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»

Выполнил: студент группу ИСТ-22-1б Зверев А.А.

Проверил: Нетбай Георгий Владимирович

Пермь, 2023

**Содержание**

[Задание 1 4](#_Toc136047746)

[1.1. Постановка задачи 4](#_Toc136047747)

[1.2. Решение задачи, код программы 4](#_Toc136047748)

[1.3. Тестирование работы программы 4](#_Toc136047749)

[Задание 2 5](#_Toc136047750)

[2.1. Постановка задачи 5](#_Toc136047751)

[2.2. Решение задачи, код программы 5](#_Toc136047752)

[2.3. Тестирование работы программы 5](#_Toc136047753)

[Задание 3 6](#_Toc136047754)

[3.1. Постановка задачи 6](#_Toc136047755)

[3.2. Решение задачи, код программы 6](#_Toc136047756)

[3.3. Тестирование работы программы 6](#_Toc136047757)

[Задание 4 6](#_Toc136047758)

[4.1. Постановка задачи 6](#_Toc136047759)

[4.2. Решение задачи, код программы 6](#_Toc136047760)

[4.3. Тестирование работы программы 6](#_Toc136047761)

[Задание 5 6](#_Toc136047762)

[5.1. Постановка задачи 6](#_Toc136047763)

[5.2. Решение задачи, код программы 6](#_Toc136047764)

[5.3. Тестирование работы программы 6](#_Toc136047765)

[Задание 6 6](#_Toc136047766)

[6.1. Постановка задачи 6](#_Toc136047767)

[6.2. Решение задачи, код программы 6](#_Toc136047768)

[6.3. Тестирование работы программы 6](#_Toc136047769)

[Задание 7 7](#_Toc136047770)

[7.1. Постановка задачи 7](#_Toc136047771)

[7.2. Решение задачи, код программы 7](#_Toc136047772)

[7.3. Тестирование работы программы 7](#_Toc136047773)

[Задание 8 7](#_Toc136047774)

[8.1. Постановка задачи 7](#_Toc136047775)

[8.2. Решение задачи, код программы 7](#_Toc136047776)

[8.3. Тестирование работы программы 7](#_Toc136047777)

[Задание 9 7](#_Toc136047778)

[9.1. Постановка задачи 7](#_Toc136047779)

[9.2. Решение задачи, код программы 7](#_Toc136047780)

[9.3. Тестирование работы программы 7](#_Toc136047781)

[Задание 10 7](#_Toc136047782)

[10.1. Постановка задачи 7](#_Toc136047783)

[10.2. Решение задачи, код программы 7](#_Toc136047784)

[10.3. Тестирование работы программы 8](#_Toc136047785)

[Задание 11 8](#_Toc136047786)

[11.1. Постановка задачи 8](#_Toc136047787)

[11.2. Решение задачи, код программы 8](#_Toc136047788)

[11.3. Тестирование работы программы 8](#_Toc136047789)

[Задание 12 8](#_Toc136047790)

[12.1. Постановка задачи 8](#_Toc136047791)

[12.2. Решение задачи, код программы 8](#_Toc136047792)

[12.3. Тестирование работы программы 8](#_Toc136047793)

# Задание 1

## 1.1. Постановка задачи

Дан массив символов. Определить, сколько в нем символов «:», «;».

## 1.2. Решение задачи, код программы

public static char[] InputArrOfChar(int n) {  
 Scanner in = new Scanner(System.*in*);  
 char[] array = new char[n];  
 System.*out*.println("Введите элементы массива:");  
 for (int i = 0; i < n; i++) {  
 System.*out*.print("Элемент " + (i + 1) + ": ");  
 array[i] = in.next().charAt(0);  
 }  
 return array;  
}

public static void num1(int n) {  
 char[] syms = *InputArrOfChar*(n);  
 int dd = 0;  
 int dnd = 0;  
 for (char sym : syms) {  
 if (sym == ':') {  
 dd++;  
 } else if (sym == ';') {  
 dnd++;  
 }  
 }  
 System.*out*.println("Количество символов ':' в массиве: " + dd);  
 System.*out*.println("Количество символов ';' в массиве: " + dnd);  
}

## 1.3. Тестирование работы программы

Таблица 1

Тестирование работы программы.

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

# Задание 2

## 2.1. Постановка задачи

Дан массив символов, содержащий число в двоичной системе счисления. Проверить правильность ввода этого числа (в его записи должны быть только символы 0 и 1). 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 7-ую систему счисления, записать число в новый массив символов.

## 2.2. Решение задачи, код программы

public static boolean isbinstr(String numb) {  
 for (char d : numb.toCharArray()) {  
 if (d != '0' && d != '1') {  
 return false;  
 }  
 }  
 return true;  
}  
public static void num2() {  
 Scanner in = new Scanner(System.*in*);  
 String binstr;  
 while (true) {  
 System.*out*.print("Введите число в двоичной системе счисления: ");  
 binstr = in.nextLine();  
 if (*isbinstr*(binstr)) {  
 break;  
 } else {  
 System.*out*.println("Неверный ввод! Число должно содержать только символы 0 и 1.");  
 }  
 }  
  
 int decNum = Integer.*parseInt*(binstr, 2);  
 String sepNum = Integer.*toString*(decNum, 7);  
  
 char[] sepArr = sepNum.toCharArray();  
  
 System.*out*.println("Число в семеричной системе счисления: " + sepNum);  
 System.*out*.println("Массив символов семеричного числа:");  
 for (char d : sepArr) {  
 System.*out*.print(d + " ");  
 }  
}

## 2.3. Тестирование работы программы

Таблица 2

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Java | Проверка |
| 1 |  |  |
| 2 |  |  |

# Задание 3

## 3.1. Постановка задачи

Дан массив символов. Проанализировать массив и ответить на вопросы:

а) Определить количество латинских прописных букв.

б) Выяснить, входит ли в последовательность буква, введенная с клавиатуры, например «*ю»*, регистр не учитывать*.*

в) Выяснить, верно ли, что среди символов имеются две и более запятые*.*

г) Выяснить, имеется ли среди символов пара соседствующих букв «*во»* или «*ов».*

д) Выяснить, имеется ли среди символов пара соседствующих одинаковых цифр.

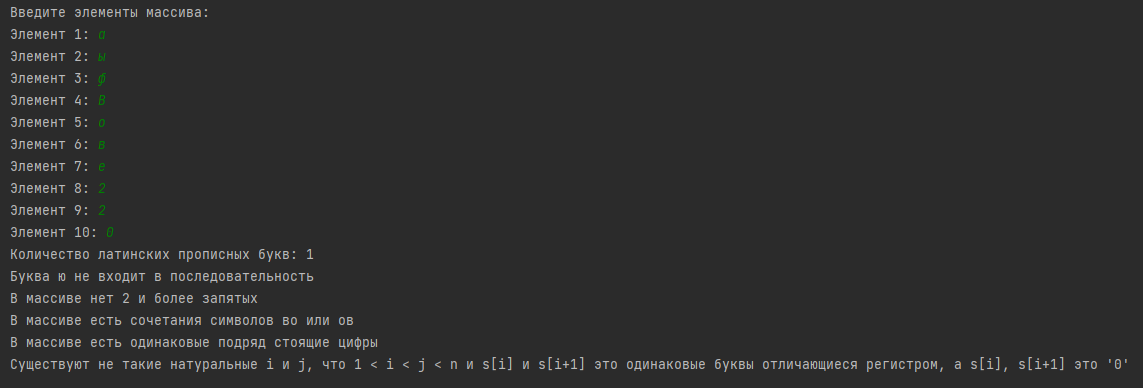
е) Выяснить, верно ли. что существуют такие натуральные *i* и *j,* что *1 <i <j < п* и что *si*, *si+1*это одинаковые буквы отличающиеся регистром, a *sj,* *sj+1* это 0.

## 3.2. Решение задачи, код программы

public static int num3\_a(char[] arr) {  
 int Counter = 0;  
 for (char c : arr) {  
 if (Character.*isUpperCase*(c)) {  
 Counter++;  
 }  
 }  
 return Counter;  
}  
public static boolean num3\_b(char[] arr) {  
 boolean check = false;  
 for (char c : arr) {  
 if (Character.*toLowerCase*(c) == 'ю') {  
 check = true;  
 break;  
 }  
 }  
 return check;  
  
}  
public static boolean num3\_c(char[] arr) {  
 boolean check = false;  
 int Count = 0;  
 for (char c : arr) {  
 if (c == ',') {  
 Count++;  
 if (Count >= 2) {  
 check = true;  
 break;  
 }  
 }  
 }  
 return check;  
}  
public static boolean num3\_d(char[] arr) {  
 boolean check = false;  
 for (int i = 0; i < arr.length - 1; i++) {  
 if ((Character.*toLowerCase*(arr[i]) == 'в' && Character.*toLowerCase*(arr[i + 1]) == 'о') ||  
 (Character.*toLowerCase*(arr[i]) == 'о' && Character.*toLowerCase*(arr[i + 1]) == 'в')) {  
 check = true;  
 break;  
 }  
 }  
 return check;  
}  
public static boolean num3\_e(char[] arr) {  
 boolean check = false;  
 for (int i = 0; i < arr.length - 1; i++) {  
 if (Character.*isDigit*(arr[i]) && Character.*isDigit*(arr[i + 1]) && arr[i] == arr[i + 1]) {  
 check = true;  
 break;  
 }  
 }  
 return check;  
}  
public static boolean num3\_f(char[] arr) {  
 boolean check = false;  
 for (int i = 1; i < arr.length - 2; i++) {  
 if (Character.*isLetter*(arr[i]) && Character.*isLetter*(arr[i + 1]) && Character.*toLowerCase*(arr[i]) == Character.*toLowerCase*(arr[i + 1]) && arr[i] != arr[i + 1] && arr[i] == '0' && arr[i + 1] == '0') {  
 check = true;  
 break;  
 }  
 }  
 return check;  
}  
public static void main(String[] args){  
 Scanner in = new Scanner(System.*in*);  
 int n;  
 String text;  
System.*out*.println("Input length of array of char");  
n = in.nextInt();  
char[] arr = *InputArrOfChar*(n);  
System.*out*.println("Количество латинских прописных букв: " + *num3\_a*(arr));  
System.*out*.println("Буква ю " + (*num3\_b*(arr) ? "входит" : "не входит") + " в последовательность");  
System.*out*.println("В массиве " + (*num3\_c*(arr) ? "есть" : "нет") + " 2 и более запятых");  
System.*out*.println("В массиве " + (*num3\_d*(arr) ? "есть" : "нет") + " сочетания символов во или ов");  
System.*out*.println("В массиве " + (*num3\_e*(arr) ? "есть" : "нет") + " одинаковые подряд стоящие цифры");  
System.*out*.println("Существуют " + (*num3\_f*(arr) ? "" : "не ") + "такие натуральные i и j, что 1 < i < j < n и s[i] и s[i+1] это одинаковые буквы отличающиеся регистром, а s[i], s[i+1] это '0'");

}

## 3.3. Тестирование работы программы



# Задание 4

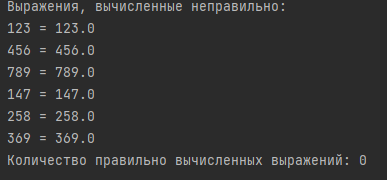
## 4.1. Постановка задачи

Задана матрица символов N×N. Определить построчно и по столбцам, есть ли в них выражения вида ln«число» \* (4\*«число» - «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

public static boolean isVal\_num4(String expr) {  
 return expr.matches("(\\d+\\\*?)+");  
}  
public static double unknownepr\_num4(String expr){  
 String[] numb = expr.split("\\\*");  
 double result = Double.*parseDouble*(1numb[0]);  
 for (int i = 1; i < numb.length; i++) {  
 result \*= Double.*parseDouble*(numb[i]);  
 }  
 return result;  
}  
public static void num4(){  
 //for example  
 char[][] matrix = {  
 {'1', '2', '3'},  
 {'4', '5', '6'},  
 {'7', '8', '9'}  
 };  
 ArrayList<String> incor = new ArrayList<>();  
 int cor = 0;  
 for (int r = 0; r < matrix.length; r++) {  
 String expr = "";  
 for (int col = 0; col < matrix[r].length; col++) {  
 if (Character.*isDigit*(matrix[r][col])) {  
 expr += matrix[r][col];  
 } else {  
 if (*isVal\_num4*(expr)) {  
 double result = *unknownepr\_num4*(expr);  
 if (result == Double.*parseDouble*(String.*valueOf*(matrix[r][col]))) {  
 cor++;  
 } else {  
 incor.add(expr + " = " + result);  
 }  
 }  
 expr = "";  
 }  
 }  
 if (*isVal\_num4*(expr)) {  
 double result = *unknownepr\_num4*(expr);  
 if (result == Double.*parseDouble*(String.*valueOf*(matrix[r][matrix[r].length - 1]))) {  
 cor++;  
 } else {  
 incor.add(expr + " = " + result);  
 }  
 }  
 }  
 for (int col = 0; col < matrix[0].length; col++) {  
 String expr = "";  
 for (int r = 0; r < matrix.length; r++) {  
 if (Character.*isDigit*(matrix[r][col])) {  
 expr += matrix[r][col];  
 } else {  
 if (*isVal\_num4*(expr)) {  
 double result = *unknownepr\_num4*(expr);  
 if (result == Double.*parseDouble*(String.*valueOf*(matrix[r][col]))) {  
 cor++;  
 } else {  
 incor.add(expr + " = " + result);  
 }  
 }  
 expr = "";  
 }  
 }  
 if (*isVal\_num4*(expr)) {  
 double result = *unknownepr\_num4*(expr);  
 if (result == Double.*parseDouble*(String.*valueOf*(matrix[matrix.length - 1][col]))) {  
 cor++;  
 } else {  
 incor.add(expr + " = " + result);  
 }  
 }  
 }  
 System.*out*.println("Выражения, вычисленные неправильно:");  
 for (String expr : incor) {  
 System.*out*.println(expr);  
 }  
 System.*out*.println("Количество правильно вычисленных выражений: " + cor);  
}

## 4.3. Тестирование работы программы



# Задание 5

## 5.1. Постановка задачи

Дана строка. Изменить регистр на верхний у тех слов, которые содержат индекс введенный с клавиатуры 2 и более раз.

## 5.2. Решение задачи, код программы

public static String num5(String input, int index){  
 Map<String, Integer> wordCount = new HashMap<>();  
 String[] words = input.split(" ");  
 for (String word : words) {  
 if (wordCount.containsKey(word)) {  
 wordCount.put(word, wordCount.get(word) + 1);  
 } else {  
 wordCount.put(word, 1);  
 }  
 }  
 StringBuilder result = new StringBuilder();  
 for (String word : words) {  
 if (wordCount.get(word) >= index) {  
 result.append(word.toUpperCase()).append(" ");  
 } else {  
 result.append(word).append(" ");  
 }  
 }  
 String output = result.toString().trim();  
 return output;  
 }

## 5.3. Тестирование работы программы

Таблица 3

Тестирование работы программы и проверка результатов решения

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

# 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 городам, у которых рейтинг больше 2,5, есть не менее 5 бестселлеров, Количество авторов в 3 и более раза превышает количество сотрудников; количество издательств по странам, у которых 20 % сотрудников имеют проф. образование в области перевода и не менее 30 % сотрудников имеют филологическое образование, а расходы на рекламу составляют менее 20 % от среднего дохода.

1. - 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.
2. - Вторая программа: Сделать нерегламентированный ввод данных об издательствах. Т.е. пользователи вводят информацию в произвольном виде.

## 6.2. Решение задачи, код программы

public static void num6\_1(){  
 int count = 20;  
 String[][] publishers = new String[count][18];  
 for (int i = 0; i < count; i++) {  
 System.*out*.println("Введите информацию об издательстве " + (i + 1) + ":");  
  
 System.*out*.print("Страна: ");  
 publishers[i][0] = *StrIn*();  
  
 System.*out*.print("Город: ");  
 publishers[i][1] = *StrIn*();  
  
 System.*out*.print("Название: ");  
 publishers[i][2] = *StrIn*();  
  
 System.*out*.print("Рейтинг: ");  
 publishers[i][3] = *StrIn*();  
  
 System.*out*.print("Год создания: ");  
 publishers[i][4] = *StrIn*();  
  
 System.*out*.print("Количество сотрудников: ");  
 publishers[i][5] = *StrIn*();  
  
 System.*out*.print("Количество сотрудников с профессиональным образованием в области перевода: ");  
 publishers[i][6] = *StrIn*();  
  
 System.*out*.print("Количество сотрудников с филологическим образованием: ");  
 publishers[i][7] = *StrIn*();  
  
 System.*out*.print("Количество авторов, с которыми сотрудничает издательство: ");  
 publishers[i][8] = *StrIn*();  
  
 System.*out*.print("Среднее количество книг, выпускаемых в год: ");  
 publishers[i][9] = *StrIn*();  
  
 System.*out*.print("Количество бестселлеров: ");  
 publishers[i][10] = *StrIn*();  
  
 System.*out*.print("Средний доход в год: ");  
 publishers[i][11] = *StrIn*();  
  
 System.*out*.print("Средняя стоимость 1-ой книги: ");  
 publishers[i][12] = *StrIn*();  
  
 System.*out*.print("Расходы на рекламу: ");  
 publishers[i][13] = *StrIn*();  
  
 System.*out*.print("Наличие собственных книжных магазинов (да/нет): ");  
 publishers[i][14] = *StrIn*();  
  
 System.*out*.print("Количество собственных магазинов: ");  
 publishers[i][15] = *StrIn*();  
  
 System.*out*.print("Наличие наград (да/нет): ");  
 publishers[i][16] = *StrIn*();  
  
 System.*out*.print("Количество наград: ");  
 publishers[i][17] = *StrIn*();  
  
 System.*out*.println();  
 }  
  
 int countHighRate = 0;  
 int countBestseilers = 0;  
 int countAthrs = 0;  
 int countTrsl = 0;  
 int countPhilosophy = 0;  
 int countAdv = 0;  
  
 for (int i = 0; i < count; i++) {  
 String city = publishers[i][1];  
 double rating = Double.*parseDouble*(publishers[i][3]);  
 int bestsellers = Integer.*parseInt*(publishers[i][10]);  
 int staff = Integer.*parseInt*(publishers[i][5]);  
 int authors = Integer.*parseInt*(publishers[i][8]);  
 double translationEducationPercentage = (Double.*parseDouble*(publishers[i][6]) / staff) \* 100;  
 double philologyEducationPercentage = (Double.*parseDouble*(publishers[i][7]) / staff) \* 100;  
 double advPercentage = (Double.*parseDouble*(publishers[i][13]) / Double.*parseDouble*(publishers[i][11])) \* 100;  
  
 if (rating > 2.5) {  
 countHighRate++;  
 }  
 if (bestsellers >= 5) {  
 countBestseilers++;  
 }  
 if (authors > 3 \* staff) {  
 countAthrs++;  
 }  
 if (translationEducationPercentage >= 20) {  
 countTrsl++;  
 }  
 if (philologyEducationPercentage >= 30) {  
 countPhilosophy++;  
 }  
 if (advPercentage < 20) {  
 countAdv++;  
 }  
 }  
 System.*out*.println("Количество издательств по городам с рейтингом больше 2.5: " + countHighRate);  
 System.*out*.println("Количество издательств по городам с не менее 5 бестселлерами: " + countBestseilers);  
 System.*out*.println("Количество издательств, у которых количество авторов в 3 и более раза превышает количество сотрудников: " + countAthrs);  
 System.*out*.println("Количество издательств по странам, у которых 20% сотрудников имеют профессиональное образование в области перевода: " + countTrsl);  
 System.*out*.println("Количество издательств по странам, у которых не менее 30% сотрудников имеют филологическое образование: " + countPhilosophy);  
 System.*out*.println("Количество издательств по странам, у которых расходы на рекламу составляют менее 20% от среднего дохода: " + countAdv);  
 }  
 public static void num6\_2(){  
 List<String[]> publishers = new ArrayList<>();  
 boolean continueInput = true;  
 while (continueInput) {  
 System.*out*.println("Введите информацию об издательстве (или введите 'q' для выхода):");  
  
 System.*out*.print("Страна: ");  
 String country = *StrIn*();  
  
 if (country.equalsIgnoreCase("q")) {  
 break;  
 }  
  
 System.*out*.print("Город: ");  
 String city = *StrIn*();  
  
 System.*out*.print("Название: ");  
 String name = *StrIn*();  
  
 System.*out*.print("Рейтинг: ");  
 String rating = *StrIn*();  
  
 System.*out*.print("Год создания: ");  
 String year = *StrIn*();  
  
 System.*out*.print("Количество сотрудников: ");  
 String countStuff = *StrIn*();  
  
 System.*out*.print("Количество сотрудников с профессиональным образованием в области перевода: ");  
 String countTrsl = *StrIn*();  
  
 System.*out*.print("Количество сотрудников с филологическим образованием: ");  
 String countPsihology = *StrIn*();  
  
 System.*out*.print("Количество авторов, с которыми сотрудничает издательство: ");  
 String countAuther = *StrIn*();  
  
 System.*out*.print("Среднее количество книг, выпускаемых в год: ");  
 String countbook = *StrIn*();  
  
 System.*out*.print("Количество бестселлеров: ");  
 String countBestsailer = *StrIn*();  
  
 System.*out*.print("Средний доход в год: ");  
 String annualIncome = *StrIn*();  
  
 System.*out*.print("Средняя стоимость 1-ой книги: ");  
 String bookPrice = *StrIn*();  
  
 System.*out*.print("Расходы на рекламу: ");  
 String adv = *StrIn*();  
  
 System.*out*.print("Наличие собственных книжных магазинов (да/нет): ");  
 String hasBookstores = *StrIn*();  
  
 System.*out*.print("Количество собственных магазинов: ");  
 String countBookStores = *StrIn*();  
  
 System.*out*.print("Наличие наград (да/нет): ");  
 String hasAwards = *StrIn*();  
  
 System.*out*.print("Количество наград: ");  
 String countAwards = *StrIn*();  
  
 String[] publisherData = {country, city, name, rating, year, countStuff, countTrsl, countPsihology, countAuther, countbook, countBestsailer, annualIncome, bookPrice, adv, hasBookstores, countBookStores, hasAwards, countAwards};  
 publishers.add(publisherData);  
  
 System.*out*.print("Продолжить ввод данных? (да/нет): ");  
 String continueInputStr = *StrIn*();  
  
 continueInput = continueInputStr.equalsIgnoreCase("да");  
  
 System.*out*.println();  
 }  
  
 int countRate = 0;  
 int countBestsailers = 0;  
 int countAuthorsOverStaff = 0;  
 int countTrslations = 0;  
 int countPsyho = 0;  
 int countLow = 0;  
  
 for (String[] publisherData : publishers) {  
 String city = publisherData[1];  
 double rating = Double.*parseDouble*(publisherData[3]);  
 int bestsellers = Integer.*parseInt*(publisherData[10]);  
 int staff = Integer.*parseInt*(publisherData[5]);  
 int authors = Integer.*parseInt*(publisherData[8]);  
 double translationEducationPercentage = (Double.*parseDouble*(publisherData[6]) / staff) \* 100;  
 double philologyEducationPercentage = (Double.*parseDouble*(publisherData[7]) / staff) \* 100;  
 double advPercentage = (Double.*parseDouble*(publisherData[13]) / Double.*parseDouble*(publisherData[11])) \* 100;  
  
 if (rating > 2.5) {  
 countRate++;  
 }  
  
 if (bestsellers >= 5) {  
 countBestsailers++;  
 }  
  
 if (authors > 3 \* staff) {  
 countAuthorsOverStaff++;  
 }  
  
 if (translationEducationPercentage >= 20) {  
 countTrslations++;  
 }  
  
 if (philologyEducationPercentage >= 30) {  
 countPsyho++;  
 }  
  
 if (advPercentage < 20) {  
 countLow++;  
 }  
 }  
  
 System.*out*.println("Количество издательств по городам с рейтингом больше 2.5: " + countRate);  
 System.*out*.println("Количество издательств по городам с не менее 5 бестселлерами: " + countBestsailers);  
 System.*out*.println("Количество издательств, у которых количество авторов в 3 и более раза превышает количество сотрудников: " + countAuthorsOverStaff);  
 System.*out*.println("Количество издательств по странам, у которых 20% сотрудников имеют профессиональное образование в области перевода: " + countTrslations);  
 System.*out*.println("Количество издательств по странам, у которых не менее 30% сотрудников имеют филологическое образование: " + countPsyho);  
 System.*out*.println("Количество издательств по странам, у которых расходы на рекламу составляют менее 20% от среднего дохода: " + countLow);  
}

## 6.3. Тестирование работы программы

Ввод и конечный вывод работают отлично. Так как долго проводить полные тесты, и кол-во скриншотов будет более 3, прошу в случае не уверенности проверить со своими значениями.

# Задание 7

## 7.1. Постановка задачи

Заданы две строки одинаковой длины. Построить новую строку, состоящую из символов как одной, так и другой строки, чередующихся между собой.

## 7.2. Решение задачи, код программы

public static String num7(String a, String b) {  
 StringBuilder result = new StringBuilder();  
 int length = a.length();  
 for (int i = 0; i < length; i++) {  
 result.append(a.charAt(i));  
 result.append(b.charAt(i));  
 }  
 return result.toString();  
}

public static String StrIn(){  
 Scanner in = new Scanner(System.*in*);  
 return (in.nextLine());  
}

public static void main(String[] args){  
 Scanner in = new Scanner(System.*in*);  
 String text;

System.*out*.println("Input 1 line: ");  
String a = *StrIn*();  
System.*out*.println("Input 2 line: ");  
String b = *StrIn*();  
if (a.length()==b.length()) {  
 System.*out*.println("Результат: " + *num7*(a, b));  
}  
else {System.*out*.println("ОДИНАКОВЫЕ ПО ДЛИНЕ");}

}

## 7.3. Тестирование работы программы

Таблица 4

Тестирование работы программы и проверка результатов решения

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

# Задание 8

## 8.1. 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.

## 8.2. Решение задачи, код программы

public static String num8(String text){  
 StringBuilder compressedText = new StringBuilder();  
 int count = 1;  
  
 for (int i = 0; i < text.length() - 1; i++) {  
 char fstchar = text.charAt(i);  
 char nextChar = text.charAt(i + 1);  
  
 if (fstchar == nextChar) {  
 count++;  
 } else {  
 compressedText.append(fstchar);  
 if (count > 1) {  
 compressedText.append(count);  
 }  
 count = 1;  
 }  
 }  
 compressedText.append(text.charAt(text.length() - 1));  
 if (count > 1) {  
 compressedText.append(count);  
 }  
  
 return compressedText.toString();  
}

## 8.3. Тестирование работы программы

Таблица 5

Тестирование работы программы и проверка результатов решения

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

# Задание 9

## 9.1. Постановка задачи

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

## 9.2. Решение задачи, код программы

public static String StrIn(){  
 Scanner in = new Scanner(System.*in*);  
 return (in.nextLine());  
}

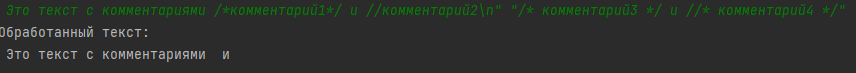
public static String num9(String text){  
 StringBuilder newText = new StringBuilder();  
 boolean insideComment = false;  
 for (int i = 0; i < text.length(); i++) {  
 if (i < text.length() - 1 && text.charAt(i) == '/' && text.charAt(i + 1) == '\*') {  
 insideComment = true;  
 i++;   
 } else if (i < text.length() - 1 && text.charAt(i) == '\*' && text.charAt(i + 1) == '/') {  
 insideComment = false;  
 i++;   
 } else if (i < text.length() - 1 && text.charAt(i) == '/' && text.charAt(i + 1) == '/') {  
 while (i < text.length() && text.charAt(i) != '\n') {  
 i++;  
 }  
 } else if (!insideComment) {  
 newText.append(text.charAt(i));  
 }  
 }  
 return newText.toString();  
 }

public static void main(String[] args){  
 String text;

text = *StrIn*();  
String newText = *num9*(text);  
if (newText.isEmpty()) {  
 System.*out*.println("Текст не содержит комментариев.");  
} else {  
 System.*out*.println("Обработанный текст:\n" + newText);  
}

}

## 9.3. Тестирование работы программы



# 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

## 10.3. Тестирование работы программы

# Задание 11

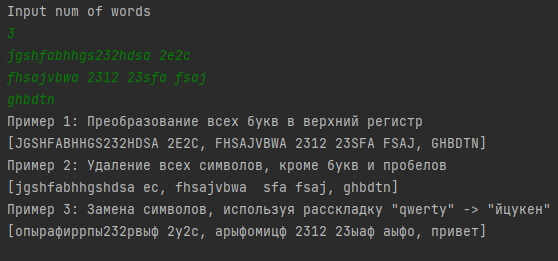
## 11.1. Постановка задачи

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

public static void num11(String [] strings){  
 String[] formattedStrings1 = *num11\_1*(strings);  
 System.*out*.println("Пример 1: Преобразование всех букв в верхний регистр");  
 System.*out*.println(Arrays.*toString*(formattedStrings1));  
 String[] formattedStrings2 = *num11\_2*(strings);  
 System.*out*.println("Пример 2: Удаление всех символов, кроме букв и пробелов");  
 System.*out*.println(Arrays.*toString*(formattedStrings2));  
 String[] formattedStrings3 = *num11\_3*(strings);  
 System.*out*.println("Пример 3: Замена символов, используя расскладку \"qwerty\" -> \"йцукен\"");  
 System.*out*.println(Arrays.*toString*(formattedStrings3));  
 }  
public static String[] num11\_1(String[] strings) {  
 String[] formattedStrings = new String[strings.length];  
 for (int i = 0; i < strings.length; i++) {  
 formattedStrings[i] = strings[i].toUpperCase();  
 }  
 return formattedStrings;  
}  
public static String[] num11\_2(String[] strings) {  
 String[] formattedStrings = new String[strings.length];  
 for (int i = 0; i < strings.length; i++) {  
 formattedStrings[i] = strings[i].replaceAll("[^a-zA-Z\\s]", "");  
 }  
 return formattedStrings;  
}  
public static String[] num11\_3(String[] strings) {  
 String[] formattedStrings = new String[strings.length];  
 for (int i = 0; i < strings.length; i++) {  
 formattedStrings[i] = strings[i].replaceAll("q", "й")  
 .replaceAll("w", "ц")  
 .replaceAll("e", "у")  
 .replaceAll("r", "к")  
 .replaceAll("t", "е")  
 .replaceAll("y", "н")  
 .replaceAll("u", "г")  
 .replaceAll("i", "ш")  
 .replaceAll("o", "щ")  
 .replaceAll("p", "з")  
 .replaceAll("a", "ф")  
 .replaceAll("s", "ы")  
 .replaceAll("d", "в")  
 .replaceAll("f", "а")  
 .replaceAll("g", "п")  
 .replaceAll("h", "р")  
 .replaceAll("j", "о")  
 .replaceAll("k", "л")  
 .replaceAll("l", "д")  
 .replaceAll("z", "я")  
 .replaceAll("x", "ч")  
 .replaceAll("c", "с")  
 .replaceAll("v", "м")  
 .replaceAll("b", "и")  
 .replaceAll("n", "т")  
 .replaceAll("m", "ь");  
 }  
 return formattedStrings;  
}

## 11.3. Тестирование работы программы



# Задание 12

## 12.1. Постановка задачи

Дан массив строк. Создать массив регулярных выражений [учащийся, обучающийся, ученик, студент, бакалавр, магистр, 2020] и массив замен [студент, отчисленный, специалист, выпускник, магистр, аспирант, 2021]. В цикле с использование регулярных выражений выполнить автозамену информации. Исходную и измененную информацию вывести построчно: сначала исходная строка, затем измененная.

## 12.2. Решение задачи, код программы

public static void main(String[] args){

String[] stringes = {  
 "учащийся образовательного учереждения",  
 "обучающийся в ВУЗе",  
 "профильнаая работа для ученика",  
 "студент ПНИПУ 2023",  
 "я получил степень бакалавра",  
 "я обучаюсь на магистра",  
 "событие датируемое 2020 годом"  
 };  
 String[] searchWords = {  
 "учащийся",  
 "обучающийся",  
 "ученик",  
 "студент",  
 "бакалавр",  
 "магистр",  
 "2020"  
 };  
 String[] replacements = {  
 "студент",  
 "отчисленный",  
 "специалист",  
 "выпускник",  
 "магистр",  
 "аспирант",  
 "2021"  
 };  
 *num12*(stringes, searchWords, replacements);  
 break;  
}

public static void num12(String[] strings, String[] searchWords, String[] replacements){  
 for (int i = 0; i < strings.length; i++) {  
 String originalString = strings[i];  
 String modifiedString = originalString;  
 for (int j = 0; j < searchWords.length; j++) {  
 modifiedString = modifiedString.replaceAll(searchWords[j], replacements[j]);  
 }  
 System.*out*.println("Исходная строка: " + originalString);  
 System.*out*.println("Измененная строка: " + modifiedString);  
 System.*out*.println();  
 }  
}

## 12.3. Тестирование работы программы

