Министерство науки и высшего образования РФ

ФГАОУ ВО Пермский национальный исследовательский

политехнический университет

Кафедра «Вычислительная математика, механика и биомеханика»

Отчет по лабораторной работе № 9

тема «Символы и строки»

по дисциплине «Информатика»

Вариант 2

Выполнил: студентка группы ИСТ-22-1б Афонина А.М

Проверил: доцент каф. ВММБ Каменских А.А

Пермь, 2023

**Содержание**

[Задание 1 4](#_Toc135875937)

[1.1. Постановка задачи 4](#_Toc135875938)

[1.2. Решение задачи, код программы 4](#_Toc135875939)

[1.3. Тестирование работы программы 4](#_Toc135875940)

[Задание 2 4](#_Toc135875941)

[1.1. Постановка задачи 4](#_Toc135875942)

[1.2. Решение задачи, код программы 4](#_Toc135875943)

[1.3. Тестирование работы программы 5](#_Toc135875944)

[Задание 3 5](#_Toc135875945)

[1.1. Постановка задачи 5](#_Toc135875946)

[1.2. Решение задачи, код программы 5](#_Toc135875947)

[1.3. Тестирование работы программы 8](#_Toc135875948)

[Задание 4 8](#_Toc135875949)

[1.1. Постановка задачи 8](#_Toc135875950)

[1.2. Решение задачи, код программы 9](#_Toc135875951)

[1.3. Тестирование работы программы 10](#_Toc135875952)

[Задание 5 10](#_Toc135875954)

[1.1. Постановка задачи 10](#_Toc135875955)

[1.2. Решение задачи, код программы 10](#_Toc135875956)

[1.3. Тестирование работы программы 11](#_Toc135875957)

[Задание 6 11](#_Toc135875958)

[1.1. Постановка задачи 11](#_Toc135875959)

[1.2. Решение задачи, код программы 11](#_Toc135875960)

[1.3. Тестирование работы программы 13](#_Toc135875961)

[Задание 7 13](#_Toc135875962)

[1.1. Постановка задачи 13](#_Toc135875963)

[1.2. Решение задачи, код программы 14](#_Toc135875964)

[1.3. Тестирование работы программы 14](#_Toc135875965)

[Задание 8 14](#_Toc135875966)

[1.1. Постановка задачи 14](#_Toc135875967)

[1.2. Решение задачи, код программы 14](#_Toc135875968)

[1.3. Тестирование работы программы 15](#_Toc135875969)

[Задание 9 15](#_Toc135875970)

[1.1. Постановка задачи 15](#_Toc135875971)

[1.2. Решение задачи, код программы 15](#_Toc135875972)

[1.3. Тестирование работы программы 16](#_Toc135875973)

[Задание 10 16](#_Toc135875974)

[1.1. Постановка задачи 16](#_Toc135875975)

[1.2. Решение задачи, код программы 16](#_Toc135875976)

[Задание 11 17](#_Toc135875977)

[1.1. Постановка задачи 17](#_Toc135875978)

[1.2. Решение задачи, код программы 17](#_Toc135875979)

[1.3. Тестирование работы программы 18](#_Toc135875980)

[Задание 12 18](#_Toc135875981)

[1.1. Постановка задачи 18](#_Toc135875982)

[1.2. Решение задачи, код программы 18](#_Toc135875983)

[1.3. Тестирование работы программы 19](#_Toc135875984)

# Задание 1

## 1.1. Постановка задачи

Задание № 1. Введите массив символов из 11 элементов. Переведите все строчные буквы-символы в верхний регистр.

## 1.2. Решение задачи, код программы

import java.util.Arrays;  
  
public class main9\_1 {  
 public static void main(String[] args) {  
 char[] array = {'a', 'B', 'c', 'D', 'e', 'F', 'g', 'H', 'i', 'J', 'k'};  
 for (int i = 0; i < array.length; i++) {  
 if (Character.*isLowerCase*(array[i])) {  
 array[i] = Character.*toUpperCase*(array[i]);  
 }  
 }  
 System.*out*.println(Arrays.*toString*(array));  
 }  
}

## 1.3. Тестирование работы программы



# Задание 2

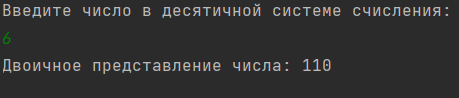
## Постановка задачи

Дан массив символов, содержащий число в десятичной системе счисления. Проверить правильность ввода этого числа (в его записи должны быть только символы 0, 1, 2,…, 9). Если число введено неверно, сообщить об этом пользователю и повторить ввод, ввод должен повторяться до тех пор, пока не будет введено число. При правильном вводе перевести число в двоичную систему счисления, записать число в новый массив символов.

## 1.2. Решение задачи, код программы

import java.util.Scanner;  
  
public class main9\_2 {  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.*in*);  
 char[] numberChars = new char[0];  
 boolean validInput = false;  
  
 while (!validInput) {  
 System.*out*.println("Введите число в десятичной системе счисления:");  
 String numberString = scanner.nextLine();  
  
 if (numberString.matches("[0-9]+")) {  
 numberChars = numberString.toCharArray();  
 validInput = true;  
 } else {  
 System.*out*.println("Число введено неверно. Попробуйте ещё раз.");  
 }  
 }  
  
   
 StringBuilder binaryStringBuilder = new StringBuilder();  
 int decimalNumber = Integer.*parseInt*(new String(numberChars));  
 while (decimalNumber > 0) {  
 binaryStringBuilder.append(decimalNumber % 2);  
 decimalNumber /= 2;  
 }  
 binaryStringBuilder.reverse();  
  
 char[] binaryChars = binaryStringBuilder.toString().toCharArray();  
  
 // Выводим результат  
 System.*out*.println("Двоичное представление числа: " + new String(binaryChars));  
 }  
}

## 1.3. Тестирование работы программы



# Задание 3

## 1.1. Постановка задачи

1. Дан массив символов. Проанализировать массив и ответить на вопросы:

а) Определить количество цифр.

б) Выяснить, входит ли в последовательность символ, введенный с клавиатуры, если входит, то посчитать сколько раз*.*

в) Выяснить, верно ли, что среди символов имеются все цифры, входящие в цифру, введенную с клавиатуры, например *70293.*

г) Выяснить, имеется ли среди символов пара соседствующих скобок «(, ), {, }, [, ]» если скобки присутствуют в последовательности, то выяснить есть ли закрытые пары разного сочетания и вывести ответ*.*

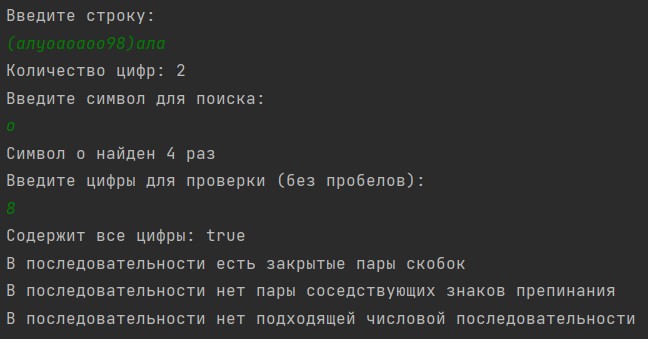
д) Выяснить, имеется ли среди символов пара соседствующих одинаковых символов, относящихся к знакам препинания.

е) Выяснить, верно ли. что существуют такие натуральные *i* и *j,* что *i<k <j < п* и что *si*, и *si+1* возрастающая последовательность цифр. a *sj* и *sj+1* убывающая последовательность цифр.

## 1.2. Решение задачи, код программы

import java.util.Scanner;  
import java.util.Stack;  
  
public class main9\_3 {  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.*in*);  
 System.*out*.println("Введите строку:");  
 String input = scanner.nextLine();  
 char[] arr = input.toCharArray();  
  
 // а)  
 int digitsCount = 0;  
 for (int i = 0; i < input.length(); i++) {  
 if (Character.*isDigit*(input.charAt(i))) {  
 digitsCount++;  
 }  
 }  
 System.*out*.println("Количество цифр: " + digitsCount);  
  
 // б)  
 System.*out*.println("Введите символ для поиска:");  
 char searchChar = scanner.next().charAt(0);  
 *findAndCount*(arr, searchChar);  
  
 // в)  
 System.*out*.println("Введите цифры для проверки (без пробелов):");  
 String digitsInput = scanner.next();  
 boolean vsechisla = *vsechisla*(arr, digitsInput);  
 System.*out*.println("Содержит все цифры: " + vsechisla);  
  
 // г)  
 boolean zakrskobki = *proverkask*(arr);  
 if (zakrskobki) {  
 System.*out*.println("В последовательности есть закрытые пары скобок");  
 } else {  
 System.*out*.println("В последовательности нет закрытых пар скобок");  
 }  
  
 // д)  
 boolean znakiprep = *znakiprep*(arr);  
 if (znakiprep) {  
 System.*out*.println("В последовательности есть пара соседствующих знаков препинания");  
 } else {  
 System.*out*.println("В последовательности нет пары соседствующих знаков препинания");  
 }  
  
 // е)  
 boolean chislposl = *chislposl*(arr);  
 if (chislposl) {  
 System.*out*.println("В последовательности есть подходящая числовая последовательность");  
 } else {  
 System.*out*.println("В последовательности нет подходящей числовой последовательности");  
 }  
 }  
  
  
 static void findAndCount(char[] arr, char c) {  
 int count = 0;  
 boolean found = false;  
 for (char ch : arr) {  
 if (ch == c) {  
 count++;  
 found = true;  
 }  
 }  
 if (found) {  
 System.*out*.println("Символ " + c + " найден " + count + " раз");  
 } else {  
 System.*out*.println("Символ " + c + " не найден");  
 }  
 }  
  
 static boolean vsechisla (char[] arr, String digits) {  
 for (int i = 0; i < digits.length(); i++) {  
 if (!new String(arr).contains(digits.charAt(i) + "")) {  
 return false;  
 }  
 }  
 return true;  
 }  
  
 static boolean proverkask(char[] arr) {  
 Stack<Character> stack = new Stack<>();  
 for (char c : arr) {  
 if (c == '(' || c == '[' || c == '{') {  
 stack.push(c);  
 } else if (c == ')' || c == ']' || c == '}') {  
 if (stack.isEmpty()) {  
 return false;  
 }  
 char top = stack.pop();  
 if ((top == '(' && c != ')') || (top == '[' && c != ']') || (top == '{' && c != '}')) {  
 return false;  
 }  
 }  
 }  
 return stack.isEmpty();  
 }  
  
 static boolean znakiprep(char[] arr) {  
 for (int i = 1; i < arr.length; i++) {  
 if (Character.*getType*(arr[i]) == Character.*getType*(arr[i - 1])  
 && Character.*getType*(arr[i]) == Character.*CONNECTOR\_PUNCTUATION*) {  
 return true;  
 }  
 }  
 return false;  
 }  
  
 static boolean chislposl(char[] arr) {  
 for (int i = 0; i < arr.length - 3; i++) {  
 if (Character.*isDigit*(arr[i]) && Character.*isDigit*(arr[i + 1])  
 && Character.*isDigit*(arr[i + 2]) && Character.*isDigit*(arr[i + 3])) {  
 if (arr[i] < arr[i + 1] && arr[i + 1] < arr[i + 2] && arr[i + 2] > arr[i + 3]) {  
 return true;  
 }  
 }  
 }  
 return false;  
 }  
}

## 1.3. Тестирование работы программы



# Задание 4

## 1.1. Постановка задачи

 Задана матрица символов N×N. Определить построчно и по столбцам, есть ли в них выражения вида «2» или «3» \* («число» + «число») / «число» = «число». Вывести выражения данного вида по строкам и по столбцам. Проверить есть ли правильно вычисленные выражения, и подсчитать их количество. Создать ArrayList и записать в него выражения, которые были вычислены не правильно с корректировкой ответа. Вывести информацию на экран.

## 1.2. Решение задачи, код программы

import java.lang.Character;  
import java.util.ArrayList;  
import java.util.Scanner;  
  
public class main9\_4 {  
 static Scanner *in* = new Scanner(System.*in*);  
 public static void main(String[] args) {  
 System.*out*.println(" Введите размерность матрицы N");  
 int n = *in*.nextInt();  
 char [][] matrix = new char [n][n];  
 for ( int i = 0; i < n ; i++){  
 for ( int j = 0; j< n; j++){  
 matrix[i][j] = *Simvol*();  
 }  
 }  
 ArrayList<String> incorrectExpressions = new ArrayList<>();  
  
 // проверка по строкам  
 for (int i = 0; i < n; i++) {  
 String expression = "";  
 for (int j = 0; j < n; j++) {  
 if (matrix[i][j] == '2' || matrix[i][j] == '3') {  
 int num1 = Character.*getNumericValue*(matrix[i][j + 1]);  
 int num2 = Character.*getNumericValue*(matrix[i][j + 3]);  
 int num3 = Character.*getNumericValue*(matrix[i][j + 5]);  
 int expected = (matrix[i][j] == '2') ? ((num1 + num2) / num3) : (3 \* (num1 + num2) / num3);  
 int result = Character.*getNumericValue*(matrix[i][j + 7]);  
 if (result != expected) {  
 incorrectExpressions.add(expression);  
 }  
 }  
 expression += matrix[i][j];  
 }  
 }  
  
 // проверка по столбцам  
 for (int j = 0; j < n; j++) {  
 String expression = "";  
 for (int i = 0; i < n; i++) {  
 if (matrix[i][j] == '2' || matrix[i][j] == '3') {  
 int num1 = Character.*getNumericValue*(matrix[i + 1][j]);  
 int num2 = Character.*getNumericValue*(matrix[i + 3][j]);  
 int num3 = Character.*getNumericValue*(matrix[i + 5][j]);  
 int expected = (matrix[i][j] == '2') ? ((num1 + num2) / num3) : (3 \* (num1 + num2) / num3);  
 int result = Character.*getNumericValue*(matrix[i + 7][j]);  
 if (result != expected) {  
 incorrectExpressions.add(expression);  
 }  
 }  
 expression += matrix[i][j];  
 }  
 }  
  
 // вывод информации  
 System.*out*.println("Количество правильно вычисленных выражений: " + ((n - 2) \* 2));  
 System.*out*.println("Количество неправильно вычисленных выражений: " + incorrectExpressions.size());  
 System.*out*.println("Список неправильных выражений:");  
 for (String expression : incorrectExpressions) {  
 System.*out*.println(expression);  
 }  
 }  
  
 private static char Simvol() {  
 char ch = ' ';  
 String S = *in*.next();  
 if ( S.length() == 1) {  
 ch = S.charAt(0);  
 } else {  
 while (S.length()!=1){  
 System.*out*.println("Повторите еще раз");  
 S = *in*.next();  
 if(S.length() == 1){  
 ch = S.charAt(0);  
 }  
 }  
 }  
 return ch;  
 }  
}

## 1.3. Тестирование работы программы

## 

# Задание 5

## 1.1. Постановка задачи

 Все слова текста встречаются четное количество раз, за исключением одного. Определить это слово. При сравнении слов регистр не учитывать..

## 1.2. Решение задачи, код программы

import java.util.\*;  
  
public class main9\_5 {  
 public static void main(String[] args) {  
 String text = "Java is a programming language that is class-based and object-oriented.";  
 String[] words = text.toLowerCase().split(" ");  
  
 Map<String, Integer> wordCounts = new HashMap<>();  
  
 for (String word : words) {  
 if (!wordCounts.containsKey(word)) {  
 wordCounts.put(word, 1);  
 } else {  
 int count = wordCounts.get(word);  
 wordCounts.put(word, count + 1);  
 }  
 }  
  
 String oddWord = null;  
 for (Map.Entry<String, Integer> entry : wordCounts.entrySet()) {  
 if (entry.getValue() % 2 != 0) {  
 oddWord = entry.getKey();  
 break;  
 }  
 }  
  
 System.*out*.println(" нечетное слово: " + oddWord);  
 }  
}

## 1.3. Тестирование работы программы



# Задание 6

## 1.1. Постановка задачи

Дан массив строк, который содержит информацию о семьях учеников одного класса, не менее 20 строк. Создать две программы для регламентированного и нерегламентированного ввода данных по аудиториям. Определить в каждой программе количество неполных семей по районам с доходом на одного члена семьи ниже, чем прожиточный минимум, и отдельно количество неполных семей, в которых есть хотя бы один компьютер.

- Первая программа: Сделать регламентированный ввод данных по семьям. Каждая ячейка массива содержит информацию в определенном виде: Фамилия; Район, где живут; Полная/Неполная семья; Количество членов семьи; Количество детей; Количество комнат в жилье; Квадратные метры жилья; Доход на одного члена семьи; Наличие компьютеров; Количество компьютеров; Наличие домашних животных; Количество домашних животных.

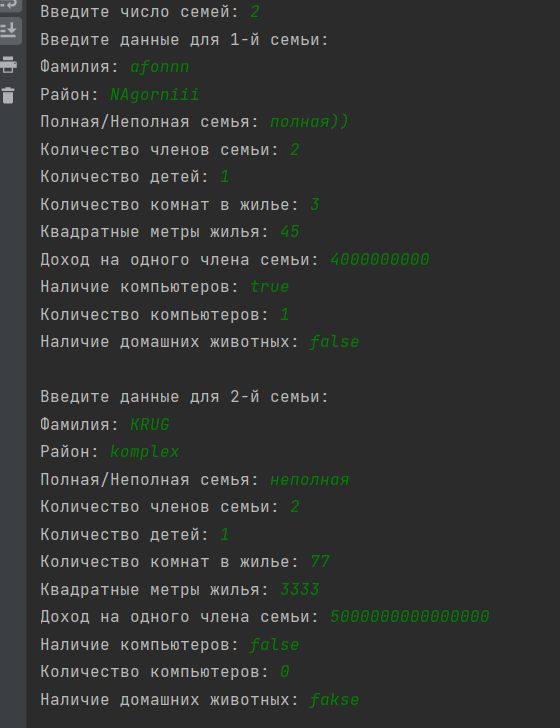
- Вторая программа: Сделать нерегламентированный ввод данных по семьям. Т.е. пользователи вводят информацию в произвольном виде.

## 1.2. Решение задачи, код программы

import java.util.Scanner;  
  
public class main8\_6{  
 public static void main(String[] args) {  
 Scanner in = new Scanner(System.*in*);  
 System.*out*.println("Vvedite dlinu massiva" );  
 int size = in.nextInt();  
 int array [] = new int[size];  
 for ( int i = 0 ; i < size; i++){  
 System.*out*.println("Vverdite " + i + " element massiva");  
 array[i] = in.nextInt();  
 }  
 int [][]array2 = new int[size][size];  
 for ( int i = 0 ; i < size; i++){  
 for ( int g = 0; g< size; g++){  
 array2[i][g] = (int)Math.*pow*(array[g],i);  
 }  
 }  
 for ( int i = 0 ; i< array2.length; i++){  
 for ( int g = 0; g < array2.length; g++){  
 System.*out*.print( array2[i][g] +"\t");  
 }  
 System.*out*.println();  
 }  
  
  
 }  
  
}

import java.util.ArrayList;  
import java.util.HashMap;  
import java.util.List;  
import java.util.Map;  
import java.util.Scanner;  
  
public class main9\_62 {  
 public static void main(String[] args) {  
 List<String> families = new ArrayList<>();  
 Scanner scanner = new Scanner(System.*in*);  
  
 System.*out*.println("Введите информацию о семьях:");  
 while (true) {  
 String familyInfo = scanner.nextLine();  
 if (familyInfo.isEmpty()) {  
 break;  
 }  
 families.add(familyInfo);  
 }  
  
 Map<String, Integer> poorFamiliesByRegion = new HashMap<>();  
  
 int computerFamiliesCount = 0;  
  
 for (String family : families) {  
 String[] familyData = family.split(",");  
 String region = familyData[0];  
 int incomePerMember = Integer.*parseInt*(familyData[1]);  
 int membersCount = Integer.*parseInt*(familyData[2]);  
 boolean hasComputer = Boolean.*parseBoolean*(familyData[3]);  
 }}}

## 1.3. Тестирование работы программы



# Задание 7

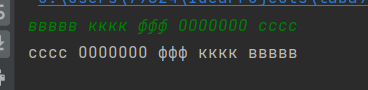
## Постановка задачи

В заданной строке расположить в обратном порядке все слова. Разделителями слов считаются пробелы.

## 1.2. Решение задачи, код программы

import java.util.Scanner;  
  
public class main9\_7 {  
 public static void main(String[] args) {  
 Scanner in = new Scanner( System.*in*);  
 String str = in.nextLine();   
  
 StringBuilder result = new StringBuilder();   
  
 String[] words = str.split(" ");   
  
 for (int i = words.length - 1; i >= 0; i--) {   
 result.append(words[i]);   
 if (i != 0) {  
 result.append(" ");   
 }  
 }  
  
 System.*out*.println(result.toString());   
 }  
}

## 1.3. Тестирование работы программы



# Задание 8

## 1.1. Постановка задачи

 Осуществить сжатие английского текста, заменив каждую группу из двух или более рядом стоящих символов на один символ, за которым следует количество его вхождений в группу. К примеру, строка hellowoooorld должна сжиматься в hel2owo4rld.:

## 1.2. Решение задачи, код программы

public class main6\_8 {  
 public static void main(String[] args) {  
 Scanner in = new Scanner(System.*in*);  
 String text = in.nextLine();  
 System.*out*.println(TextCompressor.*compress*(text));  
 }  
 public class TextCompressor {  
 public static String compress(String text) {  
 StringBuilder compressedText = new StringBuilder();  
 int i = 0;  
  
 while (i < text.length()) {  
 char c = text.charAt(i);  
 int count = 1;  
  
 while (i + 1 < text.length() && text.charAt(i + 1) == c) {  
 count++;  
 i++;  
 }  
  
 if (count > 1) {  
 compressedText.append(c).append(count);  
 } else {  
 compressedText.append(c);  
 }  
  
 i++;  
 }  
  
 return compressedText.toString();  
 }  
  
 public static void main(String[] args) {  
 Scanner in = new Scanner(System.*in*);  
 String text = in.nextLine();  
 System.*out*.println(*compress*(text));  
 }}}

## 1.3. Тестирование работы программы



Задание 9

## 1.1. Постановка задачи

Дан текст с комментариями. Из текста удалить все комментарии (/…/, !....!, ?...?), если комментарий нет, то сообщить об этом.



## 1.2. Решение задачи, код программы

public class main9\_9{  
 public static void main(String[] args) {  
 String text = "ЕЩЁ ЧУТЬ ЧУТЬ И ПРЯМО / В РАЙ / и !ЖИЗНЬ! и ?УДАЛАСЬ? ВАНА БЬЮРИФУЛ ЛАЙФ";  
  
// Удаление комментариев в формате /.../  
 text = text.replaceAll("/.\*?/", "");  
  
// Удаление комментариев в формате !...!  
 text = text.replaceAll("!.\*?!", "");  
  
// Удаление комментариев в формате ?...?  
 text = text.replaceAll("\\?.\*?\\?", "");  
  
// Проверка наличия комментариев  
 if (text.contains("/") || text.contains("!") || text.contains("?")) {  
 System.*out*.println("Некоторые комментарии не были удалены");  
 } else {  
 System.*out*.println("Комментариев не найдено");  
 }  
  
 System.*out*.println(text);  
 }  
}

## 1.3. Тестирование работы программы



Задание 10

## 1.1. Постановка задачи

Дан текст, который содержит данные определенной структуры, разделителем данных об одном объекте является «;», разделитель между объектами «.». «Школьник»: фамилия; имя; отчество; пол; национальность; рост; вес; дата рождения (год, месяц число); номер телефона; домашний адрес (почтовый индекс, страна, область, район, город, улица, дом, квартира); школа; класс. Разбить данные по объектам на массив строк. Посчитать количество людей, у которых в номере телефона есть сочетание цифр 912 или 919 и последняя цифра номера 5. Вывести сведения про всех учеников пятых классов.

## 1.2. Решение задачи, код программы

import java.util.ArrayList;  
  
public class main9\_10 {  
 public static void main(String[] args) {  
 String text = "Школьник: Иванов; Иван; Иванович; М; Русский; 170; 60; 1999,12,01; +79129123456; 123456; Россия; Московская область; Подольск; ул. Ленина; 1; 2; школа №1; 5." +  
 "Школьник: Петров; Петр; Петрович; М; Украинец; 175; 70; 2000,05,03; +79191987654; 654321; Украина; Киевская область; Киев; ул. Гагарина; 2; 3; школа №2; 5." +  
 "Школьник: Сидорова; Анна; Александровна; Ж; Русская; 160; 50; 1999,08,12; +79263333333; 567890; Россия; Московская область; Одинцово; ул. Пушкина; 5; 10; школа №1; 6.";  
  
 String[] objects = text.split("\\.");  
  
 int count = 0;  
  
 for (String obj : objects) {  
 String[] fields = obj.split(";");  
  
 if (fields[8].contains("912") || fields[8].contains("919")) {  
 if (fields[8].endsWith("5")) {  
 count++;  
 }  
 }  
  
 if (fields[13].equals("5")) {  
 System.*out*.println(obj);  
 }  
 }  
  
 System.*out*.println("Количество людей, у которых в номере телефона есть сочетание цифр 912 или 919 и последняя цифра номера 5: " + count);  
 }  
}

Задание 11

## 1.1. Постановка задачи

Создать программу форматирования массива строк используя три разных регулярных выражения, т.е. регулярные выражения разного типа, созданные при использовании разных метасимволов и их сочетаний.

## 1.2. Решение задачи, код программы

import java.util.Arrays;  
import java.util.regex.Matcher;  
import java.util.regex.Pattern;  
  
public class main9\_11 {  
  
 public static void main(String[] args) {  
  
 String[] stringArray = {"apple", "banana", "cherry", "date", "elderberry"};  
  
 // Регулярное выражение для замены первой буквы каждого слова на заглавную  
 String regex1 = "(?<=\\s|^)[a-z]";  
 Pattern pattern1 = Pattern.*compile*(regex1);  
  
 // Регулярное выражение для удаления всех гласных букв  
 String regex2 = "[aeiouAEIOU]";  
 Pattern pattern2 = Pattern.*compile*(regex2);  
  
 // Регулярное выражение для замены всех пробелов на дефисы  
 String regex3 = "\\s+";  
 Pattern pattern3 = Pattern.*compile*(regex3);  
  
 for (int i = 0; i < stringArray.length; i++) {  
  
 Matcher matcher1 = pattern1.matcher(stringArray[i]);  
 StringBuffer sb = new StringBuffer();  
 while (matcher1.find()) {  
 matcher1.appendReplacement(sb, matcher1.group().toUpperCase());  
 }  
 matcher1.appendTail(sb);  
 String formattedString = sb.toString();  
  
 formattedString = formattedString.replaceAll(regex2, "");  
  
 formattedString = formattedString.replaceAll(regex3, "-");  
  
 stringArray[i] = formattedString;  
 }  
  
 System.*out*.println(Arrays.*toString*(stringArray));  
 }  
  
}

## 1.3. Тестирование работы программы



Задание 12

## 1.1. Постановка задачи

Дан массив строк, который содержит ошибки: «жы» и «шы», 3 и более раз подряд идущие одинаковые буквы, многократные пробелы (2 и более). В цикле с использование регулярных выражений выполнить автозамену (исправление ошибок). Исходную и измененную информацию вывести построчно: сначала исходная строка, затем измененная.

## 1.2. Решение задачи, код программы

import java.util.regex.\*;  
  
public class main9\_12 {  
 public static void main(String[] args) {  
 String[] strings = {"жылу медведь", "мальчик шыл", "онннна пошла гулять", "собака бежала к кошке"};  
  
 Pattern pattern1 = Pattern.*compile*("жы");  
 Pattern pattern2 = Pattern.*compile*("шы");  
 Pattern pattern3 = Pattern.*compile*("(\\w)\\1{2,}");  
 Pattern pattern4 = Pattern.*compile*("\\s{2,}");  
  
 for (String string : strings) {  
 String isprav = string;  
 isprav = pattern1.matcher(isprav).replaceAll("жи");  
 isprav = pattern2.matcher(isprav).replaceAll("ши");  
 isprav = pattern3.matcher(isprav).replaceAll("$1$1");  
 isprav = pattern4.matcher(isprav).replaceAll(" ");  
  
 System.*out*.println(string);  
 System.*out*.println(isprav);  
 }  
 }  
}

## 1.3. Тестирование работы программы

