

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
«БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

ФАКУЛЬТЕТ ЭЛЕКТРОННО-ИНФОРМАЦИОННЫХ СИСТЕМ

Кафедра интеллектуальных информационных технологий

Отчет по лабораторной работе №1

Специальность ПО9(3)

Выполнил
Д. Н. Кухарев,
студент группы ПО9

Проверил
А. А. Крощенко,
ст. преп. кафедры ИИТ,
«__k_____2024 г.

Брест 2024

Цель работы: научиться разрабатывать простейшие программы на языке программирования Java, получить практический опыт работы с компилятором javac.

Вариант 8

Задание 1. Для переданной в качестве параметра последовательности из N целых чисел написать утилиту, выводящую все уникальные числа.

Задание 2. Написать метод `randomPerturbations(double[][] matrix)`, который переставляет элементы матрицы `matrix` случайным образом.

Задание 3. Написать метод `boolean isAllLowerCase(String cs)`, проверяющий все ли символы строки в нижнем регистре или нет.

Выполнение:

Код программы

Main.java:

```
public class Main {
    public static void main(String[] args){
        String array_task1 = args[0];
        String cs = args[1];
        System.out.println("Task1: ");
        LabTasks lab1 = new LabTasks();
        lab1.showUnique(lab1.ParceStringToIntArray(array_task1));

        System.out.println("\nTask2: ");
        double[][] matrix = {{1, 3, 5}, {4, 6, 8}, {1, 4, 7}};
        lab1.SetMatrix(matrix);
        System.out.println("Initial matrix: ");
        lab1.Show();
        System.out.println("\nRandom element position matrix: ");
        lab1.randomPetrubations();
        lab1.Show();

        System.out.println("\nTask3: ");
        if(lab1.isAllLowerCase(cs)){
            System.out.print("All symbols are lower case\n");
        }else{
            System.out.print(lab1.isAllLowerCase("Not all symbols are lower case\n"));
        }
    }
}
```

LabTasks.java:

```
public class LabTasks {
    private double[][] matrix;
```

```

public IntDynamicArray ParseStringToIntArray(String arg){
    IntDynamicArray parsed_array = new IntDynamicArray();
    String[] tmp_array = new String[arg.length()];
    for(int i = 0; i < arg.length(); ++i){
        tmp_array[i] = "";
    }
    int tmp_size = 0; boolean isNew = false;
    char[] digit_check = new char[arg.length()];
    arg.getChars(0, arg.length(), digit_check, 0);

    for(int i = 0; i < digit_check.length; ++i){
        if(Character.isDigit(digit_check[i]) || digit_check[i] == '-'){
            isNew = true;
            tmp_array[tmp_size] += digit_check[i];
        } else if(digit_check[i] == '/'){
            break;
        }
        else if(isNew){
            parsed_array.add(Integer.parseInt(tmp_array[tmp_size]));
            ++tmp_size;
            isNew = false;
        }
    }
    parsed_array.add(Integer.parseInt(tmp_array[tmp_size]));

    return parsed_array;
}

public void showUnique(IntDynamicArray args){
    IntDynamicArray taskarr = new IntDynamicArray();
    ShowArray(args);
    for(int i = 0; i < args.size(); ++i){
        if(!taskarr.find(args.get(i))){
            taskarr.add(args.get(i));
        }
    }
    System.out.print("Unique elements array: ");
    taskarr.show();
}

public void ShowArray(IntDynamicArray args){
    System.out.print("Initial array:      ");
    System.out.print(' ');
    for(int i = 0; i < args.size(); ++i){
        if(i < args.size()-1){
            System.out.print(args.get(i) + ", ");
        }
    }
}

```

```

        else{
            System.out.print(args.get(i));
        }
    }
    System.out.print("\n");
}

public void SetMatrix(double[][] argMatrix){
    int arrWidth = argMatrix.length, arrHeight = argMatrix[0].length;
    matrix = new double[arrWidth][arrHeight];
    for(int i = 0; i < arrWidth; ++i){
        System.arraycopy(argMatrix[i], 0, matrix[i], 0, arrHeight);
    }
}

public void randomPetrubations(){
    int arrWidth = matrix.length, arrHeight = matrix[0].length;
    double[][] new_matrix = new double[arrWidth][arrHeight];
    double[] wttmp = new double[arrHeight];
    double tmp;
    int rand_pos;
    for(int i = arrWidth - 1; i >= 0; --i){
        rand_pos = (int) (Math.random() * i);
        System.arraycopy(matrix[i], 0, wttmp, 0, arrHeight);
        System.arraycopy(matrix[rand_pos], 0, matrix[i], 0, arrHeight);
        System.arraycopy(wtmp, 0, matrix[rand_pos], 0, arrHeight);
    }
    for(int i = 0; i < arrWidth; ++i){
        for(int j = arrHeight - 1; j >= 0; --j){
            rand_pos = (int) (Math.random() * j);
            tmp = matrix[i][j];
            matrix[i][j] = matrix[i][rand_pos];
            matrix[i][rand_pos] = tmp;
        }
    }
}

public void Show(){
    for(int i = 0; i < matrix.length; ++i){
        System.out.print('| ');
        for(int j = 0; j < matrix[i].length; ++j){
            if(j < matrix[j].length-1){
                System.out.print(matrix[i][j] + ", ");
            }
            else{
                System.out.print(matrix[i][j]);
            }
        }
    }
}

```

```

        System.out.print('|'); System.out.println();
    }
}
public boolean isAllLowerCase(String cs){
    System.out.println("Line is: " + cs);
    if(cs == null){
        return false;
    }
    if(cs.toLowerCase() == cs.toUpperCase()){
        return false;
    }
    if(cs == cs.toLowerCase()){
        return true;
    }else{
        return false;
    }
}
}

```

IntDynamicArray.java:

```

public class IntDynamicArray {
    private int[] array;
    private int size;
    private int capacity;
    public IntDynamicArray(){
        capacity = 10;
        array = new int[capacity];
        size = 0;
    }
    public int size(){
        return size;
    }
    public boolean find(int number){
        for(int i = 0; i < size; ++i){
            if(array[i] == number){
                return true;
            }
        }
        return false;
    }
    public void resize(){
        capacity *= 2;
        int[] newArray = new int[capacity];
        System.arraycopy(array, 0, newArray, 0, size);
        array = newArray;
    }
}

```

```

public void add(int number){
    if(size == capacity){
        resize();
    }
    array[size] = number;
    ++size;
}
public void remove(int index){
    if(index < 0 || index >= size){
        throw new IndexOutOfBoundsException("Index out of bounds: " + index);
    }
    for(int i = index; i < size-1; ++i){
        array[i] = array[i+1];
    }
    --size;
}
public int get(int index){
    if(index < 0 || index >= size){
        throw new IndexOutOfBoundsException("Index out of bounds: " + index);
    }
    return array[index];
}
public void show(){
    System.out.print('[');
    for(int i = 0; i < size; ++i){
        if(i < size-1){
            System.out.print(array[i] + ", ");
        }
        else{
            System.out.print(array[i]);
        }
    }
    System.out.print(']'); System.out.println();
}
}

```

Спецификация ввода

>java Main “<Строка для задания №1 + символ конца строки />” “<Строка для задания №3>”

Пример

>java Main “abraKadabra”

Рисунки с результатами работы программы

```
1 public class Main {
2     public static void main(String[] args){
3         String array_task1 = args[0];
4         String cs = args[1];
5         System.out.println("Task1: ");
6         LabTasks lab1 = new LabTasks();
7         lab1.showUnique(lab1.ParceStringToIntArray(array_task1));
8
9         System.out.println("\nTask2: ");
10        double[][] matrix = {{1, 3, 5}, {4, 6, 8}, {1, 4, 7}};
11        lab1.SetMatrix(matrix);
12        System.out.println("Initial matrix: ");
13        lab1.Show();
14        System.out.println("\nRandom element position matrix: ");
15        lab1.randomPetrubations();
16        lab1.Show();
17
18        System.out.println("\nTask3: ");
19        if(lab1.isAllLowerCase(cs)){
20            System.out.print("All symbols are lower case\n");
21        }else{
22            System.out.print(lab1.isAllLowerCase("Not all symbols are lower case\n"));
23        }
24    }
25 }
```

Скомпилируем программу.

```
C:\Users\Dmitriy\Desktop>cd JavaProjects
C:\Users\Dmitriy\Desktop\JavaProjects>cd Learning
C:\Users\Dmitriy\Desktop\JavaProjects\Learning>cd src
C:\Users\Dmitriy\Desktop\JavaProjects\Learning\src>javac Main.java LabTasks.java IntDynamicArray.java
```

Запустим программу и проверим Task3 с разными параметрами.

```
Task3:
Line is: saduhg
All symbols are lower case

C:\Users\Dmitriy\Desktop\JavaProjects\Learning\src>javac Main.java LabTasks.java IntDynamicArray.java

C:\Users\Dmitriy\Desktop\JavaProjects\Learning\src>java Main "1, 3, 434, -3, 3, 1, 2, 2, 43/" "abraKadabra"
Task1:
Initial array:          [1, 3, 434, -3, 3, 1, 2, 2, 43]
Unique elements array: [1, 3, 434, -3, 2, 43]

Task2:
Initial matrix:
|1.0, 3.0, 5.0|
|4.0, 6.0, 8.0|
|1.0, 4.0, 7.0|

Random element position matrix:
|6.0, 8.0, 4.0|
|7.0, 1.0, 4.0|
|3.0, 5.0, 1.0|

Task3:
Line is: abraKadabra
Line is: Not all symbols are lower case
false
```

Вывод: научился разрабатывать простейшие программы на языке программирования Java, получил практический опыт работы с компилятором javac.