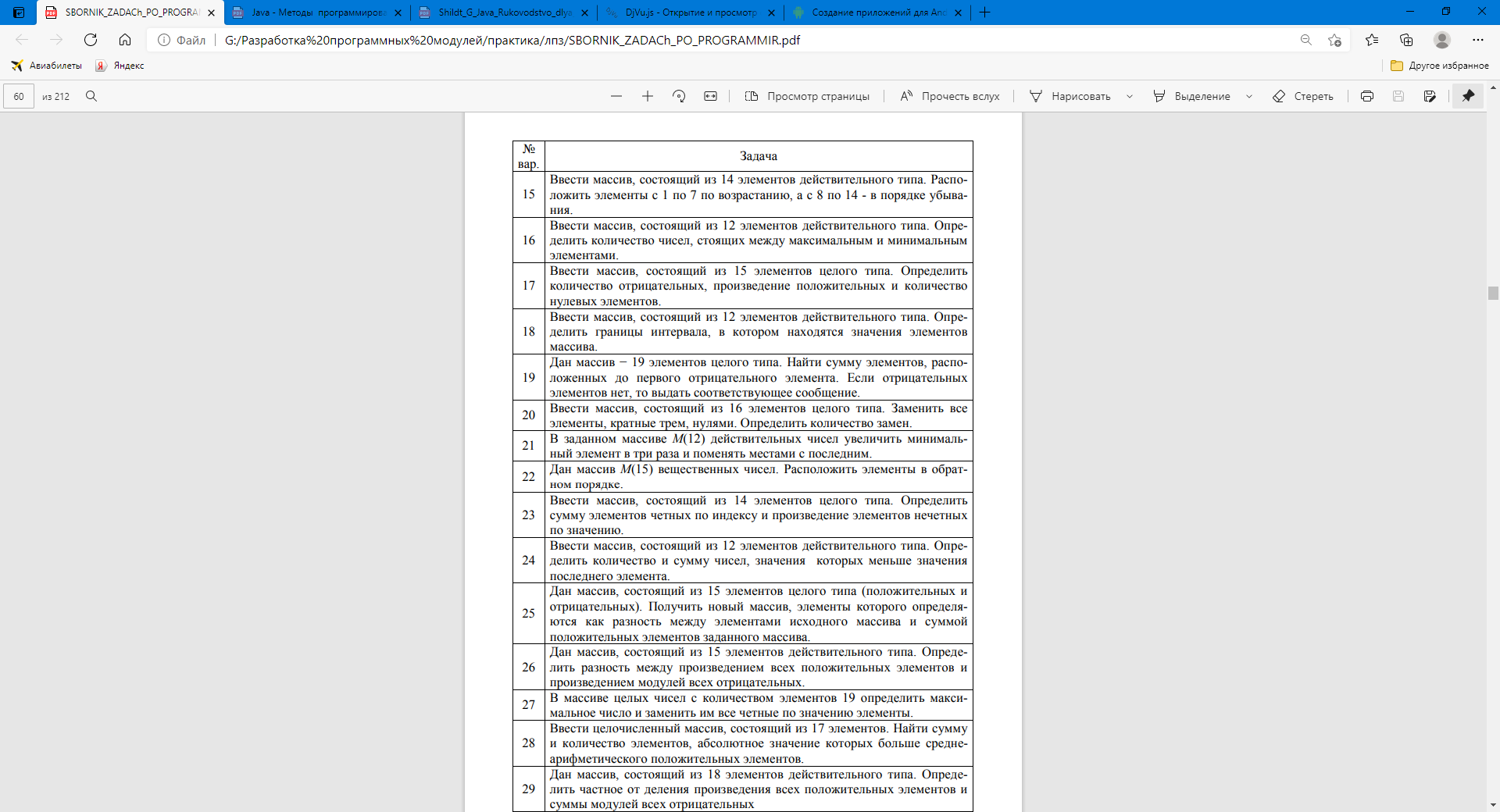
**Лабораторная работа №13**

**Основы программирования на Java: одномерные массивы**

**Цель работы:** Освоить основные способы создания Java-программ либо с помощью обычного редактора, либо с помощью среды разработки. Приобрести навыки программирования с использованием одномерных массивов в Java.

Вариант 17

**Задача №1**



|  |  |  |
| --- | --- | --- |
| Наименование переменной | Тип данных | Назначение |
| array | Int | Входная |
| negativeCount | Выходная |
| positiveProduct |
| zeroCount |

Начало

Ввод массив

int i = 0; i < array.length; i++

Ввод значений в массив

int num : array

Ввод значений в массив

int num : array

ДА

НЕТ

Конец

num > 0

ДА

НЕТ

negativeCount++

zeroCount++

positiveProduct \*= num

Выв negativeCount, positiveProduct, zeroCount

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

// Создаем массив из 15 элементов

int[] array = new int[15];

// Вводим элементы массива с клавиатуры

System.out.println("Введите 15 элементов массива:");

for (int i = 0; i < array.length; i++) {

array[i] = scanner.nextInt();

}

// Переменные для хранения количества отрицательных, произведения положительных и количества нулевых элементов

int negativeCount = 0;

int positiveProduct = 1;

int zeroCount = 0;

// Проходим по массиву и выполняем необходимые вычисления

for (int num : array) {

if (num < 0) {

negativeCount++;

} else if (num > 0) {

positiveProduct \*= num;

} else {

zeroCount++;

}

}

// Выводим результаты

System.out.println("Количество отрицательных элементов: " + negativeCount);

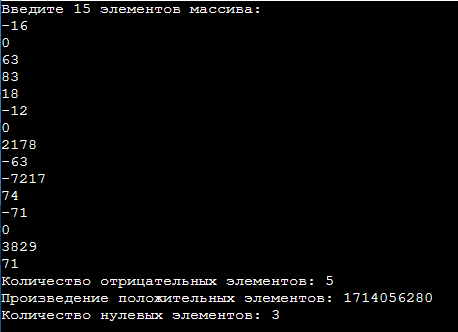
System.out.println("Произведение положительных элементов: " + positiveProduct);

System.out.println("Количество нулевых элементов: " + zeroCount);

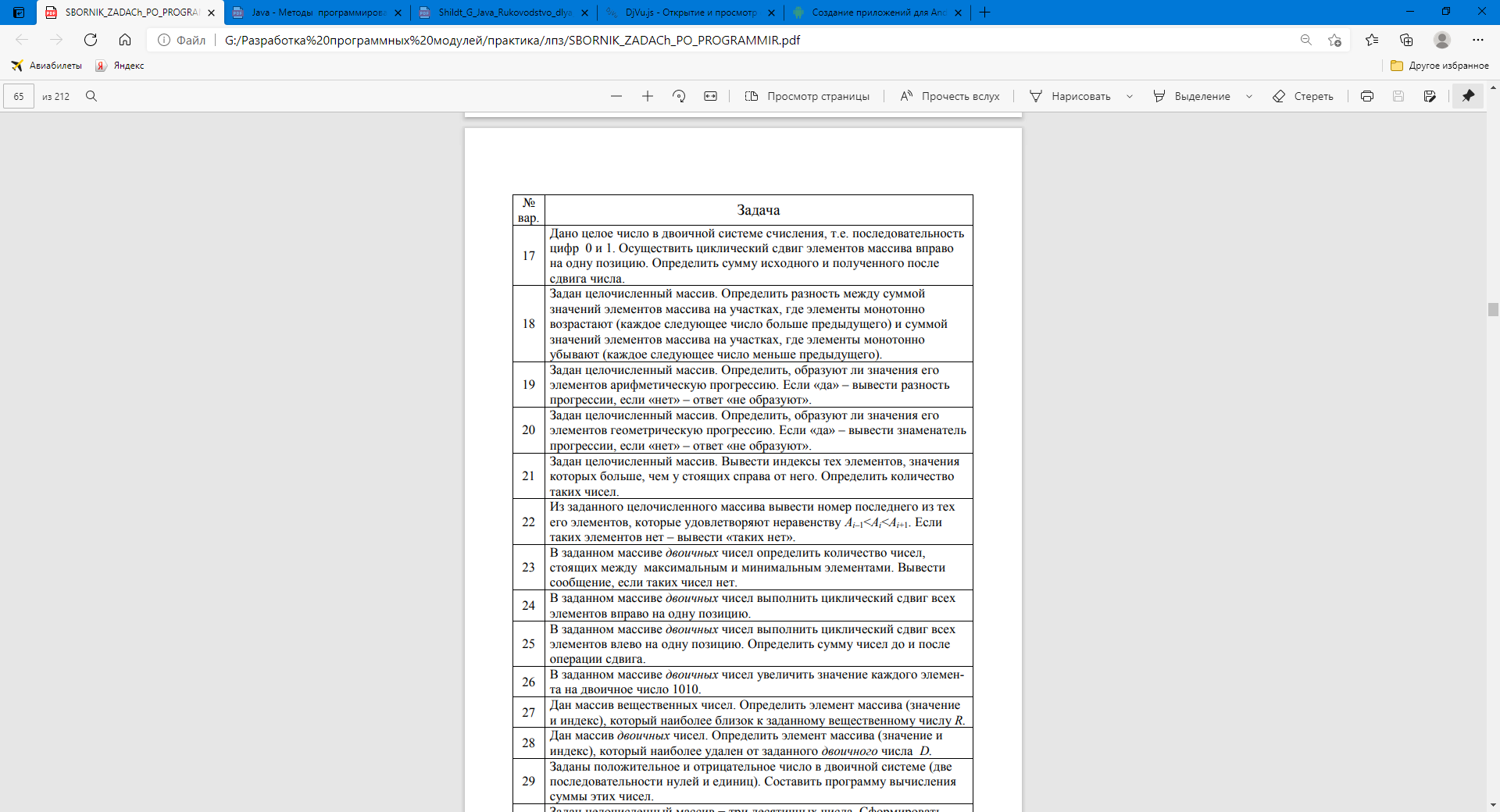
scanner.close();

}

}



**Задача №2**



|  |  |  |
| --- | --- | --- |
| Наименование переменной | Тип данных | Назначение |
| binaryString | String | Входная |
| shiftedBinaryString | Вспомогательная |
| binaryArray | char |
| lastChar |
| originalDecimal | int |
| shiftedDecimal |
| sum | Выходная |

Начало

Ввод двоич число

Преобразуем строку в массив

Выводим сумму

Двоичные в целые числа и сумма

Выводим binaryArray

int i = binaryArray.length - 1; i > 0; i--

Конец

Массив в строку

Вносим получившееся в binaryArray

циклический сдвиг вправо

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

// Вводим двоичное число как строку

System.out.print("Введите двоичное число: ");

String binaryString = scanner.nextLine();

// Преобразуем двоичную строку в массив символов

char[] binaryArray = binaryString.toCharArray();

// Выполняем циклический сдвиг вправо

char lastChar = binaryArray[binaryArray.length - 1];

for (int i = binaryArray.length - 1; i > 0; i--) {

binaryArray[i] = binaryArray[i - 1];

}

binaryArray[0] = lastChar;

// Выводим результат после сдвига

System.out.print("Результат циклического сдвига вправо: ");

System.out.println(binaryArray);

// Преобразуем массив символов обратно в строку

String shiftedBinaryString = new String(binaryArray);

// Преобразуем двоичные строки в целые числа и вычисляем их сумму

int originalDecimal = Integer.parseInt(binaryString, 2);

int shiftedDecimal = Integer.parseInt(shiftedBinaryString, 2);

int sum = originalDecimal + shiftedDecimal;

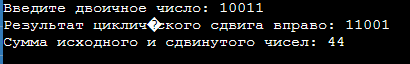
// Выводим сумму исходного и полученного числа после сдвига

System.out.println("Сумма исходного и сдвинутого чисел: " + sum);

scanner.close();

}

}



**Контрольные вопросы для защиты**

1. Что такое массив?

упорядоченный набор элементов одного типа данных, который обладает фиксированным размером. Элементы массива доступны по их индексу.

1. Как объявить массив в Java?

// Объявление массива целых чисел

int[] myArray;

// Объявление массива строк

String[] stringArray;

1. Как обратиться к элементу массива? Примеры.

Для доступа к элементу массива в Java используется индекс элемента.

int[] numbers = {1, 2, 3, 4, 5};

System.out.println(numbers[0]); // Выведет первый элемент массива, т.е. 1

String[] fruits = {"Apple", "Banana", "Orange"};

System.out.println(fruits[1]); // Выведет второй элемент массива, т.е. “Banana”

1. Способы инициализации массива.

* Инициализация при объявлении:

int[] numbers = {1, 2, 3};

* Инициализация с использованием конструктора:

int[] numbers = new int[3]; // Создает массив из 3 элементов, заполненный нулями

* Инициализация значением по умолчанию:

int[] numbers = new int[]{1, 2, 3}; // Частичное заполнение массива