Министерство науки и высшего образования РФ

ФГАОУ ВО Пермский национальный исследовательский

политехнический университет

Кафедра «Вычислительная математика, механика и биомеханика»

Отчет по лабораторной работе № 10

тема «Символы и строки»

по дисциплине «Информатика»

Выполнил: студент группу ИСТ-22-1б Ипанов М. В.

Проверил: Нетбай Георгий Владимирович

Пермь, 2022

**Содержание**

[Задание 1 3](#_Toc116149685)

[1.1. Постановка задачи 3](#_Toc116149686)

[1.2. Решение задачи, код программы 3](#_Toc116149687)

[1.3. Тестирование работы программы с проверкой 3](#_Toc116149688)

[2.1. Постановка задачи 5](#_Toc116149689)

[2.2. Решение задачи, код программы 5](#_Toc116149690)

[2.3. Тестирование работы программы с проверкой 5](#_Toc116149691)

[3.1. Постановка задачи 6](#_Toc116149692)

[3.2. Решение задачи, код программы 6](#_Toc116149693)

[3.3. Тестирование работы программы с проверкой 6](#_Toc116149694)

[4.1. Постановка задачи 7](#_Toc116149695)

[4.2. Решение задачи, код программы 7](#_Toc116149696)

[4.3. Тестирование работы программы с проверкой 7](#_Toc116149697)

# Задание 1

## 1.1. Постановка задачи

Переделать программу задания 3 лабораторной работы 9. Массив данных записан в файл и считывается в программе для обработки. Результаты обработки массива согласно вопросам задания 3 лабораторной работы 9 записываются в конец файла построчно с заголовками расшифровывающими, что за данные представлены ниже.

## 1.2. Решение задачи, код программы

import java.io.\*;  
import java.util.\*;  
  
public class zadanie\_1 {  
 public static char vvod() {  
 Scanner in = new Scanner(System.*in*);  
 String x = in.nextLine();  
 char t = ' ';  
 while (true) {  
 if (x.length() != 1) {  
 System.*out*.println("Элемент массива должен быть одним символом");  
 x = in.nextLine();  
 } else {  
 t = x.charAt(0);  
 break;  
 }  
 }  
 return t;  
 }  
 public static void write(String a,boolean bo) {  
 try(FileWriter writer = new FileWriter("text1.txt", bo))  
 {  
 writer.append(a);  
 writer.flush();  
 }  
 catch(IOException ex){  
  
 System.*out*.println(ex.getMessage());  
 }  
 }  
 public static void mas(char[] array) {  
 try (FileReader reader = new FileReader("text1.txt")) {  
 int c;  
 while ((c = reader.read(array)) > 0) {  
  
 if (c < 10) {  
 array = Arrays.*copyOf*(array, c);  
 }  
 System.*out*.println(array);  
 }  
 } catch (IOException ex) {  
  
 System.*out*.println(ex.getMessage());  
 }  
 }  
  
 public static int abc(char[] array, char s) {  
 int a = 0;  
 for (int i = 0; i < 10; i++) {  
 if (array[i] == s) {  
 a += 1;  
 }  
 }  
 return a;  
 }  
  
 public static boolean c(char[] array, String s) {  
 boolean cp = true;  
 for (int i = 0; i < s.length(); i++) {  
 if (*abc*(array, s.charAt(i)) == 0) {  
 cp = false;  
 break;  
 }  
 }  
 return cp;  
 }  
  
 public static boolean d(char[] array, String s) {  
 boolean a = false;  
 for (int i = 0; i < 9; i++) {  
 if ((String.*valueOf*(array[i]) + String.*valueOf*(array[i + 1])).equals(s)) {  
 a = true;  
 break;  
 }  
 }  
 return a;  
 }  
  
 public static boolean e(char[] array) {  
 boolean a = false;  
 char kostil = array[0];  
 for (int i = 1; i < 9; i++) {  
 if (array[i] == kostil) {  
 a = true;  
 break;  
 }  
 kostil = array[i];  
 }  
 return a;  
 }  
  
 public static boolean f(char[] array, int i, int l) {  
 boolean a = false;  
 if (array[i] == array[(i + l) % 10]) {  
 a = true;  
 }  
 return a;  
 }  
  
 public static void main(String[] args) {  
 Scanner in = new Scanner(System.*in*);  
 char[] array = new char[10];  
 *mas*(array);  
 System.*out*.println("б) Введите символ");  
 char b = *vvod*();  
 System.*out*.println("в) Введите строку");  
 String c = in.next();  
 int i = -1;  
 int j = -1;  
 int l = -1;  
 while (true) {  
 System.*out*.println("e) Введите l (1<i<j<11)");  
 l = in.nextInt();  
 System.*out*.println("e) Введите i (1<i<j<11)");  
 i = in.nextInt() - 1;  
 System.*out*.println("e) Введите j (1<i<j<11)");  
 j = in.nextInt() - 1;  
 if (i + 1 > l & j + 1 > i + 1 & 11 > j + 1) {  
 break;  
 }  
 }  
 *write*("\nа) " + *abc*(array, ' '), true);  
 *write*("\nб) " + *abc*(array, b), true);  
 *write*("\nв) " + *c*(array, c),true);  
 *write*("\nг) " + (*d*(array, "но") | *d*(array, "он")),true );  
 *write*("\nд) " + *e*(array),true);  
 *write*("\ne) " + (*f*(array, i, l) & *f*(array, j, l)),true);  
 }  
}

## 1.3. Тестирование работы программы

Таблица 1

Тестирование работы программы и проверка результатов решения

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

**Задача 2**

## 2.1. Постановка задачи

Записать в файл таблицу из задания 7 лабораторной работы 6. Если в таблице присутствуют дробные ответы, то вывести данные в формате 4 знака после запятой.

## 2.2. Решение задачи, код программы

import java.io.\*;  
public class zadanie\_2 {  
 public static void file(String a,boolean bo) {  
 try(FileWriter writer = new FileWriter("text2.txt", bo))  
 {  
 writer.write(a);  
 writer.flush();  
 }  
 catch(IOException ex){  
  
 System.*out*.println(ex.getMessage());  
 }  
 }  
 public static void main(String[] args) {  
 *file*("",false);  
 for (int i = 1; i < 10; i++) {  
 for (int j = 1; j < 10; j++) {  
 *file*(j+"-"+i+"="+(i-j)+" ",true);  
 }  
 *file*("\n",true);  
 }  
 }  
}

## 2.3. Тестирование работы программы

Таблица 2

Тестирование работы программы и проверка результатов решения

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

**Задача 3**

## 3.1. Постановка задачи

Напечатать все слова из текстового файла, начинающиеся на большую и заканчивающиеся на маленькую букву.

## 3.2. Решение задачи, код программы

import java.io.\*;  
  
public class zadanie\_3 {  
 public static String file() {  
 StringBuilder a = new StringBuilder();  
 try (FileReader reader = new FileReader("text3.txt")) {  
 int c;  
 while ((c = reader.read()) != -1) {  
 a.append((char) c);  
 }  
 } catch (IOException ex) {  
 System.*out*.println(ex.getMessage());  
 }  
 return a.toString();  
 }  
  
 public static void main(String[] args) {  
 String st = *file*() + " ";  
 for (int i = 0; i < st.length(); i++) {  
 if ((st.charAt(i) == ' ' | i == st.length() - 1)) {  
 int j = 1;  
 if (Character.*isLowerCase*(st.charAt(i - j))) {  
 while (i - j != 0) {  
 j++;  
 if (i - j == 0) {  
 if (Character.*isUpperCase*(st.charAt(0))) {  
 System.*out*.println(st.substring(0,i));  
 }  
 break;  
 } else if (st.charAt(i - j - 1) == ' ') {  
 if (Character.*isUpperCase*(st.charAt(i - j))) {  
 System.*out*.println(st.substring(i-j,i));  
 }  
 break;  
 }  
 }  
 }  
 }  
 }  
 }  
}

## 3.3. Тестирование работы программы

Таблица 3

Тестирование работы программы и проверка результатов решения

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

**Задача 4**

## 4.1. Постановка задачи

Выделить в текстовом файле все слова, разделенные символами-разделителями «\_.,;:\n\t!?», и все слова, в которых нет согласных латинских букв, и оно является симметричным, построчно записать в другой текстовый файл.

## 4.2. Решение задачи, код программы

import java.io.\*;  
  
public class zadanie\_4 {  
 public static String read() {  
 StringBuilder a = new StringBuilder();  
 try (FileReader reader = new FileReader("text4.txt")) {  
 int c;  
 while ((c = reader.read()) != -1) {  
 a.append((char) c);  
 }  
 } catch (IOException ex) {  
 System.*out*.println(ex.getMessage());  
 }  
 return a.toString();  
 }  
  
 public static void write(String a, boolean bo) {  
 try (FileWriter writer = new FileWriter("text4\_1.txt", bo)) {  
 writer.write(a);  
 writer.flush();  
 } catch (IOException ex) {  
  
 System.*out*.println(ex.getMessage());  
 }  
 }  
  
 public static void main(String[] args) {  
 String st = *read*() + " ";  
 *write*("", false);  
 String st2 = "\_.,;:\n\t!?";  
 String st3 = "BCDFGHJKLMNPQRSTVWXZbcdfghjklmnpqrstvwxz";  
 for (int k = 0; k < st.length(); k++) {  
 if (st2.indexOf(st.charAt(k)) > -1) {  
 st = st.replace(st.charAt(k), '\n');  
 }  
 }  
 for (int i = 0; i < st.length(); i++) {  
 if ((st.charAt(i) == ' ' | i == st.length() - 1)) {  
 int j = 1;  
 boolean t = false;  
 boolean t2 = true;  
 while (i - j >= 0) {  
 if (st.charAt(i - j) == '\n') {  
 t = true;  
 t2 = false;  
 }  
 if (st3.indexOf(st.charAt(i - j)) > -1) {  
 t2 = false;  
 }  
 if (i - j == 0) {  
 if (t | t2) {  
 *write*(st.substring(0, i) + "\n", true);  
 }  
 break;  
 } else if (st.charAt(i - j - 1) == ' ') {  
 if (t | t2) {  
 *write*(st.substring(i - j, i) + "\n", true);  
 }  
 break;  
 }  
 j++;  
 }  
 }  
 }  
 }  
}

## 4.3. Тестирование работы программы

Таблица 4

Тестирование работы программы и проверка результатов решения

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

**Задача 5**

## 5.1. Постановка задачи

Дано два файла с предложениями. Дописать в конец второго файла предложения из первого файла, в которых есть слова, начинающиеся и заканчивающиеся на одну и ту же букву.

## 5.2. Решение задачи, код программы

import java.io.\*;  
  
public class zadanie\_5 {  
 public static String read() {  
 StringBuilder a = new StringBuilder();  
 try (FileReader reader = new FileReader("text5.txt")) {  
 int c;  
 while ((c = reader.read()) != -1) {  
 a.append((char) c);  
 }  
 } catch (IOException ex) {  
 System.out.println(ex.getMessage());  
 }  
 return a.toString();  
 }  
  
 public static void write(String a, boolean bo) {  
 try (FileWriter writer = new FileWriter("text5\_1.txt", bo)) {  
 writer.write(a);  
 writer.flush();  
 } catch (IOException ex) {  
  
 System.out.println(ex.getMessage());  
 }  
 }  
  
 public static void main(String[] args) {  
 String st = (read()+" ").replaceAll("\n", "").replace("\r", " ");  
 String zn = ".?!";  
 int nap=-1, nas=-1;  
 boolean t=false;  
 for (int i = 0; i<st.length()-1; i++) {  
 if (st.charAt(i)!=' ' & zn.indexOf(st.charAt(i))==-1 & nap==-1) {  
 nap=i;  
 }  
 if (st.charAt(i)!=' ' & zn.indexOf(st.charAt(i))==-1 & nas==-1) {  
 nas=i;  
 }  
 if (nas!=-1) {  
 if ((st.charAt(i)==st.charAt(nas) | st.charAt(i)==Character.*toLowerCase*(st.charAt(nas)) | st.charAt(i)==Character.*toUpperCase*(st.charAt(nas))) &  
 !Character.*isLetter*(st.charAt(i+1)) & Character.*isLetter*(st.charAt(nas)) & nas!=i) {  
 t=true;  
 }  
 }  
 if (zn.indexOf(st.charAt(i))>-1) {  
 if (t) {  
 *write*(st.substring(nap, i + 1) + " ", true);  
 }  
 nap=-1;  
 t=false;  
 }  
 if (st.charAt(i)==' ' | zn.indexOf(st.charAt(i))>-1){  
 nas=-1;  
 }  
 }  
 }  
}

## 5.3. Тестирование работы программы с проверкой

Таблица 5

Тестирование работы программы и проверка результатов решения

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

**Задача 6**

## 6.1. Постановка задачи

Входной файл содержит сведения об автомобилях: марка автомобиля, масса автомобиля в тоннах, год выпуска, цена. Найти автомобили с массой от 2 до 5 тонн, год выпуска которых от 2000 до 2007, подсчитать их количество и дописать ответ в конец файла.

## 6.2. Решение задачи, код программы

import java.io.\*;  
import java.util.\*;  
  
public class zadanie\_6 {  
 public static String read() {  
 StringBuilder a = new StringBuilder();  
 try (FileReader reader = new FileReader("text6.txt")) {  
 int c;  
 while ((c = reader.read()) != -1) {  
 a.append((char) c);  
 }  
 } catch (IOException ex) {  
 System.*out*.println(ex.getMessage());  
 }  
 return a.toString();  
 }  
  
 public static void write(String a, boolean bo) {  
 try (FileWriter writer = new FileWriter("text6.txt", bo)) {  
 writer.write(a);  
 writer.flush();  
 } catch (IOException ex) {  
  
 System.*out*.println(ex.getMessage());  
 }  
 }  
  
 public static void main(String[] args) {  
 ArrayList<Double> car = new ArrayList<Double>();  
 String cs = *read*();  
 int k = 0, j = 1, c = 0;  
 boolean t2 = true, t3 = true;  
 for (int i = 0; i < cs.length(); i++) {  
 if (cs.charAt(i) == ';')  
 k++;  
 if (k == 2 & t2) {  
 while (true) {  
 if (cs.charAt(i - j) == ';') {  
 car.add(Double.*parseDouble*(cs.substring(i - j + 2, i)));  
 j = 1;  
 t2 = false;  
 break;  
 }  
 j++;  
 }  
 }  
 if (k == 3 & t3) {  
 while (true) {  
 if (cs.charAt(i - j) == ';') {  
 car.add(Double.*parseDouble*(cs.substring(i - j + 2, i)));  
 j = 1;  
 t3 = false;  
 break;  
 }  
 j++;  
 }  
 }  
 if (k == 4) {  
 k = 0;  
 t2 = true;  
 t3 = true;  
 }  
 }  
 for (int i = 0; i < car.size(); i += 2) {  
 if (car.get(i) >= 2 & car.get(i) <= 5 & car.get(i + 1) >= 2000 & car.get(i + 1) <= 2007) {  
 c++;  
 }  
 }  
 *write*("\nКоличество машин, удовлетворяющих критериям: " + c, true);  
 }  
}

## 6.3. Тестирование работы программы с проверкой

Таблица 5

Тестирование работы программы и проверка результатов решения

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

**Задача 7**

## 7.1. Постановка задачи

Записать в файл значения функции f(x) из задания 3 лабораторной работы 5 для всего диапазона переменной в виде: значение x; значение f(x). В начале файла должна быть указана постановка задачи, т.е. диапазон аргумента и вид функции (запись на языке java). Перед выводом цифровых значений должен быть сделан заголовок о том, в каком порядке идет вывод.

## 7.2. Решение задачи, код программы

import java.io.\*;  
  
import static java.lang.Math.\*;  
  
public class zadanie\_7 {  
 public static void write(String a, boolean bo) {  
 try (FileWriter writer = new FileWriter("text7.txt", bo)) {  
 writer.write(a);  
 writer.flush();  
 } catch (IOException ex) {  
  
 System.*out*.println(ex.getMessage());  
 }  
 }  
  
 public static void main(String[] args) {  
 *write*("Диапазон аргумента: -3<=x<=3\nФункция:\nif (x >= 0) \n\tx = sin(PI)\nelse \n\tx = PI \* pow(x, 2)\nВывод в порядке возрастания x\n", false);  
 for (Double x = -3.0; x <= 3.0; x += 0.5) {  
 if (x >= 0)  
 *write*(("x=" + x + "; f(x)=" + *sin*(*PI*) \* x + "\n"), true);  
 else  
 *write*(("x=" + x + "; f(x)=" + *PI* \* *pow*(x, 2) + "\n"), true);  
 }  
 }  
}

## 7.3. Тестирование работы программы с проверкой

Таблица 5

Тестирование работы программы и проверка результатов решения

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