|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ **Информатика, ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ и системы  
 управления**

КАФЕДРА **Компьютерные системы и сети (ИУ6)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.04.01 Информатика и вычислительная техника**

МАГИСТЕРСКАЯ ПРОГРАММА **09.04.01/07 Интеллектуальные системы анализа,**

**обработки и интерпретации больших данных**

**Отчет**

**по лабораторной работе №2**

**Название:** Арифметические операции

**Дисциплина:** Языки программирования для работы с большими данными

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ИУ6-22М |  |  | И.Л. Баришпол |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  | П.В. Степанов |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2023

**Задания**

3. Ввести n строк с консоли. Вывести на консоль те строки, длина которых меньше (больше) средней, а также длину.

import java.util.Scanner;

import java.util.Date;

import java.util.Calendar;

public class Var1\_3 {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the number of lines to input: ");

int n = scanner.nextInt();

scanner.nextLine();

String[] lines = new String[n];

int totalLength = 0;

for (int i = 0; i < n; i++) {

System.out.print("Enter a line: ");

lines[i] = scanner.nextLine();

totalLength += lines[i].length();

}

double averageLength = (double) totalLength / n;

System.out.println("Lines with length less than the average (" + averageLength + "):");

for (int i = 0; i < n; i++) {

if (lines[i].length() < averageLength) {

System.out.println(lines[i] + " (length: " + lines[i].length() + ")");

}

}

System.out.println("Lines with length greater than the average (" + averageLength + "):");

for (int i = 0; i < n; i++) {

if (lines[i].length() > averageLength) {

System.out.println(lines[i] + " (length: " + lines[i].length() + ")");

}

}

System.out.println(GetMetaData());

scanner.close();

}

public static String GetMetaData() {

Calendar cal = Calendar.getInstance();

cal.set(Calendar.YEAR, 2023);

cal.set(Calendar.MONTH, Calendar.MARCH);

cal.set(Calendar.DAY\_OF\_MONTH, 3);

cal.set(Calendar.HOUR, 10);

cal.set(Calendar.MINUTE, 30);

Date receivingDate = cal.getTime();

Date now = new Date();

return "Developer: Igor Barishpol\n" + "Task received on: " + receivingDate + "\nTask delivered on: " + now;

}

}

4. Ввести n слов с консоли. Найти слово, в котором число различных символов минимально. Если таких слов несколько, найти первое из них.

import java.util.Scanner;

public class Var1\_4 {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the number of words to input: ");

int n = scanner.nextInt();

scanner.nextLine();

String[] words = new String[n];

for (int i = 0; i < n; i++) {

System.out.print("Enter a word: ");

words[i] = scanner.nextLine();

}

String minDistinctWord = words[0];

int minDistinct = Integer.MAX\_VALUE;

for (String word : words) {

int distinct = countDistinctCharacters(word);

if (distinct < minDistinct) {

minDistinct = distinct;

minDistinctWord = word;

}

}

System.out.println("Word with smallest number of distinct characters: " + minDistinctWord);

System.out.println(Var1\_3.GetMetaData());

scanner.close();

}

private static int countDistinctCharacters(String word) {

boolean[] seen = new boolean[256];

int count = 0;

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

int c = word.charAt(i);

if (!seen[c]) {

seen[c] = true;

count++;

}

}

return count;

}

}

Ввести с консоли n – размерность матрицы a[n][n]. Задать значения элементов матрицы в интервале значений от -n до n с помощью датчика случайных чисел.

3. Найти и вывести наибольшее число возрастающих (убывающих) элементов матрицы, идущих подряд.

import java.util.Scanner;

public class Var2\_3 {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the dimension of the matrix: ");

int n = scanner.nextInt();

int[][] a = new int[n][n];

for (int i = 0; i < n; i++) {

for (int j = 0; j < n; j++) {

a[i][j] = (int) (Math.random() \* (2 \* n + 1)) - n;

}

}

int longestIncreasing = 1;

int longestDecreasing = 1;

for (int i = 0; i < n; i++) {

int increasing = 1;

int decreasing = 1;

for (int j = 1; j < n; j++) {

if (a[i][j] > a[i][j - 1]) {

increasing++;

decreasing = 1;

} else if (a[i][j] < a[i][j - 1]) {

decreasing++;

increasing = 1;

} else {

increasing = 1;

decreasing = 1;

}

if (increasing > longestIncreasing) {

longestIncreasing = increasing;

}

if (decreasing > longestDecreasing) {

longestDecreasing = decreasing;

}

}

}

System.out.println("Matrix:");

PrintMatrix(a);

System.out.println("Longest increasing sequence length: " + longestIncreasing);

System.out.println("Longest decreasing sequence length: " + longestDecreasing);

scanner.close();

}

public static void PrintMatrix(int[][] matrix) {

for (int[] row : matrix) {

for (int i : row) {

System.out.print(i + " ");

}

System.out.println();

}

}

}

4. Найти сумму элементов матрицы, расположенных между первым и вторым положительными элементами каждой строки.

import java.util.Scanner;

public class Var2\_4 {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the dimension of the matrix: ");

int n = scanner.nextInt();

int[][] a = new int[n][n];

for (int i = 0; i < n; i++) {

for (int j = 0; j < n; j++) {

a[i][j] = (int) (Math.random() \* (2 \* n + 1)) - n;

}

}

int sum = 0;

for (int i = 0; i < n; i++) {

boolean foundFirstPositive = false;

int firstPositiveIndex = -1;

for (int j = 0; j < n; j++) {

if (a[i][j] > 0) {

if (!foundFirstPositive) {

foundFirstPositive = true;

firstPositiveIndex = j;

} else {

for (int k = firstPositiveIndex + 1; k < j; k++) {

sum += a[i][k];

}

break;

}

}

}

}

System.out.println("Matrix:");

Var2\_3.PrintMatrix(a);

System.out.println("Sum of elements between first and second positive elements of each row: " + sum);

scanner.close();

}

}

**Вывод:** в ходе выполнения данной лабораторной работы были освоены основы работы с арифметическими операциями в языке программирования Java. Данные задачи помогли закрепить знания по основным конструкциям языка и приобрести опыт работы с арифметическими операциями в Java.