Частное учреждение образования

«Колледж бизнеса и права»

Лабораторная работа № 18

Учащийся Е. А. Чужавко

группа: Т-992

2022

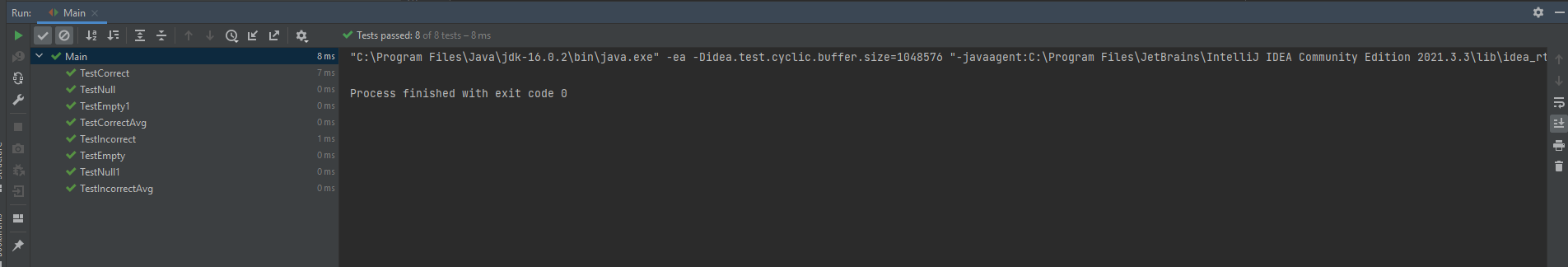
**Задание. Вариант 22**

Дана квадратная матрица. Отсортировать столбцы, индекс которых кратен числу а (вводимому с клавиатуры), по возрастанию. Для остальных найти среднее значение.

Результат работы программы представлен на рисунке 1.

import org.junit.Assert;  
import org.junit.Test;  
  
import java.lang.reflect.Array;  
import java.util.Arrays;  
import java.util.Collection;  
import java.util.Collections;  
import java.util.List;  
  
/\*\*  
 \* Класс, хранящий тесты класса Executor  
 \* @author Чужавко Егор  
 \*/  
public class Main  
{  
 /\*\*  
 \* Тест проверки на правильное возвращаемое значение  
 \*/  
 @Test  
 public void TestCorrect(){  
 var matrix1=Executor.SortColumns(new int[][]{{9,8,7},{6,5,4},{3,2,1}},2);  
 int[][] matrix2=new int[][]{{9,8,1},{6,5,4},{3,2,7}};  
 Assert.assertArrayEquals(matrix1,matrix2);  
 }  
  
  
 /\*\*  
 \* Тест проверки на неправильное возвращаемое значение  
 \*/  
 @Test  
 public void TestIncorrect(){  
 var matrix1=Executor.SortColumns(new int[][]{{9,8,7},{6,5,4},{3,2,1}},2);  
 int[][] matrix2=new int[][]{{3,3,3},{1,1,1},{2,2,2}};  
 Assert.assertNotEquals(matrix1,matrix2);  
 }  
  
  
 /\*\*  
 \* Тест проверки на возвращаемое значение null  
 \*/  
 @Test  
 public void TestNull(){  
 var matrix1=Executor.SortColumns(null,2);  
 int[][] matrix2=null;  
 Assert.assertArrayEquals(matrix1,matrix2);  
 }  
  
 /\*\*  
 \* Тест проверки на пустое возвращаемое значение  
 \*/  
 @Test  
 public void TestEmpty(){  
 var matrix1=Executor.SortColumns(new int[][]{},2);  
 int[][] matrix2=null;  
 Assert.assertArrayEquals(matrix1,matrix2);  
 }  
  
  
 /\*\*  
 \* Тест проверки на правильное возвращаемое значение  
 \*/  
 @Test  
 public void TestCorrectAvg(){  
 int[] matrix1=Executor.FindAvg(new int[][]{{8,8,7},{6,5,4},{3,2,1}},2);  
 int[] matrix2=new int[]{5};  
 Assert.assertArrayEquals(matrix1,matrix2);  
 }  
  
 /\*\*  
 \* Тест проверки на неправильное возвращаемое значение  
 \*/  
 @Test  
 public void TestIncorrectAvg(){  
 int[] matrix1=Executor.FindAvg(new int[][]{{8,8,7},{6,5,4},{3,2,1}},2);  
 int[] matrix2=new int[]{2};  
 Assert.assertNotEquals(matrix1,matrix2);  
 }  
  
 /\*\*  
 \* Тест проверки на возвращаемое значение null  
 \*/  
 @Test  
 public void TestNull1(){  
 var matrix1=Executor.FindAvg(null,2);  
 int[] matrix2=null;  
 Assert.assertArrayEquals(matrix1,matrix2);  
 }  
  
 /\*\*  
 \* Тест проверки на пустое возвращаемое значение  
 \*/  
 @Test  
 public void TestEmpty1(){  
 var matrix1=Executor.FindAvg(new int[][]{},2);  
 int[] matrix2=null;  
 Assert.assertArrayEquals(matrix1,matrix2);  
 }  
}

import java.util.ArrayList;  
import java.util.Arrays;  
  
  
/\*\*  
 \* Класс для выполнения тестов функций  
 \* Дана квадратная матрица. Отсортировать столбцы, индекс которых  
 \* кратен числу а (вводимому с клавиатуры), по возрастанию. Для  
 \* остальных найти среднее значение.  
 \* @author Чужавко Егор  
 \*/  
public class Executor  
{  
 /\*\*  
 \* Функция для сортировки колонок, индексы которых кратны вводимому числу  
 \* @param matrix сама матрица, которая будет сортироваться  
 \* @param number число, которое будет сравниваться с индексом столбца на кратность  
 \* @return возвращает отсортированный массив  
 \*/  
 public static int[][] SortColumns(int[][] matrix, int number){  
 if(matrix==null ||matrix.length==0){  
 return null;  
 }  
 for(int j =0;j<matrix.length;j++){  
 if(j%number==0 & j>0){  
 for(int e=1;e<=2;e++){  
 for(int i =0;i<matrix.length-1;i++){  
 if(matrix[i][j]>matrix[i+1][j]){  
 int temp=matrix[i][j];  
 matrix[i][j]=matrix[i+1][j];  
 matrix[i+1][j]=temp;  
 }  
 }  
 }  
 }  
 }  
 return matrix;  
 }  
  
 /\*\*  
 \* Функция для нахождения среднего числа столбцов, индексы которых не кратны вводимому числу  
 \* @param matrix сама матрица, хранящая данные  
 \* @param number число, которое будет сравниваться с индексом столбца на кратность  
 \* @return возвращает массив средних чисел  
 \*/  
 public static int[] FindAvg(int[][] matrix,int number){  
 if(matrix==null ||matrix.length==0){  
 return null;  
 }  
 ArrayList<Integer> al=new ArrayList<Integer>();  
 for(int j =0;j<matrix.length;j++){  
 if(j%number!=0 & j>0){  
 al.add(j);  
 }  
 }  
 int ch=0;  
 int index=0;  
 int[]avg=new int[al.size()];  
 for(Integer q: al){  
 ch=0;  
 for(int i =0;i<matrix.length;i++){  
 ch+=matrix[q][i];  
 }  
 avg[index]=ch/matrix.length;  
 index++;  
 }  
 return avg;  
 }  
}



*Рисунок 1.*