**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 **09.04.01/07 Интеллектуальные системы анализа, обработки и интерпретации больших данных**

О Т Ч Е Т

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

**Дисциплина:** Арифметические операции

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

|  |  |  |  |
| --- | --- | --- | --- |
| Студент | ИУ6-22М |  | Д.М. Карасёв\_\_ |
|  | (Группа) | (Подпись, дата) | (И.О. Фамилия) |
| Преподаватель |  | П.В. Степанов | |
|  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2023

**Цель:** ознакомиться с базовыми принципами языка Java для работы с большими данными

**Вариант 1:**

1. Ввести n строк с консоли, найти самую короткую и самую длинную строки. Вывести найденные строки и их длину.

2. Ввести n строк с консоли. Упорядочить и вывести строки в порядке возрастания (убывания) значений их длины.

**Решение:**

public class lr2\_task1 {  
 public static void main(String[] args)  
 {  
 Scanner in = new Scanner(System.*in*);  
 System.*out*.println("Введите n строк, если ввод окончен, то введите 'q': ");  
 String currentLine = in.nextLine();  
 String minLine = currentLine, maxLine = currentLine;  
 while (!(currentLine.equals("q"))){  
 if (minLine.length() > currentLine.length()){  
 minLine = currentLine;  
 }  
 if (minLine.length() < currentLine.length()){  
 maxLine = currentLine;  
 }  
 currentLine = in.nextLine();  
 }  
 System.*out*.println("Самая короткая строка:" + minLine + ", длина строки: " + minLine.length());  
 System.*out*.println("Самая длинная строка:" + maxLine + ", длина строки: " + maxLine.length());  
 System.*out*.println();  
 String lastname = "Карасёв";  
 System.*out*.println("Фамилия студента: " + lastname);  
 Calendar startWork = Calendar.*getInstance*();  
 startWork.set(Calendar.*YEAR*, 2023);  
 startWork.set(Calendar.*MONTH*, Calendar.*FEBRUARY*);  
 startWork.set(Calendar.*DAY\_OF\_MONTH*, 17);  
 startWork.set(Calendar.*HOUR\_OF\_DAY*, 15);  
 startWork.set(Calendar.*MINUTE*, 39);  
 System.*out*.println("Время выдачи задания: " + startWork.getTime());  
 Date currentTime = new Date();  
 System.*out*.println("Время сдачи задания: " + currentTime);  
 }  
}

На рисунке 1 представлен результат выполнения.

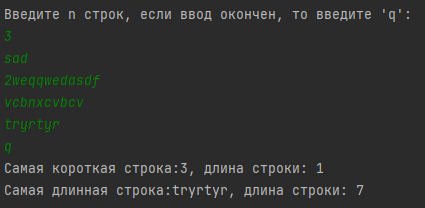


Рисунок 1 – Результат выполнения

public class lr2\_task2 {  
 public static void main(String[] args) throws IOException  
 {  
 Scanner in = new Scanner(System.*in*);  
 List<String> myArray = new ArrayList<String>();  
 System.*out*.println("Введите n строк, если ввод окончен, то введите 'q': ");  
 String currentLine = in.nextLine();  
 while (!(currentLine.equals("q"))){  
 myArray.add(currentLine);  
 currentLine = in.nextLine();  
 }  
  
 List<String> incArray = myArray, decArray = myArray;  
  
 for (int i = 0; i < decArray.size(); i++){  
 for (int j = 0; j < decArray.size()-i-1; j++){  
 if(decArray.get(j+1).length() > decArray.get(j).length()){  
 Collections.*swap*(decArray, j+1, j);  
 }  
 }  
 }  
 System.*out*.println("Сортиврока по убыванию длины строк:");  
 for (String x : decArray){  
 System.*out*.println(x);  
 }  
  
 for (int i = 0; i < incArray.size(); i++){  
 for (int j = 0; j < incArray.size()-i-1; j++){  
 if(incArray.get(j+1).length() < incArray.get(j).length()){  
 Collections.*swap*(incArray, j+1, j);  
 }  
 }  
 }  
 System.*out*.println("Сортиврока по возрастанию длины строк:");  
 for (String x : incArray){  
 System.*out*.println(x);  
 }  
  
 System.*out*.println();  
 String lastname = "Карасёв";  
 System.*out*.println("Фамилия студента: " + lastname);  
 Calendar startWork = Calendar.*getInstance*();  
 startWork.set(Calendar.*YEAR*, 2023);  
 startWork.set(Calendar.*MONTH*, Calendar.*FEBRUARY*);  
 startWork.set(Calendar.*DAY\_OF\_MONTH*, 17);  
 startWork.set(Calendar.*HOUR\_OF\_DAY*, 15);  
 startWork.set(Calendar.*MINUTE*, 39);  
 System.*out*.println("Время выдачи задания: " + startWork.getTime());  
 Date currentTime = new Date();  
 System.*out*.println("Время сдачи задания: " + currentTime);  
 }  
}

На рисунке 2 представлен результат выполнения.

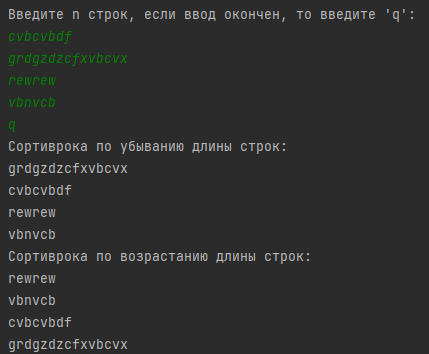


Рисунок 2 – Результат выполнения

**Вариант 2:**

1. Упорядочить строки (столбцы) матрицы в порядке возрастания значений элементов k-го столбца (строки).

2. Выполнить циклический сдвиг заданной матрицы на k позиций вправо (влево, вверх, вниз**).**

**Решение:**

public class lr2\_task1 {  
 public static void main(String[] args)  
 {  
 Scanner in = new Scanner(System.*in*);  
 System.*out*.println("Введите размерность матрицы:");  
 int n = in.nextInt();  
 int[][] myMatrix = new int[n][n], myMatrix2 = new int[n][n];  
  
 for (int i = 0; i < n; i++){  
 for (int j = 0; j < n; j++){  
 myMatrix[i][j] = (int)(Math.*random*()\*2\*(n+1)-(n+1));  
 }  
 }  
  
 System.*out*.println("Исходный массив:");  
 for (int i = 0; i < n; i++){  
 for (int j = 0; j < n; j++){  
 System.*out*.print(myMatrix[i][j] + " ");  
 }  
 System.*out*.println();  
 }  
 for (int i = 0; i < n; i++){  
 for (int j = 0; j < n; j++){  
 myMatrix2[i][j]=myMatrix[i][j];  
 }  
 }  
  
 System.*out*.println();  
 System.*out*.println("Введите номер столбца для сортировки:");  
 int k = in.nextInt()-1;  
 for (int i = 0; i < n-1; i++){  
 if(myMatrix[i+1][k] < myMatrix[i][k]){  
 for (int j = 0; j < n; j++){  
 int swap = myMatrix[i+1][j];  
 myMatrix[i+1][j] = myMatrix[i][j];  
 myMatrix[i][j] = swap;  
 }  
 }  
 }  
  
 System.*out*.println("Результат работы пограммы:");  
 for (int i = 0; i < n; i++){  
 for (int j = 0; j < n; j++){  
 System.*out*.print(myMatrix[i][j] + " ");  
 }  
 System.*out*.println();  
 }  
  
 System.*out*.println();  
 System.*out*.println("Введите номер строки для сортировки:");  
 k = in.nextInt()-1;  
 for (int i = 0; i < n-1; i++){  
 if(myMatrix2[k][i+1] < myMatrix2[k][i]){  
 for (int j = 0; j < n; j++){  
 int swap = myMatrix2[j][i+1];  
 myMatrix2[j][i+1] = myMatrix2[j][i];  
 myMatrix2[j][i] = swap;  
 }  
 }  
 }  
  
 System.*out*.println("Результат работы пограммы:");  
 for (int i = 0; i < n; i++){  
 for (int j = 0; j < n; j++){  
 System.*out*.print(myMatrix2[i][j] + " ");  
 }  
 System.*out*.println();  
 }  
 }  
}

На рисунке 3 представлен результат выполнения.

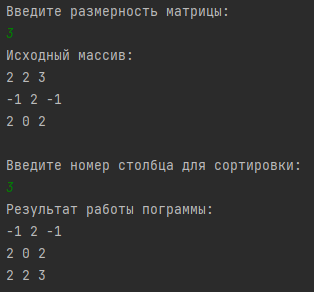


Рисунок 3 – Результат выполнения

public class lr2\_task2 {  
 public static void main(String[] args) {  
 Scanner in = new Scanner(System.*in*);  
 System.*out*.println("Введите размерность матрицы:");  
 int n = in.nextInt();  
 int[][] myMatrix = new int[n][n];  
 int[][] rightMatrix = new int[n][n], leftMatrix = new int[n][n], upMatrix = new int[n][n], downMatrix = new int[n][n];  
  
 for (int i = 0; i < n; i++) {  
 for (int j = 0; j < n; j++) {  
 myMatrix[i][j] = (int) (Math.*random*() \* 2 \* (n + 1) - (n + 1));  
 }  
 }  
  
 System.*out*.println("Исходная матрица:");  
 for (int i = 0; i < n; i++) {  
 for (int j = 0; j < n; j++) {  
 System.*out*.print(myMatrix[i][j] + " ");  
 }  
 System.*out*.println();  
 }  
  
 System.*out*.println("На сколько сделать сдвиг вправо?");  
 int k = in.nextInt();  
 for (int i = 0; i < n; i++){  
 for (int j = 0; j < n; j++){  
 if(j+k < n){  
 rightMatrix[i][j+k] = myMatrix[i][j];  
 }  
 else {  
 rightMatrix[i][(j+k)%n] = myMatrix[i][j];  
 }  
 }  
 }  
 System.*out*.println("Матрица со свдигом вправо:");  
 for (int i = 0; i < n; i++) {  
 for (int j = 0; j < n; j++) {  
 System.*out*.print(rightMatrix[i][j] + " ");  
 }  
 System.*out*.println();  
 }  
  
 System.*out*.println("На сколько сделать сдвиг влево?");  
 k = in.nextInt();  
 for (int i = 0; i < n; i++){  
 for (int j = 0; j < n; j++){  
 if(j-k >= 0){  
 leftMatrix[i][j-k] = myMatrix[i][j];  
 }  
 else {  
 leftMatrix[i][(j-k)+n] = myMatrix[i][j];  
 }  
 }  
 }  
 System.*out*.println("Матрица со свдигом влево:");  
 for (int i = 0; i < n; i++) {  
 for (int j = 0; j < n; j++) {  
 System.*out*.print(leftMatrix[i][j] + " ");  
 }  
 System.*out*.println();  
 }  
  
  
 System.*out*.println("На сколько сделать сдвиг вниз?");  
 k = in.nextInt();  
 for (int i = 0; i < n; i++){  
 for (int j = 0; j < n; j++){  
 if(i+k < n){  
 upMatrix[i+k][j] = myMatrix[i][j];  
 }  
 else {  
 upMatrix[(i+k)%n][j] = myMatrix[i][j];  
 }  
 }  
 }  
 System.*out*.println("Матрица со свдигом вниз:");  
 for (int i = 0; i < n; i++) {  
 for (int j = 0; j < n; j++) {  
 System.*out*.print(upMatrix[i][j] + " ");  
 }  
 System.*out*.println();  
 }  
  
 System.*out*.println("На сколько сделать сдвиг влево?");  
 k = in.nextInt();  
 for (int i = 0; i < n; i++){  
 for (int j = 0; j < n; j++){  
 if(i-k >= 0){  
 upMatrix[i-k][j] = myMatrix[i][j];  
 }  
 else {  
 upMatrix[(i-k)+n][j] = myMatrix[i][j];  
 }  
 }  
 }  
 System.*out*.println("Матрица со свдигом влево:");  
 for (int i = 0; i < n; i++) {  
 for (int j = 0; j < n; j++) {  
 System.*out*.print(upMatrix[i][j] + " ");  
 }  
 System.*out*.println();  
 }  
 }  
}

На рисунке 4 представлен результат выполнения.

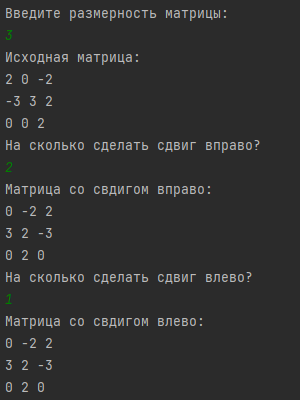


Рисунок 4 – Результат выполнения

**Вывод:** были разработаны классы и методы согласно вариантам.