|  |  |
| --- | --- |
| 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 и системы управления**

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

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.04.01 Информатика и вычислительная техника**

МАГИСТЕРСКАЯ ПРОГРАММА **09.04.01/12 Интеллектуальный анализ больших**

**данных в системах поддержки принятия решений**

**Отчет**

|  |
| --- |
| **по лабораторной работе № 2** |

Вариант 6

**Название:** Арифметические операции

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

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ИУ6-23М |  |  | В.А. Гордеев |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  | П.В. Степанов |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2023

**ВВЕДЕНИЕ**

**Задание:**

* 1. В задании необходимо вывести внизу фамилию разработчика, дату и время получения задания, а также дату и время сдачи задания. Для получения последней даты и времени следует использовать класс Date. Ввести n слов с консоли. Найти слово, символы в котором идут в строгом порядке возрастания их кодов. Если таких слов несколько, найти первое из них.
  2. В задании необходимо вывести внизу фамилию разработчика, дату и время получения задания, а также дату и время сдачи задания. Для получения последней даты и времени следует использовать класс Date. Ввести n слов с консоли. Найти слово, состоящее только из различных символов. Если таких слов несколько, найти первое из них.
  3. 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. Вычислить норму матрицы.
  4. 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. Повернуть матрицу на 90 (180, 270) градусов против часовой стрелки.

**ОСНОВНАЯ ЧАСТЬ**

**Задание 1.1**

В программе выводится внизу фамилия разработчика, дата и время получения задания, а также дата и время сдачи задания. Для получения последней даты и времени используется класс Date. Вводится n слов с консоли. Выводится слово, символы в котором идут в строгом порядке возрастания их кодов. Если таких слов несколько, найти первое из них.

Код программы представлен в листинге 1.

Листинг 1 – Код программы 2.1

import java.util.Date;  
import java.util.Scanner;  
// В задании необходимо вывести внизу фамилию разработчика, дату и время получения задания, а также дату и время сдачи задания.  
// Для получения последней даты и времени следует использовать класс Date.  
// Ввести n слов с консоли. Найти слово, символы в котором идут в строгом порядке возрастания их кодов.  
// Если таких слов несколько, найти первое из них.  
public class Main {  
 public static void main(String[] args) {  
 Scanner sc = new Scanner(System.*in*);  
  
 System.*out*.println("Введите количество слов:");  
 int n = sc.nextInt();  
  
 Boolean success = false;  
 String result = "";  
 for (int i = 0; i < n; i++) {  
 String s = sc.next();  
 if (!success && *upperWord*(s)) {  
 result = s;  
 success = true;  
 }  
 }  
 System.*out*.println("Слово, символы в котором идут в строгом порядке возрастания их кодов: " + result);  
  
 Date date = new Date();  
 System.*out*.println("Фамилия разработчика: Гордеев");  
 System.*out*.println("Дата и время выдачи задания: 17.02.2023 15:40");  
 System.*out*.println("Дата и время сдачи задания: " + date);  
 sc.close();  
 }  
 public static boolean upperWord(String word) {  
 word.chars().forEach(c -> System.*out*.print(c + " "));  
 System.*out*.println();;  
 for (int i = 0; i < word.length() - 1; i++) {  
 if (word.charAt(i) >= word.charAt(i + 1)) {  
 return false;  
 }  
 }  
 return true;  
 }  
}

Результат работы программы 2.1 представлен на рисунке 1.

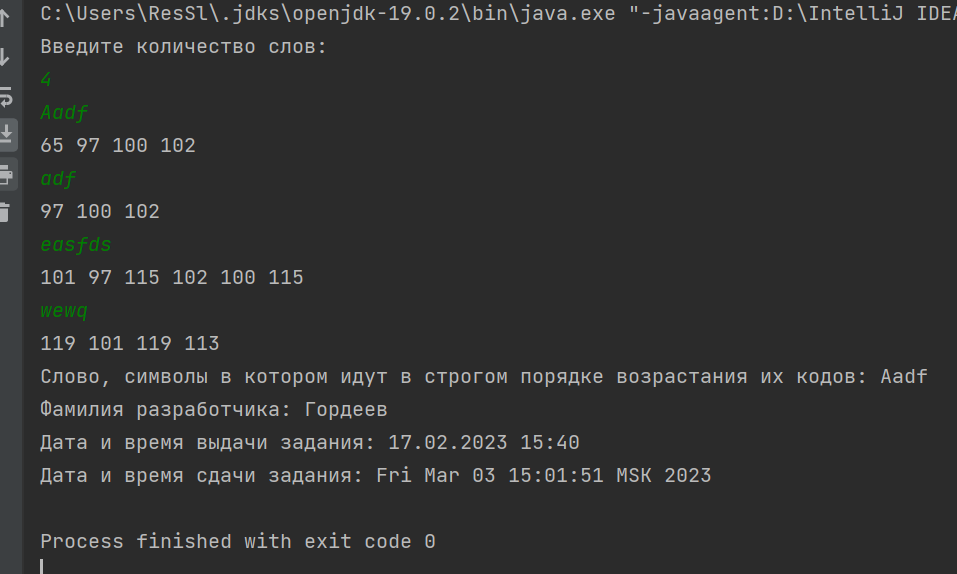


Рисунок 1 – Пример работы программы 2.1

**Задание 2.2**

В программе выводится внизу фамилия разработчика, дата и время получения задания, а также дата и время сдачи задания. Для получения последней даты и времени используется класс Date. Вводится n слов с консоли. Выводится слово, состоящее только из различных символов. Если таких слов несколько, находится первое из них.

Код программы представлен в листинге 2.

Листинг 2 – Код программы 2.2

import java.util.Date;  
import java.util.Scanner;  
// 1.2 В задании необходимо вывести внизу фамилию разработчика, дату и время получения задания, а также дату и время сдачи задания.  
// Для получения последней даты и времени следует использовать класс Date.   
// Ввести n слов с консоли. Найти слово, состоящее только из различных символов.   
// Если таких слов несколько, найти первое из них.  
public class Main {  
 public static void main(String[] args) {  
 Scanner sc = new Scanner(System.*in*);  
  
 System.*out*.println("Введите количество слов:");  
 int n = sc.nextInt();  
  
 Boolean success = false;  
 String result = "";  
 for (int i = 0; i < n; i++) {  
 String s = sc.next();  
 if (*isUnique*(s) && !success) {  
 result = s;  
 success = true;  
 }  
 }  
  
 System.*out*.println("Слово, состоящее только из уникальных символов: " + result);  
  
 Date date = new Date();  
 System.*out*.println("Фамилия разработчика: Гордеев");  
 System.*out*.println("Дата и время выдачи задания: 17.02.2023 15:40");  
 System.*out*.println("Дата и время сдачи задания: " + date);  
 sc.close();  
 }  
 public static boolean isUnique(String word) {  
 for (int i = 0; i < word.length(); i++) {  
 char c = word.charAt(i);  
 if (word.indexOf(c) != word.lastIndexOf(c)) {  
 return false;  
 }  
 }  
 return true;  
 }  
}

Пример работы программы 2.2 представлен на рисунке 2

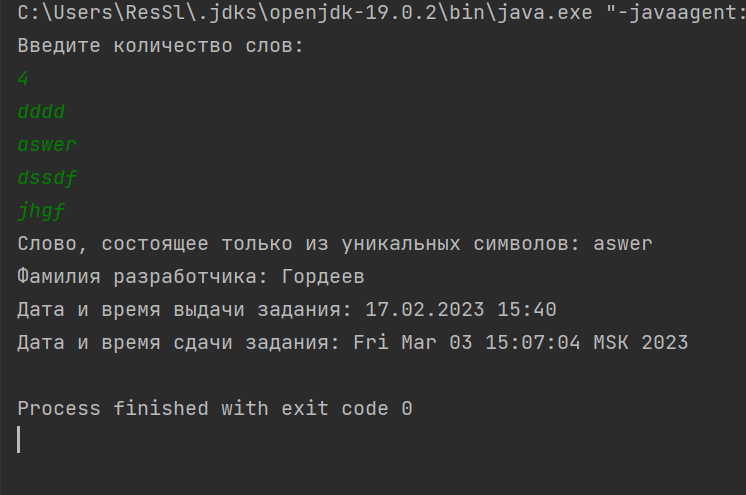


Рисунок 2 – Пример работы программы 2.2

**Задание 2.3**

Вводится с консоли n – размерность матрицы a[n][n]. Значения элементов матрицы задаются в интервале значений от -n до n с помощью датчика случайных чисел. В итоге вычисляется норма матрицы.

Код программы представлен в листинге 3.

Листинг 3 – Код программы 2.3

import java.util.Random;  
import java.util.Scanner;  
public class Main {  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.*in*);  
 Random rand = new Random();  
  
 System.*out*.println("Введите размерность матрицы ");  
 int n = scanner.nextInt();  
  
 int[][] matrix = new int[n][n];  
 for (int i = 0; i < n; i++) {  
 for (int j = 0; j < n; j++) {  
 matrix[i][j] = rand.nextInt(2\*n) - n;  
 System.*out*.print(matrix[i][j] + " ");  
 }  
 System.*out*.println();  
 }  
  
 int norm = 0;  
 for (int j = 0; j < n; j++) {  
 int sum = 0;  
 for (int i = 0; i < n; i++) {  
 sum += Math.*abs*(matrix[i][j]);  
 }  
 if (sum > norm) {  
 norm = sum;  
 }  
 }  
 System.*out*.println("1 Норма матрицы: " + norm);  
 }  
}

Результат работы программы показан на рисунке 3.

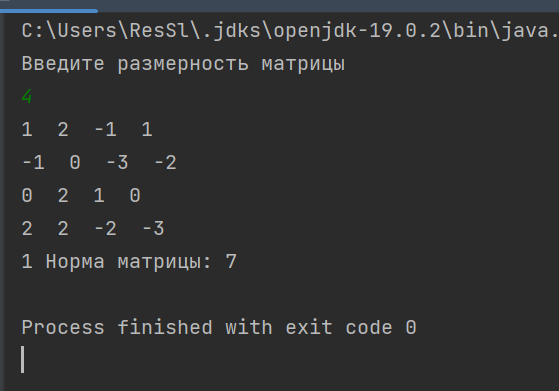


Рисунок 3 – Результат работы программы 3.

**Задание 2.4**

Вводится с консоли n – размерность матрицы a[n][n]. Задаются значения элементов матрицы в интервале значений от -n до n с помощью датчика случайных чисел. Матрица поворачивается на 90 (180, 270) градусов против часовой стрелки в зависимости от введенного угла.

Код программы представлен в листинге 4.

Листинг 4 – Код программы 1.4

import java.util.Random;  
import java.util.Scanner;  
// 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.   
// Повернуть матрицу на 90 (180, 270) градусов против часовой стрелки.  
public class Main {  
 public static void main(String[] args) {  
 Scanner scanner = new Scanner(System.*in*);  
 Random rand = new Random();  
  
 System.*out*.println("Введите размерность матрицы ");  
 int n = scanner.nextInt();  
  
 int[][] matrix = new int[n][n];  
 for (int i = 0; i < n; i++) {  
 for (int j = 0; j < n; j++) {  
 matrix[i][j] = rand.nextInt(2\*n) - n;  
 System.*out*.print(matrix[i][j] + " ");  
 }  
 System.*out*.println();  
 }  
  
 System.*out*.println("Введите угол поворота матрицы против часовой стрелки (90, 180, 270): ");  
 int angle = scanner.nextInt();  
  
 if (angle == 90 || angle == 180 || angle == 270) {  
 int countRotate = angle / 90;  
 for (int i = 0; i < countRotate; i++) {  
 *rotateMatrix90*(matrix);  
 }  
 } else {  
 System.*out*.println("Введен неверный угол, матрица повернута не будет");  
 }  
  
 System.*out*.println("Итоговая матрица: ");  
 *printMatrix*(matrix);  
 }  
 private static void rotateMatrix90(int[][] matrix) {  
 int n = matrix.length;  
 for (int i = 0; i < n/2; i++) {  
 for (int j = i; j < n-i-1; j++) {  
 int temp = matrix[i][j];  
 matrix[i][j] = matrix[j][n-i-1];  
 matrix[j][n-i-1] = matrix[n-i-1][n-j-1];  
 matrix[n-i-1][n-j-1] = matrix[n-j-1][i];  
 matrix[n-j-1][i] = temp;  
 }  
 }  
 }  
 private static void printMatrix(int[][] matrix) {  
 int n = matrix.length;  
 for (int i = 0; i < n; i++) {  
 for (int j = 0; j < n; j++) {  
 System.*out*.print(matrix[i][j] + " ");  
 }  
 System.*out*.println();  
 }  
 }  
}

Результат выполнения программы 2.4 представлен на рисунке 4.

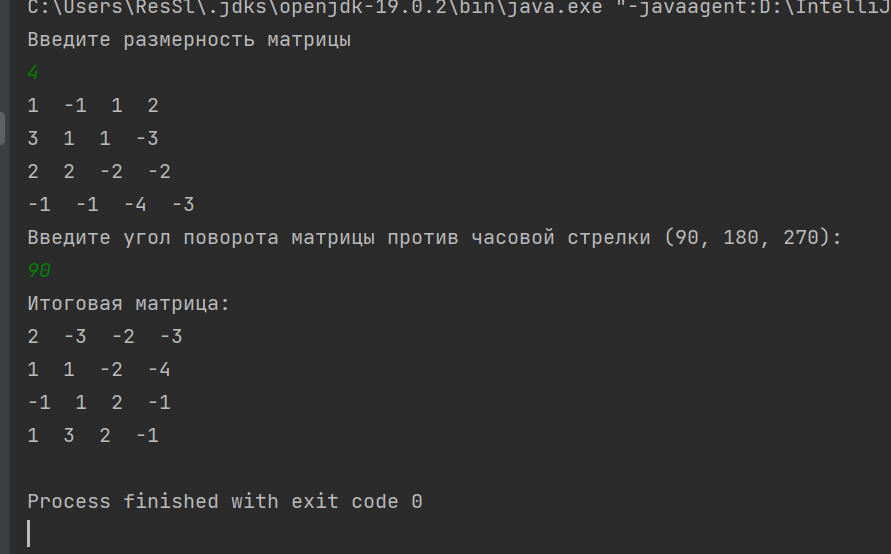


Рисунок 4 – Результат выполнения программы 2.4

**Вывод:** В результате выполнения лабораторной работы были получены практические навыки для работы с массивами, арифметическими операция и строками.