: Указывает, что метод не возвращает значение.

1. Выражение, операция, операнд, оператор:

Выражение: Комбинация значений и операторов, которая может быть вычислена.

Операция: Действие, выполняемое над операндами.

Операнд: Значение, с которым выполняется операция.

Операторы: Символы или ключевые слова, обозначающие операции.

Пример операций: + (сложение), - (вычитание), \* (умножение), / (деление), % (остаток от деления), == (равно) и другие.

1. Переменная:

Переменная - это именованное хранилище для данных, которое может изменять своё значение в течение выполнения программы.

1. Директива using:

using в C# применяется для добавления пространств имен. Это позволяет использовать типы из этих пространств имен без полного указания их пути.

1. Solution vs. Project в Visual Studio:

Solution: Объединяет один или несколько проектов. Это контейнер для проектов и других элементов.

Project: Представляет собой отдельное приложение, библиотеку или модуль внутри решения.

1. Инициализация переменной:

Инициализация переменной - это присвоение начального значения при её объявлении. Например, int x = 10; инициализирует переменную x значением 10. System.Console.WriteLine:

System: Пространство имен, где определен класс Console.

Console: Класс для работы с консолью.

WriteLine: Метод, который выводит текст в консоль и переводит строку.