Министерство науки и высшего образования РФ

ФГАОУ ВО Пермский национальный исследовательский

политехнический университет

Кафедра «Вычислительная математика, механика и биомеханика»

Отчет по лабораторной работе № 7

тема «Строки и символы»

по дисциплине «Информатика»

Вариант 20

Выполнил: студент группы ИСТ-22-2б Тлевасов Н.С.

Проверил: ассистент каф. ВММБ Носов Ю.О.

Пермь, 2023

**Содержание**

[Задание 1 4](#_Toc131100493)

[1.1. Постановка задачи 4](#_Toc131100494)

[1.2. Решение задачи, код программы 4](#_Toc131100495)

[1.3. Тестирование работы программы 4](#_Toc131100496)

[Задание 2 6](#_Toc131100497)

[2.1. Постановка задачи 6](#_Toc131100498)

[2.2. Решение задачи, код программы 6](#_Toc131100499)

[2.3. Тестирование работы программы 6](#_Toc131100500)

[Задание 3 8](#_Toc131100501)

[3.1. Постановка задачи 8](#_Toc131100502)

[3.2. Решение задачи, код программы 8](#_Toc131100503)

[3.3. Тестирование работы программы с проверкой 10](#_Toc131100504)

[Задание 4 11](#_Toc131100505)

[4.1. Постановка задачи 11](#_Toc131100506)

[4.2. Решение задачи, код программы 11](#_Toc131100507)

[4.3. Тестирование работы программы 14](#_Toc131100508)

[Задание 5 17](#_Toc131100509)

[5.1. Постановка задачи 17](#_Toc131100510)

[5.2. Решение задачи, код программы 17](#_Toc131100511)

[5.3. Тестирование работы программы 17](#_Toc131100512)

[Задание 6 18](#_Toc131100513)

[6.1. Постановка задачи 18](#_Toc131100514)

[6.2. Решение задачи, код программы 18](#_Toc131100515)

[6.3. Тестирование работы программы с проверкой 22](#_Toc131100516)

[Задание 7 24](#_Toc131100517)

[7.1. Постановка задачи 24](#_Toc131100518)

[7.2. Решение задачи, код программы 24](#_Toc131100519)

[7.3. Тестирование работы программы 24](#_Toc131100520)

[Задание 8 25](#_Toc131100521)

[8.1. Постановка задачи 25](#_Toc131100522)

[8.2. Решение задачи, код программы 25](#_Toc131100523)

[8.3. Тестирование работы программы 25](#_Toc131100524)

[Задание 9 26](#_Toc131100525)

[9.1. Постановка задачи 26](#_Toc131100526)

[9.2. Решение задачи, код программы 26](#_Toc131100527)

[9.3. Тестирование работы программы с проверкой 26](#_Toc131100528)

[Задание 10 27](#_Toc131100529)

[10.1. Постановка задачи 27](#_Toc131100530)

[10.2. Решение задачи, код программы 27](#_Toc131100531)

[10.3. Тестирование работы программы с проверкой 28](#_Toc131100532)

[Задание 11 29](#_Toc131100533)

[11.1. Постановка задачи 29](#_Toc131100534)

[11.2. Решение задачи, код программы 29](#_Toc131100535)

[11.3. Тестирование работы программы с проверкой 29](#_Toc131100536)

[Задание 12 30](#_Toc131100537)

[12.1. Постановка задачи 30](#_Toc131100538)

[12.2. Решение задачи, код программы 30](#_Toc131100539)

[12.3. Тестирование работы программы с проверкой 31](#_Toc131100540)

# Задание 1

## Постановка задачи

Дана матрица символов размером N×M. Найти сколько раз по строчкам и по столбцам встречаются знаки препинания «,», «.», «?».

## Решение задачи, код программы

1. import java.util.Objects;  
   import java.util.Scanner;  
   public class work\_1 {  
    public static void main(String[] args) {  
    Scanner mc = new Scanner(System.*in*);  
    Scanner str = new Scanner(System.*in*);  
    System.*out*.println("Введите размерность матрицы M\*N");  
    int m =mc.nextInt(), n=mc.nextInt(),cnt1=0, cnt2=0, cnt3=0;  
    String [][] mas\_1= new String [m][n];  
    for (int i=0;i < m;i++){  
    for (int j=0;j < n;j++){  
    System.*out*.println("Вводите по одному символу!!!");  
    mas\_1[i][j]=str.next();  
    while(mas\_1[i][j].length()!=1){  
    System.*out*.println("Вводите по одному символу!!!");  
    mas\_1[i][j]=str.next();  
    }  
    }  
    }  
    for (int i=0;i < m;i++){  
    for (int j=0;j < n;j++){  
    if (Objects.*equals*(mas\_1[i][j], ",")){cnt1++;}  
    if (Objects.*equals*(mas\_1[i][j], ".")){cnt2++;}  
    if (Objects.*equals*(mas\_1[i][j], "?")){cnt3++;}  
    }  
    }  
    System.*out*.println("Количество символов (,) = " + cnt1 +" Количество символов (.) = " + cnt2 + " Количество символов (?) = " + cnt3);  
    }  
   }

## 

## 1.3. Тестирование работы программы

Тестирование работы программы и проверка результатов решения

|  |  |
| --- | --- |
| № п.п. | Решение Java |
| 1 |  |

Сравнение решения задачи с увиденным показала, что решение задачи в Java правильно Данный факт подтверждает правильность написанного кода программы. Неопределенностей при решении задачи выявлено не было, возможно данная ситуация связанна с малым количеством проверок данных.

Задание 2

## 2.1. Постановка задачи

. Дан массив символов, содержащий число в троичной системе счисления. Проверить правильность ввода этого числа (в его записи должны быть только символы 0, 1 и 2). 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 12-ую систему счисления, записать число в новый массив символов.

## 2.2. Решение задачи, код программы

import java.math.BigInteger;  
import java.util.Objects;  
import java.util.Scanner;  
  
import static java.lang.Math.*pow*;  
  
public class work\_2 {  
 public static void main(String[] args) {  
 Scanner mc = new Scanner(System.*in*);  
 System.*out*.println("Сколько элементов в вашем числе?");  
 int n= mc.nextInt();  
 String[] mch = new String[n];  
 StringBuilder m = new StringBuilder();  
 for (int i = 0; i < n;i++){  
 System.*out*.println("Введите число в 3-ой СС, тобишь(0,1,2)");  
 mch[i]= mc.next();  
 while ((!Objects.*equals*(mch[i], "0") || !Objects.*equals*(mch[i], "1") || !Objects.*equals*(mch[i], "2")) & mch[i].length()!=1){  
 System.*out*.println("Ввод неверен, ещё раз!");  
 mch[i]= mc.next();  
 }  
 m.append(mch[i]);  
 }  
 int x = 0,k=0;  
 while(k!=n){  
 for (int i = n-1;i >=0;i--){  
 x+=(Integer.*parseInt*(String.*valueOf*(m.charAt(i))))\**pow*(3,k);  
 k++;  
 }  
 }  
 String l = "" + x;  
 System.*out*.println("Ввод произведён успешно, переводим ваше число!");  
 BigInteger b =new BigInteger(new BigInteger(l).toString(12),10);  
 String o= ""+b;  
 System.*out*.println(o);  
 }  
}

## 2.3. Тестирование работы программы

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Java | Решение онлайн-калькулятора |
| 1 |  |  |
| 2 |  |  |

# Задание 3

## 3.1. Постановка задачи

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.

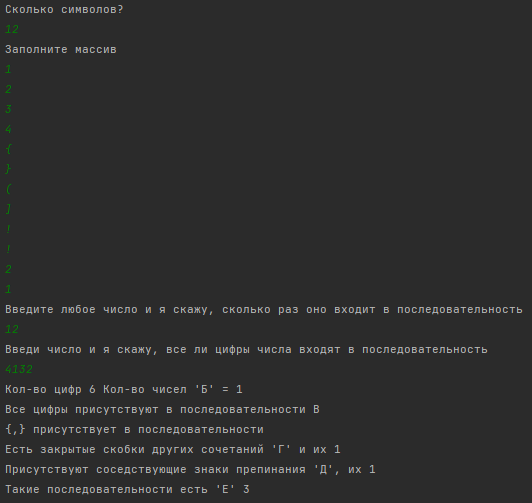
д) Выяснить, имеется ли среди символов пара соседствующих одинаковых символов, относящихся к знакам препинания.

е) Выяснить, верно ли. что существуют такие натуральные  и , что  и что , и возрастающая последовательность цифр, a  и убывающая последовательность цифр.

## 3.2. Решение задачи, код программы

import java.util.Objects;  
import java.util.Scanner;  
import java.util.regex.Matcher;  
import java.util.regex.Pattern;  
  
public class work\_3 {  
 public static void main(String[] args) {  
 Scanner mc = new Scanner(System.*in*);  
 System.*out*.println("Сколько символов?");  
 int n = mc.nextInt();  
 String[] mas = new String[n];  
 System.*out*.println("Заполните массив");  
 for (int i = 0; i < n; i++) {  
 mas[i] = mc.next();  
 while (mas[i].length() != 1) {  
 System.*out*.println("Вводи массив по одному!!!");  
 mas[i] = mc.next();  
 }  
 }  
 System.*out*.println("Введите любое число и я скажу, сколько раз оно входит в последовательность");  
 String chisB = mc.next();  
 System.*out*.println("Введи число и я скажу, все ли цифры числа входят в последовательность");  
 String chisV = mc.next();  
 int cntCifr = 0, cntV = 0;  
 for (int i = 0; i < n; i++) {  
 if (*equa*(mas[i])) {  
 cntCifr++;  
 }  
 }  
 System.*out*.println("Кол-во цифр " + cntCifr + " Кол-во чисел 'Б' = " + *count*(String.*join*("", mas), chisB));  
 for (int i = 0; i < chisV.length(); i++) {  
 if (*cifriVchis*(String.*join*("", mas), chisV)) {  
 cntV++;  
 }  
 }  
 if (cntV == chisV.length()) {  
 System.*out*.println("Все цифры присутствуют в последовательности В");  
 } else System.*out*.println("Не все числа присутствуют в последовательности В");  
  
 int cnt = 0; //Задание Г  
 for (int i = 0; i < mas.length - 1; i++) {  
 if (Objects.*equals*(mas[i], "{") & Objects.*equals*(mas[i + 1], "}")) {  
 System.*out*.println("{,} присутствует в последовательности");  
 }  
 if (Objects.*equals*(mas[i], "(") & Objects.*equals*(mas[i + 1], ")")) {  
 System.*out*.println("(,) присутствует в последовательности");  
 }  
 if (Objects.*equals*(mas[i], "[") & Objects.*equals*(mas[i + 1], "]")) {  
 System.*out*.println("[,] присутствует в последовательности");  
 }  
 if ((Objects.*equals*(mas[i], "[") & Objects.*equals*(mas[i + 1], "}")) ||  
 Objects.*equals*(mas[i], "(") & Objects.*equals*(mas[i + 1], "}") ||  
 Objects.*equals*(mas[i], "[") & Objects.*equals*(mas[i + 1], ")") ||  
 Objects.*equals*(mas[i], "{") & Objects.*equals*(mas[i + 1], ")") ||  
 Objects.*equals*(mas[i], "(") & Objects.*equals*(mas[i + 1], "]") ||  
 Objects.*equals*(mas[i], "{") & Objects.*equals*(mas[i + 1], "]")) {  
 cnt++;  
 }  
 }  
 if (cnt > 0) {  
 System.*out*.println("Есть закрытые скобки других сочетаний 'Г' и их " + cnt);  
 } else {  
 System.*out*.println("Закрытых скобок других сочетаний нет 'Г' ");  
 }  
 //Задание Д  
 int cnt1 = 0;  
 for (int i = 0; i < mas.length - 1; i++) {  
 if (*equal*(mas, i) & (Objects.*equals*(mas[i], mas[i + 1]))) {  
 cnt1++;  
 }  
 }  
 if (cnt1 > 0) {  
 System.*out*.println("Присутствуют соседствующие знаки препинания 'Д', их " + cnt1);  
 } else {  
 System.*out*.println("Нет соседствующих знаков препинания 'Д' ");  
 }  
 int cnt2 = 0; //E  
 for (int i = 0; i < mas.length - 2; i++) {  
 for (int j = i + 1; j < mas.length - 1; j++) {  
 if (*equa*(mas[i]) & *equa*(mas[i + 1]) & *equa*(mas[j]) & *equa*(mas[j + 1])) {  
 if (Integer.*parseInt*(mas[i]) + 1 == Integer.*parseInt*(mas[i + 1]) & Integer.*parseInt*(mas[j]) - 1 ==Integer.*parseInt*(mas[j + 1])) {  
 cnt2++;  
 }  
 }  
 }  
 }  
 if (cnt > 0) {  
 System.*out*.println("Такие последовательности есть 'Е' " + cnt2);  
 } else {  
 System.*out*.println("Такие последовательностей нет 'Е' ");  
 }  
 }  
 private static boolean equa(String m) {  
 if (m == null) {  
 return false;  
 }  
 try {  
 double d = Double.*parseDouble*(m);  
 }  
 catch (NumberFormatException nfe) {  
 return false;  
 }  
 return true;  
 }  
 private static boolean equal(String[] x, int n) {  
 if (!Objects.*equals*(x[n], ".")||!Objects.*equals*(x[n], ",")||!Objects.*equals*(x[n], "!")||!Objects.*equals*(x[n], "?")) {  
 return true;  
 }  
 return false;  
 }  
 public static int count(String str, String target) {  
 return (str.length() - str.replace(target, "").length()) / target.length();  
 }  
 public static boolean cifriVchis(String x, String n){  
 for(int i=0;i<n.length();i++) {  
 for (int j = 0; j < x.length(); j++) {  
 if (x.charAt(j) == n.charAt(i)) {  
 return true;  
 }  
 }  
 }  
 return false;  
 }  
 public static String numb(String [] m,int n){  
 return m[n];  
 }  
}

## 3.3. Тестирование работы программы с проверкой



Задание 4

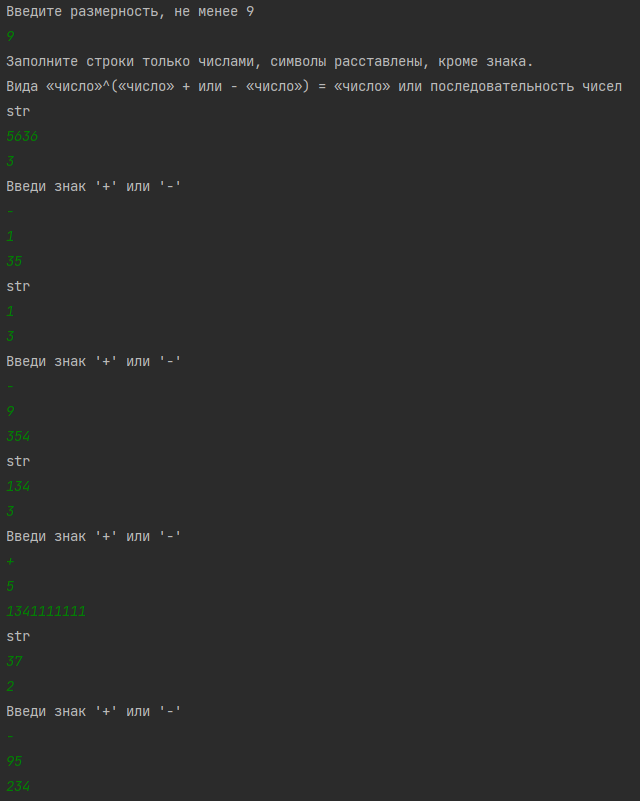
## 4.1. Постановка задачи

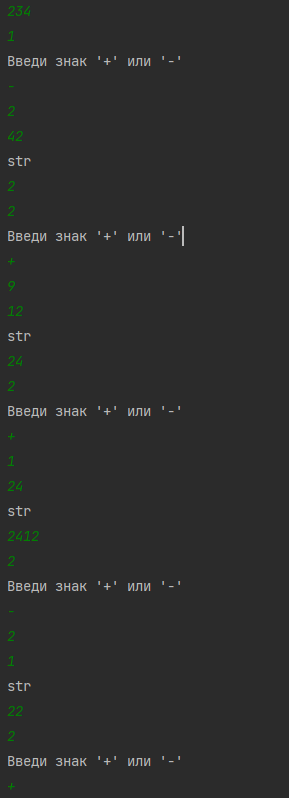
Задана матрица символов N×N. Определить построчно и по столбцам, есть ли в них выражения вида «число»^(«число» + или - «число») = «число» или последовательность чисел. Вывести выражения данного вида по строкам и по столбцам. Проверить есть ли правильно вычисленные выражения, и подсчитать их количество.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

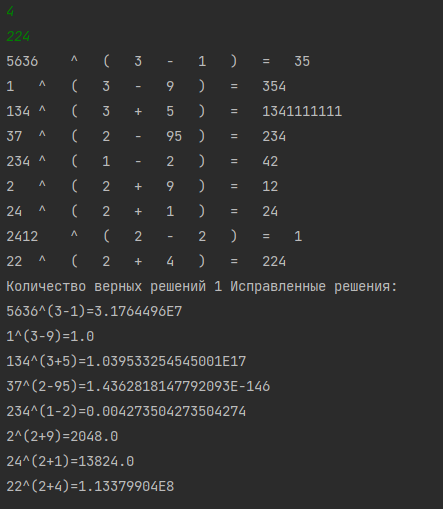
## 4.2. Решение задачи, код программы

import java.util.ArrayList;  
import java.util.Objects;  
import java.util.Scanner;  
import static java.lang.Integer.*parseInt*;  
import java.lang.String;  
public class work\_4 {  
 public static void main(String[] args) {  
 Scanner mc = new Scanner(System.*in*);  
 System.*out*.println("Введите размерность, не менее 9");  
 int n = mc.nextInt();  
 String[][] mas = new String[n][n];  
 System.*out*.println("Заполните строки только числами, символы расставлены, кроме знака.\n" +  
 "Вида «число»^(«число» + или - «число») = «число» или последовательность чисел");  
 for (int i = 0; i < n; i++) {  
 System.*out*.println("str");  
 for (int j = 0; j < n; j++) {  
 if (j == 1) {  
 mas[i][j] = "^";  
 j++;  
 }  
 if (j == 2) {  
 mas[i][j] = "(";  
 j++;  
 }  
 if (j == 4) {  
 System.*out*.println("Введи знак '+' или '-'");  
 mas[i][j] = mc.next();  
 j++;  
 }  
 if (j == 6) {  
 mas[i][j] = ")";  
 j++;  
 }  
 if (j == 7) {  
 mas[i][j] = "=";  
 j++;  
 }  
 mas[i][j] = mc.next();  
 }  
 }  
 int cnt = 0;  
 ArrayList<String> ArrayList = new ArrayList<String>();  
 for (int i = 0; i < n; i++) {  
 if (*chislo\_Li*(mas[i][0]) & *chislo\_Li*(mas[i][3]) & *chislo\_Li*(mas[i][5]) & *chislo\_Li*(mas[i][8])  
 & Objects.*equals*(mas[i][1], "^") & Objects.*equals*(mas[i][2], "(") & (Objects.*equals*(mas[i][4], "+") ||  
 Objects.*equals*(mas[i][4], "-")) & Objects.*equals*(mas[i][6], ")") & Objects.*equals*(mas[i][7], "=")) {  
 *printStr*(mas, i);  
 if (*vivodSTR*(mas, i)) {  
 cnt++;  
 } else {  
 *mas\_to\_ArrSTR*(mas, ArrayList, i);  
 }  
 } else if (*chislo\_Li*(mas[0][i]) & *chislo\_Li*(mas[3][i]) & *chislo\_Li*(mas[5][i]) & *chislo\_Li*(mas[8][i])  
 & Objects.*equals*(mas[1][i], "^") & Objects.*equals*(mas[2][i], "(") & (Objects.*equals*(mas[4][i], "+") ||  
 Objects.*equals*(mas[4][i], "-")) & Objects.*equals*(mas[6][i], ")") & Objects.*equals*(mas[7][i], "=")) {  
 *printStol*(mas, i);  
 if (*vivodSTOL*(mas, i)) {  
 cnt++;  
 } else {  
 *masArrSTOL*(mas, ArrayList, i);  
 }  
 }  
 }  
 System.*out*.println("Количество верных решений " + cnt + " Исправленные решения:" + "\r");  
 *printArr*(ArrayList);  
 }  
  
 public static boolean chislo\_Li(String m) {  
 if (m == null) {  
 return false;  
 }  
 try {  
 double d = Double.*parseDouble*(m);  
 } catch (NumberFormatException nfe) {  
 return false;  
 }  
 return true;  
 }  
  
 private static boolean vivodSTR(String[][] m, int o) {  
 int i = 8;  
 while (i != m.length) {  
 for (i = 8; i < m.length; i++) {  
 if (Objects.*equals*(m[o][4], "-")) {  
 if (Math.*pow*(*parseInt*(m[o][0]), *parseInt*(m[o][3]) - *parseInt*(m[o][5])) == *parseInt*(m[o][i])) {  
 return true;  
 }  
 } else if (Objects.*equals*(m[o][4], "+")) {  
 if (Math.*pow*(*parseInt*(m[o][0]), *parseInt*(m[o][3]) + *parseInt*(m[o][5])) == *parseInt*(m[o][i])) {  
 return true;  
 }  
 }  
 }  
 return false;  
 }  
 return false;  
 }  
  
 private static boolean vivodSTOL(String[][] m, int o) {  
 int i = 8;  
 while (i != m.length) {  
 for (i = 8; i < m.length; i++) {  
 if (Objects.*equals*(m[4][o], "-")) {  
 if (Math.*pow*(*parseInt*(m[4][o]), *parseInt*(m[4][o]) - *parseInt*(m[4][o])) == *parseInt*(m[i][o])) {  
 return true;  
 }  
 } else if (Objects.*equals*(m[4][o], "+")) {  
 if (Math.*pow*(*parseInt*(m[0][o]), *parseInt*(m[3][o]) + *parseInt*(m[5][o])) == *parseInt*(m[i][o])) {  
 return true;  
 }  
 }  
 }  
 return false;  
 }  
 return false;  
 }  
  
 public static void printStr(String[][] m, int p) {  
 for (int i = 0; i < m.length; i++) {  
 System.*out*.print(m[p][i] + "\t");  
 }  
 System.*out*.println("");  
 }  
  
 public static void printStol(String[][] m, int p) {  
 for (int i = 0; i < m[0].length; i++) {  
 System.*out*.print(m[i][p] + "\t");  
 }  
 System.*out*.println("");  
 }  
  
 public static void mas\_to\_ArrSTR(String[][] m, ArrayList<String> kik, int p) {  
 StringBuilder str = new StringBuilder();  
 for (int i = 0; i < 9; i++) {  
 if (i == 8) {  
 if (Objects.*equals*(m[p][4], "-")) {  
 str.append(Double.*toString*(Math.*pow*(*parseInt*(m[p][0]), *parseInt*(m[p][3]) - *parseInt*(m[p][5]))));  
 } else if (Objects.*equals*(m[p][4], "+")) {  
 str.append(Double.*toString*(Math.*pow*(*parseInt*(m[p][0]), *parseInt*(m[p][3]) + *parseInt*(m[p][5]))));  
 }  
 } else {  
 str.append(m[p][i]);  
 }  
 }  
 kik.add(String.*valueOf*(str));  
 }  
  
 public static void masArrSTOL(String[][] m, ArrayList<String> kik, int p) {  
 StringBuilder arr= new StringBuilder();  
 for (int i = 0; i < 9; i++) {  
 if (i == 8) {  
 if (Objects.*equals*(m[4][p], "-")) {  
 kik.add(Double.*toString*(Math.*pow*(*parseInt*(m[0][p]), *parseInt*(m[3][p]) - *parseInt*(m[5][p]))));  
 } else if (Objects.*equals*(m[4][p], "+")) {  
 kik.add(Double.*toString*(Math.*pow*(*parseInt*(m[0][p]), *parseInt*(m[3][p]) + *parseInt*(m[5][p]))));  
 }  
 } else {  
 arr.append(m[i][p]);  
 }  
 }  
 }  
  
 public static void printArr(ArrayList<String> ar) {  
 for (String i:ar){  
 System.*out*.println(i);  
 }  
 }  
}

## 4.3. Тестирование работы программы







# Задание 5

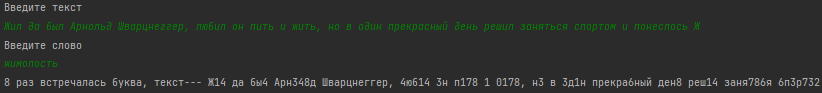
## 5.1. Постановка задачи

Дан текст. Подсчитать сколько раз в нем встречаются буквы из введенного с клавиатуры слова. Заменить буквы в тексте на их порядковый номер в веденном с клавиатуры слове.

## 5.2. Решение задачи, код программы

import java.util.Scanner;  
public class work\_5 {  
 public static void main(String[] args) {  
 Scanner mc = new Scanner(System.*in*);  
 System.*out*.println("Введите текст");  
 String text = mc.nextLine();  
 System.*out*.println("Введите слово");  
 String word = mc.nextLine();  
 int cntBUKV=0;  
 for (int i=0;i<text.length();i++){  
 for(int j=0;j<word.length();j++){  
 if (text.toCharArray()[i]==word.toCharArray()[j]){  
 cntBUKV++;  
 String m = String.*valueOf*(text.charAt(i));  
 text=text.replace(m, Integer.*toString*(j));  
 }  
 }  
 }  
 System.*out*.println(cntBUKV+" раз встречалась буква, текст--- "+text);  
 }  
}

## 5.3. Тестирование работы программы



Задание 6

## 6.1. Постановка задачи

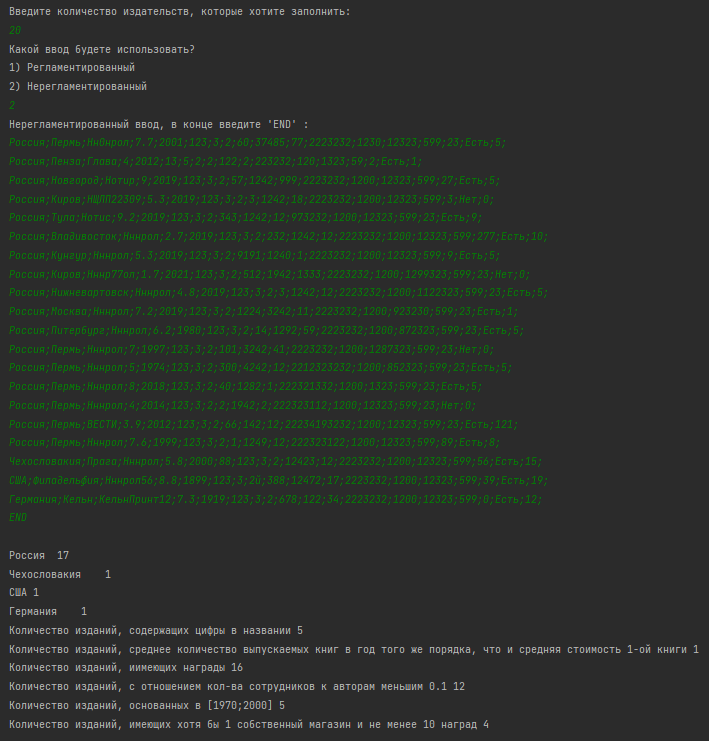
Дан массив строк, который содержит информацию об издательствах (каждая ячейка соответствует 1 издательству), не менее 20 строк. Создать две программы для регламентированного и нерегламентированного ввода данных об издательствах. Определить в каждой программе: количество издательств по странам, у которых в названии есть цифры, среднее количество выпускаемых книг в год того же прядка, что и средняя стоимость 1-ой книги, есть награды; количество издательств по городам, у которых отношение числа сотрудников к числу авторов меньше 0,1, год создания принадлежит диапазону [1970; 2000], есть хотя бы 1 собственный магазин и не менее 10 наград. - Первая программа: Сделать регламентированный ввод данных об издательствах. Каждая ячейка массива содержит информацию в определенном виде: Страна; Город; Название; Рейтинг; Год создания; Количество сотрудников; Количество сотрудников с профессиональным образованием в области перевода; Количество сотрудников с филологическим образованием; Количество авторов, с которыми сотрудничает издательство; Среднее количество книг выпускаемых в год; Количество бестселлеров; Средний доход в год; Средняя стоимость 1-ой книги; Расходы на рекламу; Наличие собственных книжных магазинов; Количество собственных магазинов; Наличие наград; Количество наград. - Вторая программа: Сделать нерегламентированный ввод данных об издательствах. Т.е. пользователи вводят информацию в произвольном виде.

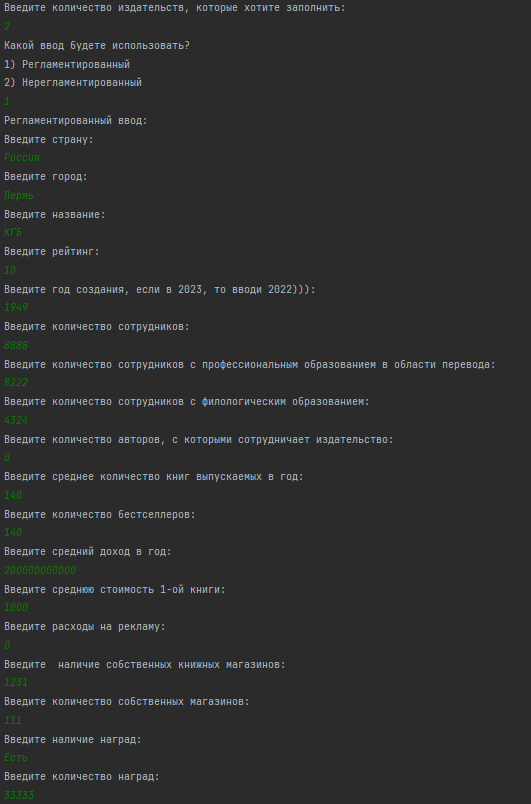
## 6.2. Решение задачи, код программы

import java.util.Objects;  
import java.util.Scanner;  
public class work\_6 {  
 public static int *count\_parametr* = 18;  
 public static void print\_text(int i) {  
 if (i == 0) {  
 System.*out*.println("Введите страну:");  
 }  
 if (i == 1) {  
 System.*out*.println("Введите город:");  
 }  
 if (i == 2) {  
 System.*out*.println("Введите название:");  
 }  
 if (i == 3) {  
 System.*out*.println("Введите рейтинг:");  
 }  
 if (i == 4) {  
 System.*out*.println("Введите год создания, если в 2023, то вводи 2022))):");  
 }  
 if (i == 5) {  
 System.*out*.println("Введите количество сотрудников:");  
 }  
 if (i == 6) {  
 System.*out*.println("Введите количество сотрудников с профессиональным образованием в области перевода:");  
 }  
 if (i == 7) {  
 System.*out*.println("Введите количество сотрудников с филологическим образованием:");  
 }  
 if (i == 8) {  
 System.*out*.println("Введите количество авторов, с которыми сотрудничает издательство:");  
 }  
 if (i == 9) {  
 System.*out*.println("Введите среднее количество книг выпускаемых в год:");  
 }  
 if (i == 10) {  
 System.*out*.println("Введите количество бестселлеров:");  
 }  
 if (i == 11) {  
 System.*out*.println("Введите средний доход в год:");  
 }  
 if (i == 12) {  
 System.*out*.println("Введите среднюю стоимость 1-ой книги:");  
 }  
 if (i == 13) {  
 System.*out*.println("Введите расходы на рекламу:");  
 }  
 if (i == 14) {  
 System.*out*.println("Введите наличие собственных книжных магазинов:");  
 }  
 if (i == 15) {  
 System.*out*.println("Введите количество собственных магазинов:");  
 }  
 if (i == 16) {  
 System.*out*.println("Введите наличие наград:");  
 }  
 if (i == 17) {  
 System.*out*.println("Введите количество наград:");  
 }  
  
 }  
 public static void reglam\_print(int str, int stol, String [][] text){ //Метод для регламентированного ввода  
 Scanner mc=new Scanner(System.*in*);  
 text[str][stol]=mc.next();  
 }  
 public static void find\_information(String[][] ma) { //Метод который ищет нужные данные об издательствах  
 String [][] mas=ma;  
 String STR = "";  
 int k = 1,cnt=0,cntNagrad=0,cntPoryadok=0,cntOsn=0,cntAvtory=0,cntMagaz=0;  
 for (int i = 0; i < mas.length; i++) {  
 for (int z=0;z<10;z++){//кол-во издательств с цифрами  
 if (mas[i][2].contains(String.*valueOf*(z))){  
 cnt++;break;  
 }  
 }  
 if (Objects.*equals*(mas[i][16], "Есть")){//nagrady  
 cntNagrad++;  
 }  
 if (Integer.*parseInt*(mas[i][4])>=1970 & Integer.*parseInt*(mas[i][4])<=2000){//god\_osn  
 cntOsn++;  
 }  
 if (Integer.*parseInt*(mas[i][15]) >= 1 & Integer.*parseInt*(mas[i][17]) >= 10){//nagrady i magazin  
 cntMagaz++;  
 }  
 if (Integer.*parseInt*(mas[i][5]) / Integer.*parseInt*(mas[i][8])>=0.1){//Otnoshenie  
 cntAvtory++;  
 }  
 String chis1=Integer.*toString*(Integer.*parseInt*(mas[i][9])/(2023-Integer.*parseInt*(mas[i][4]))),chis2=mas[i][12];  
 if(chis2.length()==chis1.length()){  
 cntPoryadok++;  
 }  
 STR = mas[i][0];  
 STR = "";  
 k = 1;  
 if (!mas[i][0].equals("")) {//cifri  
 STR = "";  
 k=1;  
 STR = mas[i][0];  
 for (int j = i + 1; j < mas.length; j++) {  
 if (STR.equals(mas[j][0])) {  
 mas[j][0] = "";  
 k++;  
 }  
 }  
 mas[i][0]="";  
 System.*out*.println(STR + "\t" + k);  
 }  
 }  
 System.*out*.println("Количество изданий, содержащих цифры в названии " + cnt);  
 System.*out*.println("Количество изданий, среднее количество выпускаемых книг в год того же порядка," +  
 " что и средняя стоимость 1-ой книги " + cntPoryadok);  
 System.*out*.println("Количество изданий, иимеющих награды " + cntNagrad);  
 System.*out*.println("Количество изданий, с отношением кол-ва сотрудников к авторам меньшим 0.1 " + cntAvtory);  
 System.*out*.println("Количество изданий, основанных в [1970;2000] " + cntOsn);  
 System.*out*.println("Количество изданий, имеющих хотя бы 1 собственный магазин и не менее 10 наград " + cntMagaz);  
  
 }  
  
 public static void main(String[] args) {  
 Scanner mc = new Scanner(System.*in*);  
 int count\_city;  
 int which\_print;  
 System.*out*.println("Введите количество издательств, которые хотите заполнить:");  
 count\_city = mc.nextInt();  
 String[][] TEXT = new String[count\_city][*count\_parametr*];  
 String[][] CONEC = new String[count\_city][10];  
 System.*out*.println("Какой ввод будете использовать?" + "\n" +  
 "1) Регламентированный" + "\n" +  
 "2) Нерегламентированный");  
 which\_print = mc.nextInt();  
 if (which\_print == 1){System.*out*.println("Регламентированный ввод:");  
 for (int i = 0;i<count\_city;i++){  
 for (int j=0;j<*count\_parametr*;j++){  
 *print\_text*(j);  
 *reglam\_print*(i,j,TEXT);  
 }

*find\_information*(TEXT);  
 }  
 }  
 if (which\_print == 2) {  
 System.*out*.println("Нерегламентированный ввод, в конце введите 'END' :");  
 Scanner pi=new Scanner(System.*in*);  
 int i=0;  
 while (pi.hasNextLine()) {  
 StringBuilder str = new StringBuilder();  
 String line = pi.nextLine();  
 if (!line.equals("END")) {  
 str.append(line).append(System.*lineSeparator*());  
 }  
 else {break;}  
 String [] mas;  
 mas = String.*valueOf*(str).split(";");  
 for (int j = 0; j < *count\_parametr*; j++) {  
 TEXT[i][j] = mas[j];  
 }  
 i++;  
 }  
 *find\_information*(TEXT);  
 }  
 else {  
 System.*out*.println("Ввод не верен!!!!!!!!!!!!!, ВВЕДИ 1 или 2) ;))");  
 }  
 }  
 public static void printmas(String[][] ms) {  
 for (int i = 0; i < ms.length; i++) {  
 for (int j = 0; j < ms[0].length; j++) {  
 System.*out*.print(ms[i][j] + "\t");  
 }  
 System.*out*.println();  
 }  
 }  
}

## 6.3. Тестирование работы программы с проверкой





# Задание 7

## 7.1. Постановка задачи

Дан текст. Поменять порядок слов в предложениях на обратный, если предложение начинается и заканчивается одинаковыми буквами российского алфавита.

## 7.2. Решение задачи, код программы

import java.util.Scanner;  
  
public class work\_7 {  
 public static void main(String[] args) {  
 Scanner mc = new Scanner(System.*in*);  
 System.*out*.println("Введи текст");  
 String text = mc.nextLine();  
 String[] array = text.split(" ");  
 String zxxz = "";  
 if (text.charAt(0)==text.charAt(text.length()-1)){  
 for (int i = array.length - 1; i >= 0; i--) {  
 zxxz += array[i] + " ";  
 }  
 System.*out*.println("Перевернул " + zxxz);  
 }  
 else System.*out*.println("Не перевернул " + zxxz);  
 }  
}

## 7.3. Тестирование работы программы



# Задание 8

## 8.1. Постановка задачи

Дан текст, который нужно расшифровать по следующему принципу: если в слове есть цифра 1 то заменить ее на «один»; если в слове есть цифра 2 то заменить ее на «втор»; если в слове есть цифра 3 то заменить ее на «три». Выполнить расшифровку текста.

## 8.2. Решение задачи, код программы

import java.util.Scanner;  
public class work\_8 {  
 public static void main(String[] args) {  
 Scanner mc = new Scanner(System.*in*);  
 System.*out*.println("Введи текст");  
 String text = mc.nextLine();  
 String q="1",a="один",w="2",s="втор",e="3",d="три";  
 String m;  
 m=text.replace(q,a).replace(w,s).replace(e,d);  
 System.*out*.println(m);  
 }  
}

## 8.3. Тестирование работы программы



Задание 9

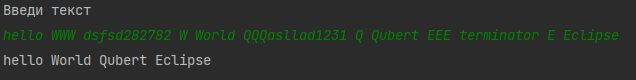
## 9.1. Постановка задачи

Дан текст с комментариями. Из текста удалить все комментарии (WWW …W, EEE....E, QQQ...Q), если комментарий нет, то сообщить об этом.

## 9.2. Решение задачи, код программы

import java.util.Scanner;  
import java.lang.String;  
  
public class work\_9 {  
 public static void main(String[] args) {  
 Scanner mc = new Scanner(System.*in*);  
 System.*out*.println("Введи текст");  
 String komm = mc.nextLine();  
 System.*out*.println(komm.replaceAll("\\s?WWW\\s?(\\w\*)\\s?W","")  
 .replaceAll("\\s?QQQ\\s?(\\w\*)\\s?(Q)","")  
 .replaceAll("\\s?EEE\\s?(\\w\*)\\s?(E)",""));  
 }  
}

## 9.3. Тестирование работы программы с проверкой



Задание 10

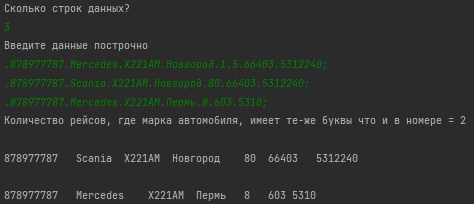
## 10.1. Постановка задачи

Дан текст, который содержит данные определенной структуры, разделителем данных об одном объекте является «;», разделитель между объектами «.».«Рейс»: марка автомобиля; номер автомобиля; пункт назначения; грузоподъемность (в тоннах); стоимость единицы груза; общая стоимость груза. Разбить данные по объектам на массив строк. Посчитать количество автомобилей, у которых в номере содержится хотя бы одна из букв марки. Вывести данные про автомобили, грузоподъемность которых больше 2 тонн.

## 10.2. Решение задачи, код программы

import java.util.Arrays;  
import java.util.Scanner;  
import java.lang.String;  
  
import static java.lang.System.*out*;  
  
public class work\_10 {  
 public static <mas1> void main(String[] args) {  
 Scanner mc = new Scanner(System.*in*);  
 System.*out*.println("Сколько строк данных?");  
 int n = mc.nextInt();  
 String [] mas1 = new String [n];  
 String [][] mas2 = new String [n][7];  
 System.*out*.println("Введите данные построчно");  
 for (int i =0;i<n;i++){  
 mas1[i]=mc.next();  
 }  
 for(int i = 0;i<n;i++){  
 for (int j=0;j<7;j++){  
 if (j==0){  
 String p = String.*join* ("", mas1[i]);  
 mas2[i][j]= p.substring(p.indexOf(".")+1,p.indexOf(".",1));  
 StringBuilder g = new StringBuilder(mas1[i]).delete(0,mas1[i].indexOf(".",1));  
 mas1[i]= String.*valueOf*(g);  
 }  
 else {  
 String p = String.*join*("", mas1[i]);  
 if (j == 6) {  
 mas2[i][j] = p.substring(p.indexOf(".")+1, p.indexOf(";"));  
 }  
 else {mas2[i][j] = p.substring(p.indexOf(".")+1, p.indexOf(".", 1));  
 StringBuilder g = new StringBuilder(mas1[i]).delete(0,mas1[i].indexOf(".",1));  
 mas1[i]=String.*valueOf*(g);}  
 }  
 }  
 }  
 int cntPov=0;  
 for (int k=0;k<n;){  
 for (int i = 0; i<mas2[k][1].length();i++){//авто имеющие те-же буквы что и в номере  
 for(int j=0;j<mas2[k][2].length();j++){  
 String marka = mas2[k][1].toString();  
 String numb = mas2[k][2].toString();  
 if(marka.charAt(i)==numb.charAt(j)){  
 cntPov++;break;  
 }  
 }  
 }  
 k++;  
 }  
 if(cntPov!=0){  
 *out*.println("Количество рейсов, где марка автомобиля, имеет те-же буквы что и в номере = "+cntPov);  
 }  
 else *out*.println("Нет таких рейсов");;  
 for(int i = 0;i<n;i++){  
 int op=Integer.*parseInt*(mas2[i][4]);  
 if(op>2){  
 *out*.println();*printmas2*(mas2,i);  
 }  
 }  
 }  
 public static void printmas(String[] array) {  
 for (int i=0; i<=array.length-1; i++) {  
 for (int j=0; j<7; j++) {  
 *out*.print(array[i]+"\t");  
 }  
 *out*.println();  
 }  
 }  
 public static void printmas2(String[][] array,int i) {  
 for (int j=0; j<7; j++) {  
 *out*.print(array[i][j]+"\t");  
 }  
 *out*.println();  
 }  
}

## 10.3. Тестирование работы программы с проверкой



Задание 11

## 11.1. 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.

## 11.2. Решение задачи, код программы

import java.util.Arrays;  
import java.util.regex.Matcher;  
import java.util.regex.Pattern;  
  
public class work\_11 {  
 public static void main(String[] args) {  
 String strings = "привет всем, рад вас видеть уважаемые коллеги."  
 ,strings1=strings,strings2=strings,strings3=strings;  
 String [] pr={"всем", "Всем", "Коллеги", "коллеги"};  
 String prZam="ребятки",delUv="\\s?(у\\D{8})\\s?";  
 Pattern patt;  
 Matcher matt;  
 for (int i = 0;i < pr.length;i++){  
 patt=Pattern.*compile*(pr[i]);  
 matt=patt.matcher(strings1);  
 if(matt.find()){  
 strings1=matt.replaceAll(prZam);  
 }  
 }  
 System.*out*.println("1) Замена --- "+strings1);  
 System.*out*.println("2) Все заглавными --- "+strings2.toUpperCase());  
 System.*out*.println("3) Замена регулярными --- "+strings3.replaceAll(delUv," любимые "));  
 }  
}

## 11.3. Тестирование работы программы с проверкой



# Задание 12

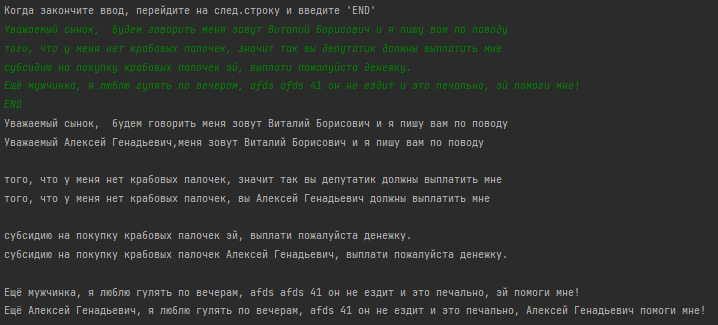
## 12.1. Постановка задачи

Дан текст письма дедушки к депутату по предложениям в виде массива строк, в котором встречаются несколько ошибок: 1. Повторение слов через один или несколько пробелов; 2. Есть слова паразиты «будем говорить», «ну», «значит так», «короче», «типа»; 3. Обращение к депутату «Алексей Геннадьевич» в тексте записано, как «эй», «депутатик», «мужчинка» или «сынок». Помогите дедушке не опозориться и исправить его стандартные ошибки. В цикле с использование регулярных выражений выполнить автозамену информации с использованием массивов регулярных выражений. Исходную и измененную информацию вывести построчно: сначала исходная строка, затем измененная.

## 12.2. Решение задачи, код программы

import java.util.Objects;  
import java.util.Scanner;  
import java.util.regex.Matcher;  
import java.util.regex.Pattern;  
  
public class work\_12 {  
 public static void main(String[] args) {  
 Scanner mc =new Scanner(System.*in*);  
 System.*out*.println("Когда закончите ввод, перейдите на след..строку и введите 'END'");  
 while (mc.hasNextLine()) {  
 StringBuilder str = new StringBuilder();  
 String line = mc.nextLine();  
 if (!line.equals("END")) {  
 str.append(line).append(System.*lineSeparator*());  
 System.*out*.println(String.*valueOf*(str).replaceAll("(\\r|\\n)", ""));  
 }  
 else{break;}  
 String m = String.*valueOf*(str);  
 String [] mas={"эй","депутатик","мужчинка", "сынок","эй ","депутатик ","мужчинка ", "сынок ",  
 "ну ","будем говорить ","значит так ","короче ","типа ","будем говорить","значит так",  
 "короче","типа","\\s+\\s+"};  
 for(int i =0;i<mas.length;i++){  
 Pattern q1=Pattern.*compile*(*numb*(mas,i));  
 Matcher textDeda1 = q1.matcher(m);  
 while (textDeda1.find()) {  
 if(i <= 7) {  
 m = textDeda1.replaceAll("Алексей Генадьевич");}  
 else if (i > 7 || i <= 16) {  
 m= textDeda1.replaceAll("");  
 } else if (i==17){m= textDeda1.replaceAll(" ");}  
 }  
 }  
 String[] pp = m.split(" ");  
 for (int j = 0;j < pp.length-2;j++){  
 if (Objects.*equals*(pp[j], pp[j + 1])){  
 pp[j+1] ="";  
 }  
 else if (Objects.*equals*(pp[j], pp[j + 2])){  
 pp[j+2]="";  
 }  
 }  
 m= String.*join*(" ",pp);  
 Pattern q1=Pattern.*compile*(*numb*(mas,17));  
 Matcher textDeda1 = q1.matcher(m);  
 m= textDeda1.replaceAll(" ");  
 System.*out*.println(m);  
 System.*out*.println("");  
 }  
 }  
 public static String numb(String [] m,int n){  
 return m[n];  
 }  
}

## 12.3. Тестирование работы программы с проверкой



Как мы можем видеть, программа убрала все ошибки дедушки. Код верен.