Отчет по лабораторной работе № 7

“Исследование шаблонов функций”

по дисциплине

ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ

Выполнил студент группы ИВТ/б-22о

Горбенко К.Н.

Проверил:

Тимофеев И.С.

1. **Цель работы:** исследование назначения и способов описания шаблонов функций, применение их при написании объектно-ориентированных программ.
2. **Задание на работу:**
   1. Изучить основы работы с шаблонами функций и классов.
   2. Разработать программу, которая обрабатывает данные разных типов (int, char, и др.). Функция обработки данных должна быть реализована как шаблон.
   3. Разработать тестовые примеры.
   4. Выполнить отладку программы.
   5. Получить результаты работы программы и исследовать ее свойства для различных режимов работы, сформулировать выводы.

**Вариант № 8:** написать функцию-шаблон, переставляющую i-ю и j-ю строки в матрице.

1. **Программа на языке С#.** Класс Extend:

|  |
| --- |
| public static class Extend  {  public static Array SwapLines(this Array array, int i, int j)  {  if (array == null) throw new ArgumentNullException(nameof(array));  if (i == j) throw new InvalidOperationException("Swapping lines should be different");   var temp = array.GetValue(0, 0);   for (int column = 0; column < array.GetLength(0); column++)  {  temp = array.GetValue(i, column);  array.SetValue(array.GetValue(j, column), i, column);  array.SetValue(temp, j, column);  }   return array;  }   public static T[,] SwapLines<T>(T[,] array, int i, int j)  {  if (array == null) throw new ArgumentNullException(nameof(array));  if (i == j) throw new InvalidOperationException("Swapping lines should be different");   var temp = array[0, 0];   for (int column = 0; column < array.GetLength(0); column++)  {  temp = array[i, column];  array[i, column] = array[j, column];  array[j, column] = temp;  }   return array;  }  } |

Класс Program:

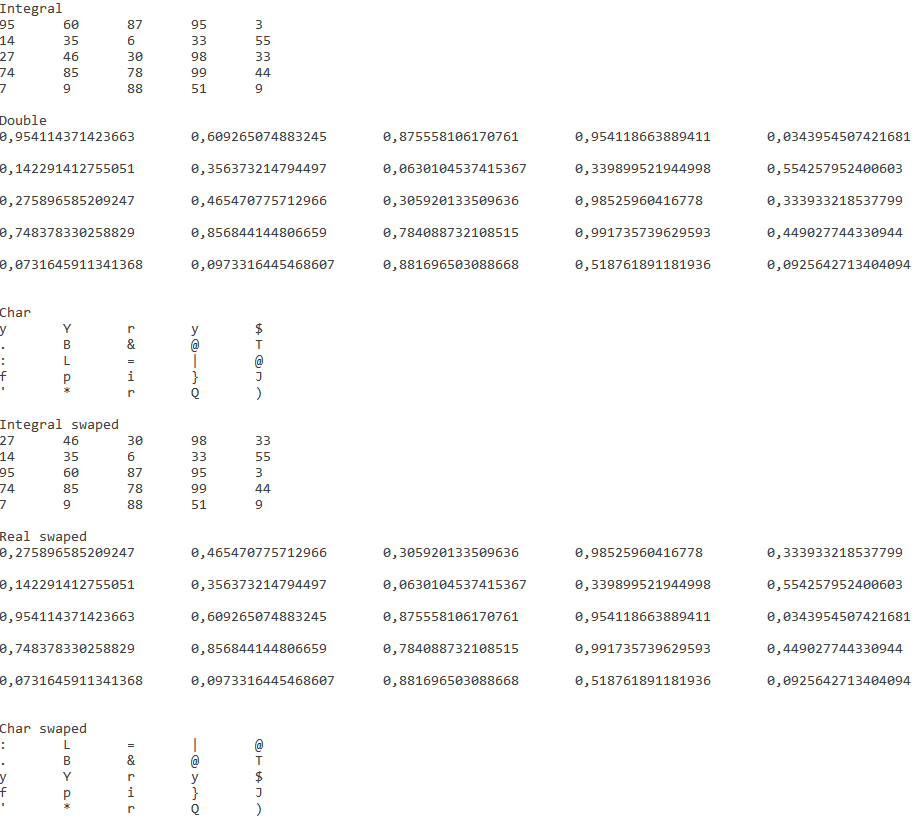
|  |
| --- |
| class Program  {  static void Main(string[] args)  {  var integralMatrix = FillMatrixWithRandomIntegral(new int[5, 5]);  var floatingMatrix = FillMatrixWithRandomFloating(new double[5, 5]);  var characterMatrix = FillMatrixWithRandomCharacter(new char[5, 5]);   WriteLine("Integral");  PrintMatrix(integralMatrix);   WriteLine("Double");  PrintMatrix(floatingMatrix);   WriteLine("Char");  PrintMatrix(characterMatrix);   WriteLine("Integral swaped");  PrintMatrix(Extend.SwapLines(integralMatrix, 0, 2));   WriteLine("Real swaped");  PrintMatrix(Extend.SwapLines(floatingMatrix, 0 ,2));   WriteLine("Char swaped");  PrintMatrix(characterMatrix.SwapLines(0, 2));   ReadLine();  }   static void PrintMatrix(Array array)  {  if (array == null) throw new ArgumentNullException(nameof(array));   for (int i = 0; i < array.GetLength(0); i++)  {  for (int j = 0; j < array.GetLength(1); j++)  Write($"{array.GetValue(i, j)}\t");  WriteLine();  }  WriteLine();  }   public static int[,] FillMatrixWithRandomIntegral(int[,] array)  {  if (array == null) throw new ArgumentNullException(nameof(array));   var random = new Random();   for (int i = 0; i < array.GetLength(0); i++)  {  for (int j = 0; j < array.GetLength(1); j++)  array[i, j] = random.Next(100);  }   return array;  }   public static double[,] FillMatrixWithRandomFloating(double[,] array)  {  if (array == null) throw new ArgumentNullException(nameof(array));   var random = new Random();   for (int i = 0; i < array.GetLength(0); i++)  {  for (int j = 0; j < array.GetLength(1); j++)  array[i, j] = random.NextDouble();  }   return array;  }   public static char[,] FillMatrixWithRandomCharacter(char[,] array)  {  if (array == null) throw new ArgumentNullException(nameof(array));   var random = new Random();   for(int i = 0; i < array.GetLength(0); i++)  {  for (int j = 0; j < array.GetLength(1); j++)  array[i, j] = (char)random.Next(33, 126);  }   return array;  }  } |

1. **Описание программы.**

Для выполнения задания метод, переставляющий i-ю и j-ю строки матрицы, был объявлен generic-методом по начальному условию. Но, учитывая спецификацию задания, был объявлен extension-метод для класса Array, который выполняет ту же функцию, но более удобен в использовании.

Для упрощения тестирования были объявлены методы, заполняющие используемые в лабораторной работе массивы случайными значениями соответствующего типа. Для заполнения каждого типа был создан отдельный метод, принимающий массив соответствующего типа.

Результат работы программы:



1. **Вывод:** в ходе лабораторной работы был разработан метод, осуществляющий перестановку i-й и j-й строк матрицы. Этот метод принимает массив любого типа. Для отладки программы массивы, используемые в лабораторной работе, заполнялись случайными значениями.