**Лабораторная работа №1.  
Массивы**

**Цель работы**

Приобретение навыков работы с массивами, циклами, функциями.

**Теоретические сведения**

Троелсен Э. Язык программирования C# 5.0 и платформа .NET 4.5 (Вильямс, 6-е изд., 2013). Глава 4

**Задание**

1. Написать псевдокод программы, реализующий алгоритм решения задачи.
2. Создать консольное приложение на языке C# в соответствии со следующими требованиями:

* размеры прямоугольной матрицы задаются через параметры командной строки (кол-во и тип переданных параметров должны проверяться программой);
* в методе main должен быть реализован вызов созданных функций;
* матрица заполняется положительными и отрицательными случайными числами с помощью метода, возвращающего массив (целые числа – не более 2 разрядов, вещественные – не более 2 разрядов до десятичной точки и 1 разряда после десятичной точки);
* вывод на консоль массивов реализовать с помощью одноименных методов, в качестве параметра которым передается либо одномерный, либо двумерный массив;
* решение задачи реализовать с помощью метода, имеющего тип void. Данные возращать через параметр с модификатором ref или out.

Варианты заданий:

|  |  |  |
| --- | --- | --- |
| **№ вар.** | **Тип элементов матрицы** | **Задача** |
| 1 | целый | Сформировать одномерный массив как максимальные элементы строк матрицы. |
| 2 | вещественный | Сформировать одномерный массив как сумму отрицательных элементов столбцов матрицы. |
| 3 | целый | Сформировать одномерный массив как среднее арифметическое элементов столбцов матрицы. |
| 4 | вещественный | Сформировать одномерный массив как суммы положительных элементов строк матрицы. |
| 5 | вещественный | Сформировать одномерный массив как произведение элементов строк матрицы. |
| 6 | целый | Вычислить среднее арифметическое элементов побочной диагонали матрицы. |
| 7 | целый | Вычислить сумму максимального и минимального элементов главной диагонали матрицы. |
| 8 | вещественный | Вычислить произведение элементов главной диагонали матрицы. |
| 9 | целый | Сформировать одномерный массив как столбец матрицы, содержащий максимальный элемент на главной диагонали. |
| 10 | целый | Сформировать одномерный массив как минимальные элементы столбцов матрицы. |
| 11 | вещественный | Вычислить сумму элементов матрицы, расположенных над главной диагональю. |
| 12 | вещественный | Вычислить произведение положительных элементов, расположенных под главной диагональю матрицы. |
| 13 | целый | Вычислить сумму максимального и минимального элементов матрицы. |
| 14 | целый | Сформировать одномерный массив как столбец матрицы, содержащий минимальный элемент на побочной диагонали. |
| 15 | вещественный | Поменять местами элементы главной и побочной диагоналей матрицы. |
| 16 | целый | Получить вектор как строку матрицы с максимальной суммой элементов. |
| 17 | целый | Получить вектор как произведение нечетных элементов столбцов матрицы. |
| 18 | целый | Получить вектор как столбец матрицы с минимальной суммой элементов. |
| 19 | вещественный | Вычислить произведение максимальных элементов главной и побочной диагоналей матрицы. |
| 20 | вещественный | Поменять местами максимальный и минимальный элементы матрицы. |
| 21 | вещественный | Вычислить сумму модулей отрицательных элементов, расположенных в секторе над главной диагональю матрицы. |
| 22 | вещественный | Получить вектор как строку матрицы, в которой находится максимальный элемент матрицы. |
| 23 | вещественный | Вычислить сумму индексов максимального элемента матрицы. |
| 24 | целый | Получить вектор как столбец матрицы, содержащий максимальный элемент. |
| 25 | вещественный | Поменять местами максимальный и минимальный элементы главной диагонали матрицы. |
| 26 | целый | Вычислить произведение отрицательных элементов, расположенных над побочной диагональю матрицы. |
| 27 | вещественный | Вычислить произведение максимального и минимального элементов побочной диагонали матрицы. |
| 28 | целый | Получить вектор как строку матрицы, в которой произведение максимального и минимального элементов максимально. |
| 29 | целый | Поменять местами минимальные элементы главной и побочной диагоналей матрицы. |
| 30 | целый | Поменять местами строки с максимальным и минимальным элементами матрицы. |
| 31 | целый | Получить вектор как произведение четных элементов побочной диагонали матрицы. |

## Дополнительные вопросы:

* В каких случаях удобнее использовать out, в каких ref?

## Основные ошибки в лабораторных:

* Невнимательное чтение задания лабораторной.
* Использование символов однострочных комментариев «//» там, где содержимое явно требует гораздо больше одной строки.
* В комментариях используются понятия не из предметной области, а из языка программирования (слова «переменная», «метод», «инициализация массива» и т.п.). Это же касается и псевдокода, который позже преобразуется в комментарии. Старайтесь писать псевдокод так, как будто не знаете, на каком языке программирования будет написана будущая программа.
* Псевдокод написан с расчетом на то, что будущий читатель параллельно будет читать еще и исходный код программы. Например, оперирование в псевдокоде именами переменных, структур, объектов, методов.
* В псевдокоде используется разный стиль описания действий: в одних строках - глаголы («проверить», «создать»), в других - отглагольные существительные («проверка», «создание»). Старайтесь придерживаться единого стиля.
* Попытка в одной строке псевдокода перечислить все действия, которые будут выполняться. Придерживайтесь правила «одна строка псевдокода = одно действие».
* В псевдокоде для одних и тех же сущностей используются разные термины. Например, в матрице то столбец, то колонка (правильно – столбец). Непостоянство терминов сбивает с толку читающего.
* Сначала пишется псевдокод (сверху вниз) и только потом к нему добавляются команды выбранного языка программирования. Соответственно, структура программы получается такая, что главный исходный код (Main) находится вверху. Не надо после этого переворачивать исходный код «вверх ногами» в стиле языка Паскаль. Мало того, что это лишняя работа, так еще и читать такой код гораздо неудобнее.
* Отсутствие пустой строки в качестве разделителя между методами или смысловыми блоками (пустая строка может отсутствовать, только если следующий блок вложен в предыдущий) и отсутствие однострочного комментария в начале смыслового блока или метода.
* При наличии у метода параметров или возвращаемого значения отсутствует их описание. Это касается и Main.
* Пишите комментарии на грамотном русском языке. И помните: слова даже друг от друга принято отделять пробелами, не говоря уже о символах начала комментария.
* Использование snake\_case и других вариаций вместо CamelCase в идентификаторах (поля, методы, классы, имена переменных). Разрешается использовать lowerCamelCase (первое слово – со строчной буквы), если первое слово является префиксом венгерской нотации (например, arrChessGames).
* Использование транслитерации в идентификаторах (Sozdanie, Vivod, Zamena).
* Использование предлогов for, of, to и т.п. в идентификаторах. Идентификаторы следует стараться делать короче: NumberOfRow -> RowNumber. Если без for или to никак не обойтись, то обычно их стараются заменить на 4 и 2 соответственно.
* Имена методов не соответствуют их функционалу. Попробуйте догадаться, что CalculateMatr означает «найти номер столбца минимального элемента на побочной диагонали».
* Комментируйте циклы хотя бы с помощью концевых комментариев:  
   for (…) // по строкам
* Случайные числа – только положительные. Нужны и положительные, и отрицательные числа – главное, чтобы соблюдалась указанная в задании разрядность. Вещественные числа следует округлять до указанной в задании разрядности, иначе при выполнении операций с ними будут накапливаться ошибки.