|  |  |
| --- | --- |
| 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

Вариант: 2

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

**Вариант 1**

В приведенных ниже заданиях 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.

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

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

Листинг 1 – Код программы

import java.util.Date;

import java.util.Scanner;

public class Main {

static String fio;

static int days;

static Date BeginTaskDate() {

Date beginDate = new Date();

beginDate.getDate();

System.out.println("Date of begin " + beginDate);

return beginDate;

}

static Date FinishTaskDate() {

Date finishDate = new Date();

Long time = finishDate.getTime();

Scanner in = new Scanner(System.in);

System.out.println("Enter number of days ");

days = Integer.parseInt(in.nextLine());

time = time + (60 \* 60 \* 24 \* 1000 \* days);

finishDate = new Date(time);

System.out.println("Date Of Finish Task = " + finishDate);

return finishDate;

}

public static void main(String[] args) {

System.out.println("Enter number of rows:");

Scanner secin = new Scanner(System.in);

int rows = Integer.parseInt(secin.nextLine());

String min = null;

String max = null;

String[] strings = new String[rows];

String[] strings2 = new String[rows];

float average = 0;

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

Scanner in = new Scanner(System.in);

System.out.println("Enter your FIO ");

strings[i] = in.nextLine();

strings2[i] = strings[i];

average = average + strings[i].length(); //сумма длин строк

System.out.println("Your FIO is " + strings[i]);

BeginTaskDate();

FinishTaskDate();

}

for (int i = rows-1; i > 0; i--) {

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

if (strings[j].length() > strings[j + 1].length()) {

String temp = strings[j];

strings[j] = strings[j + 1];

strings[j + 1] = temp;

}

if (strings2[j].length() < strings2[j + 1].length()) {

String temp = strings2[j];

strings2[j] = strings2[j + 1];

strings2[j + 1] = temp;

}

}

}

System.out.println("Increase: "); //возрастание

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

System.out.println(strings[i] + " ");

}

System.out.println("Descending: "); //убывание

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

System.out.println(strings2[i] + " ");

}

// Ex 3

average = average / rows; //среднее арифметическое суммы длин этих строк

System.out.println("Average length: " + average);

System.out.println("More than average: ");

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

if (strings[j].length() > average) {

System.out.println(strings[j] + " length: " + strings[j].length());

}

}

System.out.println("Less than average: ");

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

if (strings[j].length() < average) {

System.out.println(strings[j] + " length: " + strings[j].length());

}

}

}

}

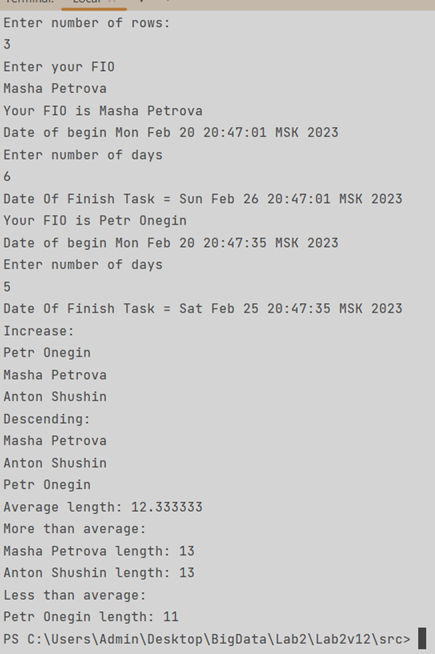


Рисунок 1 – Результат работы программы

**Вариант 2**

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

Листинг 2 – Код программы

import java.util.Arrays;

import java.util.Random;

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner in = new Scanner(System.in);

System.out.println("Enter matrix size");

int size = in.nextInt();

int arr[][] = new int[size][size];

int arr2[][] = new int[size][size];

int arr3[][] = new int[size][size];

int arr4[][] = new int[size][size];

Random random = new Random();

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

for (int j = 0; j < arr.length; j++) {

arr[j][i] = random.nextInt(2 \* size + 1) - size;

arr2[j][i] = arr[j][i];

arr3[j][i] = arr[j][i];

arr4[j][i] = arr[j][i];

}

}

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

for(int[]x:arr){

for(int z:x){

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

} System.out.println();

}

Scanner sc = new Scanner(System.in);

System.out.println("Enter shift left");

int step = sc.nextInt();

if (step > arr.length || step < 1) {

System.out.println("Shift impossible");

return;

}

int buff, i, j;

for (int r = 0; r < step; r++) {

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

buff = arr[i][0];

for (j = 0; j < arr.length - 1; j++) {

arr[i][j] = arr[i][j + 1];

}

arr[i][j] = buff;

}

}

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

for(int[]x:arr){

for(int z:x){

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

} System.out.println();

}

Scanner sc2 = new Scanner(System.in);

System.out.println("Enter shift right");

int step2 = sc.nextInt();

if (step2 > arr2.length || step2 < 1) {

System.out.println("Shift impossible");

return;

}

int buff2 = 0, i2, j2;

for (int r = 0; r < step2; r++) {

for (i2 = 0; i2 < arr2.length; i2++) {

buff2 = arr2[i2][arr2.length - 1];

for (j2 = arr2.length - 1; j2 > 0; j2--) {

arr2[i2][j2] = arr2[i2][j2 - 1];

}

arr2[i2][j2] = buff2;

}

}

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

for(int[]x:arr2){

for(int z:x){

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

} System.out.println();

}

Scanner sc3 = new Scanner(System.in);

System.out.println("Enter shift down");

int step3 = sc3.nextInt();

if (step3 > arr3.length || step3 < 1) {

System.out.println("Shift impossible");

return;

}

int buff3, i3, j3;

for (int r = 0; r < step; r++) {

for (i3 = 0; i3 < arr3.length; i3++) {

buff3 = arr3[arr3.length - 1][i3];

for (j3 = arr3.length - 1; j3 > 0; j3--) {

arr3[j3][i3] = arr3[j3 - 1][i3];

}

arr3[j3][i3] = buff3;

}

}

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

for(int[]x:arr3){

for(int z:x){

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

} System.out.println();

}

Scanner sc4 = new Scanner(System.in);

System.out.println("Enter shift up");

int step4 = sc.nextInt();

if (step4 > arr4.length || step4 < 1) {

System.out.println("Shift impossible");

return;

}

int buff4, i4, j4;

for (int r = 0; r < step4; r++) {

for (i4 = 0; i4 < arr4.length; i4++) {

buff4 = arr4[0][i4];

for (j4 = 0; j4 < arr4.length - 1; j4++) {

arr4[j4][i4] = arr4[j4 + 1][i4];

}

arr4[j4][i4] = buff4;

}

}

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

for(int[]x:arr4){

for(int z:x){

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

} System.out.println();

}

}

}

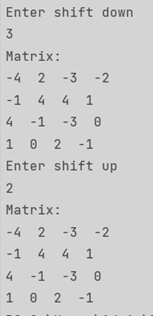
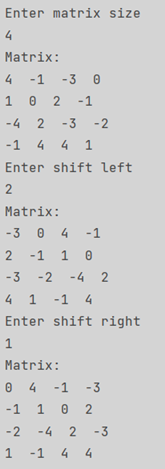


Рисунок 2 – Результат работы программы

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

import java.util.Arrays;

import java.util.Random;

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner in = new Scanner(System.in);

System.out.println("Enter matrix size");

int size = in.nextInt();

int arr[][] = new int[size][size];

Random random = new Random();

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

for (int j = 0; j < arr.length; j++) {

arr[j][i] = random.nextInt(2 \* size + 1) - size;

}

}

int[] mas = new int[arr.length \* arr.length]; // матрицу в одномерный массив

int inc=0, max=0, h=0, decr=0, min=0, h1=0, u=0;

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

for(int[]x:arr){

for(int z:x){

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

mas[u++] =z;

}

System.out.println();

}

for (int j=0; j<mas.length-1; j++){

if(mas[j] < mas[j+1]){ //если элемент больше предыдущего

inc+=1; //количество возрастаний

if(inc>max) {

max=inc;

h=j+1;} //индекс последнего возрастающего элемента

}

else{inc=0;} //возрастание прервалась

if(mas[j] > mas[j+1]){ //если элемент меньше предыдущего

decr+=1; //количество убываний

if(decr>min){

min=decr;

h1=j+1; // индекс последнего убывающего элемента

}

} else{decr=0;}// убывание прервалась

}

System.out.println("Maximum sequence of increasing numbers:"); //возрастающая последовательность

int[]m\_inc=new int[max+1];

System.arraycopy(mas,(h-max),m\_inc,0,(max+1)); //копирование последовательности в новый массив

System.out.println(Arrays.toString(m\_inc)+" "+(max+1)); // вывод массива возрастающей последовательности

System.out.println("Maximum sequence of decreasing numbers:"); // убывающая

int[]m\_decr=new int[min+1];

System.arraycopy(mas,(h1-min),m\_decr,0,(min+1)); //копирование последовательности в новый массив

System.out.println(Arrays.toString(m\_decr)+" "+(min+1)); // вывод массива убывающей последовательности

}

}

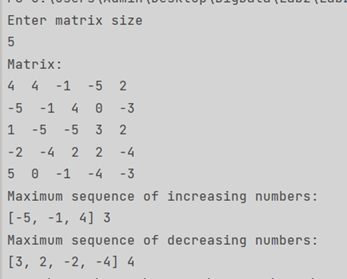


Рисунок 3 – Результат работы программы

**Вывод:** приобретен навык работы с арифметическими операциями

**Ссылка на репозиторий с программами:** [**https://github.com/nargi3/BigData**](https://github.com/nargi3/BigData)