Лабораторная работа № 1

студента группы ИТз-221

Дмитриева Дмитрия Анатольевича

*Выполнение: \_\_\_\_\_\_\_\_\_ Защита: \_\_\_\_\_\_\_\_\_\_\_*

Интегрированная среда разработки Visual Studio. Основы языка программирования C#

*Цель работы***:** знакомство с платформой .NET Framework и справочной службой MSDN. Получение сведений о структуре проекта C# в Visual Studio и навыков в разработке консольных и Windows Forms приложений с помощью C# и Visual Studio. Знакомство с типами данных языка C#. Особенности консольного ввода/вывода в приложениях C#.

**Ход работы:**

***Вариант - 12***

1. Ознакомился с примерами, встроенными типами данных в C#, выполнил пример, порядок выполнения которого приводится в разделе MSDN «Практическое руководство. Создание консольного приложения C#» (рис. 1)



Рисунок 1 – реализация примера в unix системе

1. Разработал алгоритм решения задачи в соответствии с вариантом задания, написал программу на языке C# и создал консольное приложение в среде разработки C#. (рис. 2).

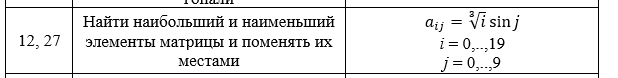
****

Рисунок 2 – индивидуальное задание

1. Решение задачи разделил на подзадачи: формирование исходного массива   
   (рис. 3), вывод на экран элементов исходного массива (рис. 4), выполнение задания и вывод результатов на экран. (рис. 5). Решение каждой подзадачи оформил в виде отдельного метода. Предусмотрел передачу значений между методами через параметры.

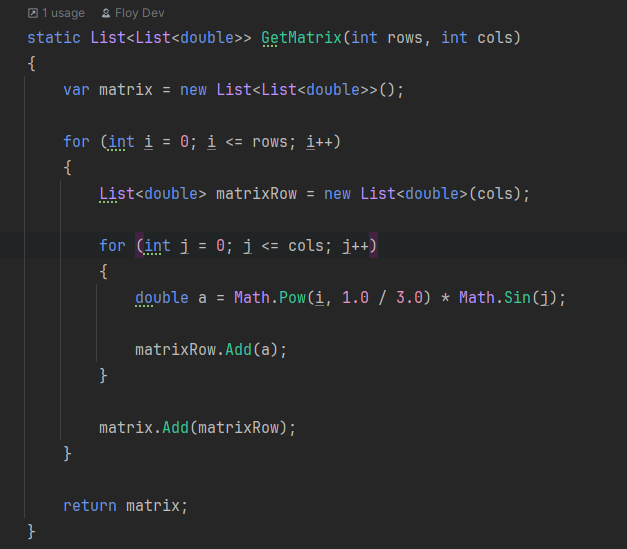


Рисунок 3 – формирование матрицы

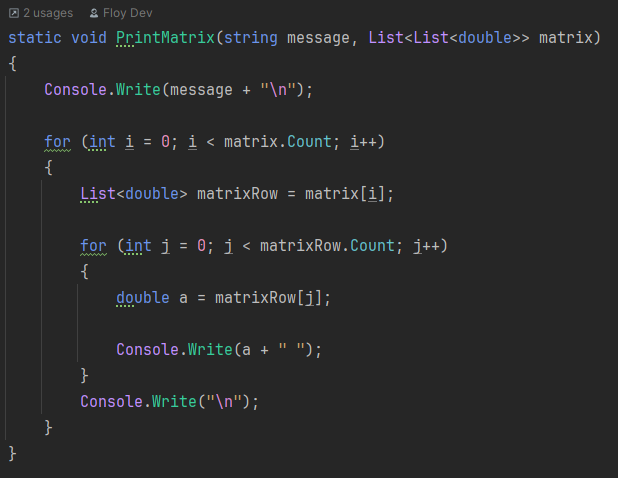


Рисунок 4 – вывод матрицы

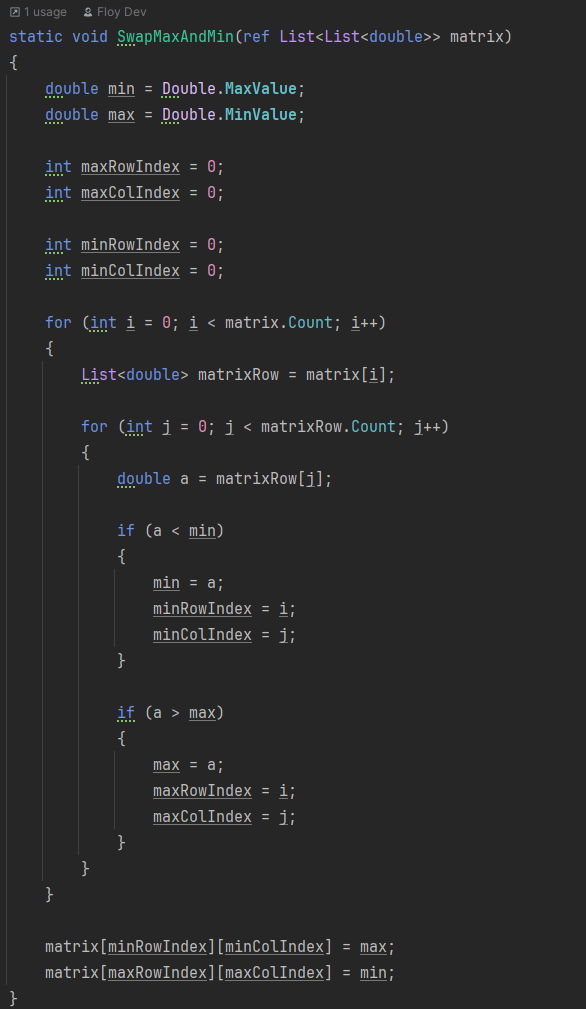


Рисунок 5 – замена минимального и максимального элемента

1. Добавил методы приложения в главный метод Main, запустил и протестировал программу. Элементы 2,6400051764664845 и -2,558795115510614 были заменены местами (рис. 6).

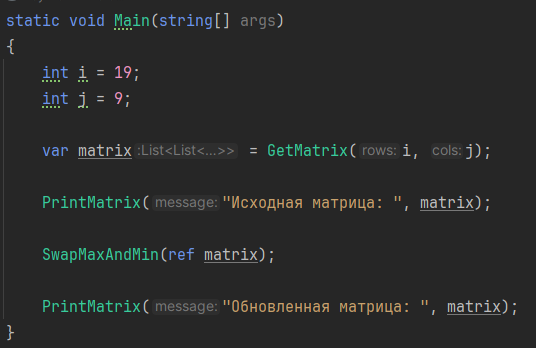


Рисунок 6 – метод Main

**Контрольные вопросы**:

1. Основные компоненты .NET Framework: CLR управляет выполнением программ, а библиотека классов предоставляет функциональность для работы с данными, сетью и пользовательским интерфейсом.
2. Этапы создания приложения в .NET Framework: включают разработку, компиляцию исходного кода, создание сборки (DLL/EXE), тестирование и развертывание на целевой системе.
3. Архитектура .NET Framework: состоит из CLR, библиотеки классов, языков программирования, а также механизмов управления памятью и сборки мусора.
4. Элементы окна Visual Studio C#: включают редактор кода, панель инструментов, окно ошибок, обозреватель решений и свойства.
5. Пространство имен .NET Framework: организует классы в иерархию для удобства доступа; для ускоренного доступа используются директивы using и полные пути к классам.
6. Структура проекта консольного приложения: обычно включает файл программы (Program.cs), файлы с кодом и конфигурационные файлы, такие как App.config.
7. Структура проекта Windows Forms приложения: содержит форму (например, Form1.cs), элементы управления (кнопки, текстовые поля) и конфигурационные файлы.
8. Структура программы на C#: включает пространство имен, классы, методы, а также конструкцию Main как точку входа в приложение.
9. Система общих типов: обеспечивает единый набор типов, доступных для всех языков .NET, таких как int, string, DateTime.
10. Наиболее востребованные встроенные типы: включают int, string, bool, DateTime, double, и object.
11. Консольный ввод/вывод: для ввода используется Console.ReadLine(), а для вывода — Console.WriteLine().

**Вывод**: ознакомился с платформой .NET Framework и справочной службой MSDN. Получил сведений о структуре проекта C# в Visual Studio, Rider и навыков в разработке консольных с помощью C# и Visual Studio, Rider. Ознакомился с типами данных языка C#. Особенности консольного ввода/вывода в приложениях C#.