|  |  |
| --- | --- |
| 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 № 3**

**Вариант № 13**

**Название:** классы, наследование и полиморфизм

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

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

Москва, 2023

**Цель:** освоить принципы ООП на языке программирования Java.

**Задание 1:** определить класс Вектор в R3. Реализовать методы для проверки векторов на ортогональность, проверки пересечения не ортогональных векторов, сравнения векторов. Создать массив из m объектов. Определить, какие из векторов компланарны.

Код класса Main:

package Zadanie\_1;

import java.util.Scanner;

public class Main{

public static void main(String[] args) {

System.out.println("Сколько векторов будет вводится?");

Scanner console = new Scanner(System.in);

int col = console.nextInt();

Vector[] vectors = new Vector[col];

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

System.out.println("Введите Х1: ");

int x1 = console.nextInt();

System.out.println("Введите Х1: ");

int x2 = console.nextInt();

System.out.println("Введите Y1: ");

int y1 = console.nextInt();

System.out.println("Введите Y2: ");

int y2 = console.nextInt();

System.out.println("Введите Z1: ");

int z1 = console.nextInt();

System.out.println("Введите Z2: ");

int z2 = console.nextInt();

vectors[i] = new Vector(x1, y1, z1, x2, y2, z2);

}

int number\_one;

int number\_two;

System.out.println("номер вектора 1: ");

number\_one = console.nextInt();

System.out.println("номер вектора 2: ");

number\_two = console.nextInt();

int x11 = vectors[number\_one].getX1();

int y11 = vectors[number\_one].getY1();

int z11 = vectors[number\_one].getZ1();

int x21 = vectors[number\_two].getX2();

int y21 = vectors[number\_two].getY2();

int z21 = vectors[number\_two].getZ2();

int x12 = vectors[number\_one].getX1();

int y12 = vectors[number\_one].getY1();

int z12 = vectors[number\_one].getZ1();

int x22 = vectors[number\_two].getX2();

int y22 = vectors[number\_two].getY2();

int z22 = vectors[number\_two].getZ2();

System.out.print("Комплонарны ли вектора? - ");

System.out.println(vectors[number\_one].ort(x11, x12, y11, y12, z11,

z12, x21, x22, y21, y22, z21, z22));

if(!vectors[number\_one].ort(x11, x12, y11, y12, z11, z12,

x21, x22, y21, y22, z21, z22)){

System.out.print("пересекаются ли вектора? - ");

System.out.println(vectors[number\_one].per(x11, x12, y11, y12, z11, z12,

x21, x22, y21, y22, z21, z22));

}

System.out.println("вектора одинаковой длинны? - ");

System.out.println(vectors[number\_one].sravn(x11, x12, y11, y12, z11, z12,

x21, x22, y21, y22, z21, z22));

}

}

Код класса Vector:

package Zadanie\_1;

import static java.lang.Math.pow;

import static java.lang.Math.sqrt;

public class Vector {

private int x1;

private int x2;

private int y1;

private int y2;

private int z1;

private int z2;

public Vector(int x1, int y1, int z1, int x2, int y2, int z2){

this.x1 = x1;

this.y1 = y1;

this.z1 = z1;

this.x2 = x2;

this.y2 = y2;

this.z2 = z2;

}

public int getX1() {

return x1;

}

public int getY1() {

return y1;

}

public int getZ1() {

return z1;

}

public int getX2() {

return x1;

}

public int getY2() {

return y1;

}

public int getZ2() {

return z1;

}

public boolean ort(int x11, int x12,

int y11, int y12,

int z11, int z12,

int x21, int x22,

int y21, int y22,

int z21, int z22){

int sum = x11 \* x12 + y11 \* y12 + z11 \* z12

\* x21 \* x22 + y21 \* y22 + z21 \* z22;

if (sum == 0){

return true;

}

else {

return false;

}

}

public boolean per(int x11, int x12,

int y11, int y12,

int z11, int z12,

int x21, int x22,

int y21, int y22,

int z21, int z22){

int v1; int v2; int v3; int v4;

v1 = vektorMulti(x22 - x21, y22 - y21, x11 - x21, y11 - y21, z11 - z21, z11 - z21);

v2 = vektorMulti(x22 - x21, y22 - y21, x12 - x21, y12 - y21, z12 - z21, z12 - z21);

v3 = vektorMulti(x12 - x11, y21 - y11, x21 - x11, y21 - y21, z21 - z11, z21 - z11);

v4 = vektorMulti(x12 - x11, y21 - y11, x22 - x11, y22 - y21, z22 - z11, z22 - z11);

if (v1 \* v2 < 0 && v3 \* v4 < 0){

return true;

}

else {

return false;

}

}

public int vektorMulti(int ax, int ay,

int az, int bx,

int by, int bz){

return ax\*by - bx\*ay;

}

public boolean sravn(int x11, int x12,

int y11, int y12,

int z11, int z12,

int x21, int x22,

int y21, int y22,

int z21, int z22){

double mod\_one = sqrt(pow(x11, 2) + pow(x12, 2) + pow(y11, 2)

+ pow(y12, 2) + pow(z12, 2) + pow(z12, 2));

double mod\_two = sqrt(pow(x21, 2) + pow(x22, 2) + pow(y21, 2)

+ pow(y22, 2) + pow(z22, 2) + pow(z22, 2));

if(mod\_one == mod\_two){

return true;

}

else{

return false;

}

}

}

Работа программы показана на рисунке 1.

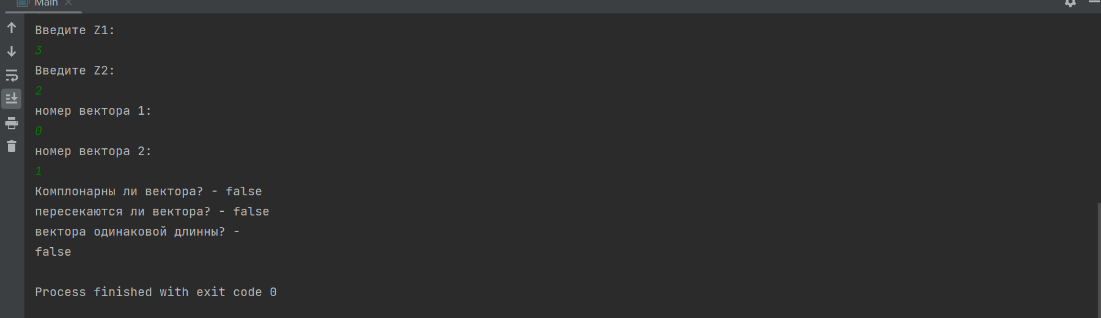


Рисунок 1 – Работа программы

**Задание 2:** определить класс Матрица размерности (n x n). Класс должен содержать несколько конструкторов. Реализовать методы для сложения, вычитания, умножения матриц. Объявить массив объектов. Создать методы, вычисляющие первую и вторую нормы матрицы

Определить, какая из матриц имеет наименьшую первую и вторую нормы.

Код класса Main:

package Zadanie\_2;

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

System.out.println("Сколько матриц будет вводится?");

Scanner console = new Scanner(System.in);

int col = console.nextInt();

Matrix[] matrices = new Matrix[col];

System.out.println("Размер матриц: ");

int n1 = console.nextInt();

System.out.println("Диапазон случайных чисел: ");

int n2 = console.nextInt();

int n3 = console.nextInt();

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

matrices[i] = new Matrix(n1, n2, n3);

}

int number\_one;

int number\_two;

System.out.println("С какими матрицами будут производится вычисления");

System.out.println("Матрица 1: ");

number\_one = console.nextInt();

System.out.println("Матрица 2: ");

number\_two = console.nextInt();

operation operations = new operation();

matrices[number\_one].zapolnenie();

matrices[number\_two].zapolnenie();

double[] Matrix\_one = matrices[number\_one].oneArray;

double[] Matrix\_two = matrices[number\_two].oneArray;

double[][] Matrix\_one\_one = matrices[number\_one].twoArray;

double[][] Matrix\_two\_two = matrices[number\_two].twoArray;

System.out.println("Сложение: ");

operations.sum(Matrix\_one, Matrix\_two, n1);

System.out.println("Вычитания: ");

operations.vuch(Matrix\_one, Matrix\_two, n1);

System.out.println("умножение: ");

operations.umnoj(Matrix\_one\_one, Matrix\_two\_two, n1);

System.out.println("У какой матрицы считать норму: ");

int norma = console.nextInt();

double[][] norma\_matrix = matrices[norma].twoArray;

System.out.println("Первая норма матрицы: ");

System.out.println(operations.one\_norma(n1, norma\_matrix));

System.out.println("Вторая норма матрицы: ");

System.out.println(operations.two\_norma(n1, norma\_matrix));

double max\_norma\_one = -1000;

double max\_norma\_two = -1000;

for (int i = 0; i < n1; i++) {

if (max\_norma\_one < operations.one\_norma(n1, matrices[i].twoArray)){

max\_norma\_one = operations.one\_norma(n1, matrices[i].twoArray);

}

}

for (int i = 0; i < n1; i++) {

if (max\_norma\_two < operations.two\_norma(n1, matrices[i].twoArray)){

max\_norma\_two = operations.two\_norma(n1, matrices[i].twoArray);

}

}

}

}

Код класса Matrix:

package Zadanie\_2;

public class Matrix {

private int n1;

private int n2;

private int n3;

public Matrix(int n1, int n2, int n3) {

this.n1 = n1;

this.n2 = n2;

this.n3 = n3;

}

public int getN1() {

return n1;

}

public void setN1(int n1) {

this.n1 = n1;

}

public int getN2() {

return n2;

}

public void setN2(int n2) {

this.n2 = n2;

}

public int getN3() {

return n3;

}

public void setN3(int n3) {

this.n3 = n3;

}

public double[][] twoArray;

public double[] oneArray;

public void zapolnenie(){

twoArray = new double[n1][n1];

for (int i = 0; i < n1; i++) {

for (int j = 0; j < n1; j++) {

twoArray[i][j] = (Math.random() \* ((n3 - n2))) + n2;

System.out.print(twoArray[i][j] + " ");

}

System.out.println();

}

int one\_i = 0;

oneArray = new double[n1 \* n1];

for (int i = 0; i < n1; i++) {

for (int j = 0; j < n1; j++) {

oneArray[one\_i] = twoArray[i][j];

one\_i++;

}

}

}

}

Код класса Operation:

package Zadanie\_2;

public class operation {

public void sum(double[] martix\_one, double[] matrix\_two, int n1){

double[] rez = new double[martix\_one.length];

for (int i = 0; i < martix\_one.length; i++){

rez[i] = martix\_one[i] + matrix\_two[i];

}

for (int i = 0; i < n1; i++) {

for (int j = 0; j < n1; j++) {

System.out.print(rez[i] + " ");

}

System.out.println();

}

}

public void vuch(double[] martix\_one, double[] matrix\_two, int n1){

double[] rez = new double[matrix\_two.length];

for (int i = 0; i < martix\_one.length; i++){

rez[i] = martix\_one[i] - matrix\_two[i];

}

for (int i = 0; i < n1; i++) {

for (int j = 0; j < n1; j++) {

System.out.print(rez[i] + " ");

}

System.out.println();

}

}

public void umnoj(double[][] martix\_one, double[][] matrix\_two, int n1){

double[][] rez = new double[n1][n1];

for (int i = 0; i < martix\_one.length; i++) {

for (int j = 0; j < matrix\_two[0].length; j++) {

for (int k = 0; k < matrix\_two.length; k++) {

rez[i][j] += martix\_one[i][k] \* matrix\_two[k][j];

}

}

}

for (int i = 0; i < n1; i++) {

for (int j = 0; j < n1; j++) {

System.out.print(rez[i][j] + " ");

}

System.out.println();

}

}

public double one\_norma(int n1, double[][] twoArray){

double[] str = new double[n1];

for (int i = 0; i < n1; i++){

for (int j = 0; j < n1; j++) {

str[i] += twoArray[i][j];

}

}

double max\_i = -10000;

for (int i = 0; i < n1; i++) {

if(str[i] > max\_i){

max\_i = str[i];

}

}

return max\_i;

}

public double two\_norma(int n1, double[][] twoArray){

double[] str = new double[n1];

for (int i = 0; i < n1; i++){

for (int j = 0; j < n1; j++) {

str[i] += twoArray[j][i];

}

}

double max\_j = -10000;

for (int i = 0; i < n1; i++) {

if(str[i] > max\_j){

max\_j = str[i];

}

}

return max\_j;

}

}

Работа программы показана на рисунке 2.

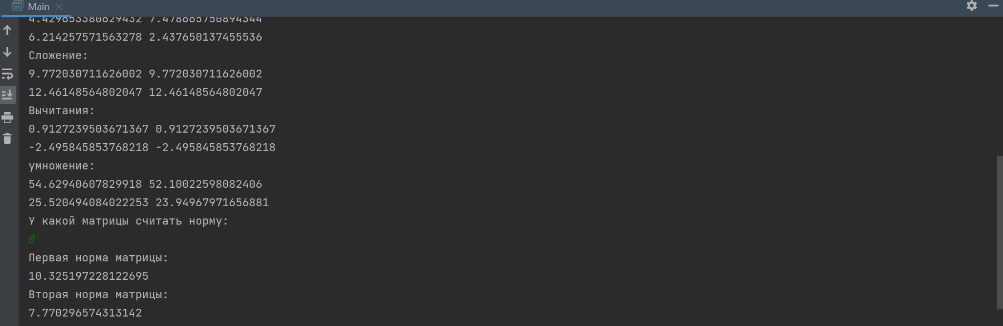


Рисунок 2 – Работа программы

**Задание 3:** создать классы, спецификации которых приведены ниже. Определить конструкторы и методы setТип(), getТип(), toString(). Определить дополнительно методы в классе, создающем массив объектов. Задать критерий выбора данных и вывести эти данные на консоль. Patient: id, Фамилия, Имя, Отчество, Адрес, Телефон, Номер медицинской карты, Диагноз. Создать массив объектов. Вывести: a) список пациентов, имеющих данный диагноз; b) список пациентов, номер медицинской карты у которых находится в заданном интервале.

Код класса Main:

package Zadanie\_3;

import java.util.Scanner;

import static java.util.function.Predicate.isEqual;

public class Main {

public static void main(String[] args) {

Patient[] patients = new Patient[100];

Scanner console = new Scanner(System.in);

String first\_name;

String second\_name;

String third\_name;

String address;

String diagnoz\_in;

System.out.println("Сколько будет пациентов: ");

int col = console.nextInt();

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

System.out.println("Введите имя: ");

first\_name = console.next();

System.out.println("Введите фамилию: ");

second\_name = console.next();

System.out.println("Введите отчество: ");

third\_name = console.next();

System.out.println("Введите адрес: ");

address = console.next();

System.out.println("Введите диагноз: ");

diagnoz\_in = console.next();

patients[i] = new Patient(i, first\_name, second\_name, third\_name,

address, i, i, diagnoz\_in);

}

System.out.println("Введите диагноз, чтобы поциентов с ним: ");

String diagnoz = console.next();

System.out.println("Пациенты с таким диагнозом: ");

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

if(patients[i].getDiagnoz().equals(diagnoz)){

System.out.println(patients[i].getFirst\_name() + " " + patients[i].getSecond\_name());

}

}

System.out.println("Введите диапазон значений: ");

int n1 = console.nextInt();

int n2 = console.nextInt();

System.out.println("Пациенты с номером медицинской карты, попадающими в этот диапазон: ");

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

if(patients[i].getNumber\_medic() > n1 && patients[i].getNumber\_medic() < n2){

System.out.println(patients[i].getFirst\_name() + " " + patients[i].getSecond\_name());

}

}

}

}

Код класса Patient:

package Zadanie\_3;

public class Patient {

private int id;

private String first\_name;

private String second\_name;

private String third\_name;

private String address;

private int number\_phone;

private int number\_medic;

private String diagnoz;

@Override

public String toString() {

return "Patient{" +

"number\_phone=" + number\_phone +

", number\_medic=" + number\_medic +

'}';

}

public Patient(int id, String first\_name, String second\_name, String third\_name, String address, int number\_phone, int number\_medic, String diagnoz) {

this.id = id;

this.first\_name = first\_name;

this.second\_name = second\_name;

this.third\_name = third\_name;

this.address = address;

this.number\_phone = number\_phone;

this.number\_medic = number\_medic;

this.diagnoz = diagnoz;

}

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getFirst\_name() {

return first\_name;

}

public void setFirst\_name(String first\_name) {

this.first\_name = first\_name;

}

public String getSecond\_name() {

return second\_name;

}

public void setSecond\_name(String second\_name) {

this.second\_name = second\_name;

}

public String getThird\_name() {

return third\_name;

}

public void setThird\_name(String third\_name) {

this.third\_name = third\_name;

}

public String getAddress() {

return address;

}

public void setAddress(String address) {

this.address = address;

}

public int getNumber\_phone() {

return number\_phone;

}

public void setNumber\_phone(int number\_phone) {

this.number\_phone = number\_phone;

}

public int getNumber\_medic() {

return number\_medic;

}

public void setNumber\_medic(int number\_medic) {

this.number\_medic = number\_medic;

}

public String getDiagnoz() {

return diagnoz;

}

public void setDiagnoz(String diagnoz) {

this.diagnoz = diagnoz;

}

}

Работа программы показана на рисунке 3.

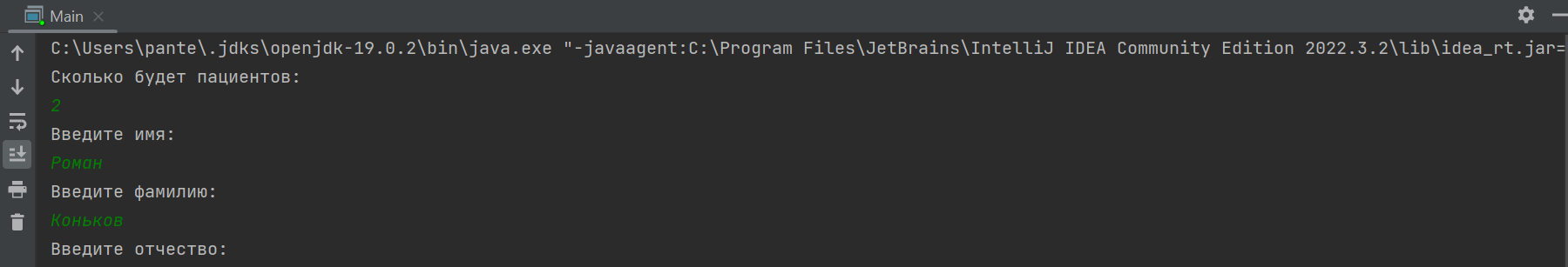


Рисунок 3 – Работа программы

**Задание 4:** создать классы, спецификации которых приведены ниже. Определить конструкторы и методы setТип(), getТип(), toString(). Определить дополнительно методы в классе, создающем массив объектов. Задать критерий выбора данных и вывести эти данные на консоль. Abiturient: id, Фамилия, Имя, Отчество, Адрес, Телефон, Оценки. Создать массив объектов. Вывести: a) список абитуриентов, имеющих неудовлетворительные оценки; b) список абитуриентов, средний балл у которых выше заданного; c) выбрать заданное число n абитуриентов, имеющих самый высокий средний балл (вывести также полный список абитуриентов, имеющих полупроходной балл).

Код класса Main:

package Zadanie\_4;

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Abiturient[] abiturients = new Abiturient[100];

Scanner console = new Scanner(System.in);

String first\_name;

String second\_name;

String third\_name;

String address;

int[] ocenki = new int[100];

System.out.println("Сколько будет аббитуриентов: ");

int col = console.nextInt();

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

System.out.println("Введите имя: ");

first\_name = console.next();

System.out.println("Введите фамилию: ");

second\_name = console.next();

System.out.println("Введите отчество: ");

third\_name = console.next();

System.out.println("Введите адрес: ");

address = console.next();

System.out.println("Сколько оценок: ");

int kol\_oc = console.nextInt();

for (int j = 0; j < kol\_oc; j++) {

ocenki[i] = console.nextInt();

}

abiturients[i] = new Abiturient(i, first\_name, second\_name,

third\_name, address, i, ocenki);

}

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

for (int j = 0; j < abiturients[i].getOcenki().length; j++) {

if (abiturients[i].getOcenki().length == 2){

System.out.println(abiturients[i].getFirst\_name() + " " + abiturients[i].getSecond\_name());

break;

}

}

}

System.out.println("Средний балл: ");

int z\_sr\_ball = console.nextInt();

double sr\_ball = 0;

int[] mass\_ocenki;

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

mass\_ocenki = abiturients[i].getOcenki();

for (int j = 0; j < abiturients[i].getOcenki().length; j++) {

sr\_ball += mass\_ocenki[i];

}

sr\_ball /= abiturients[i].getOcenki().length;

if (sr\_ball < z\_sr\_ball){

System.out.println(abiturients[i].getFirst\_name() + " " + abiturients[i].getSecond\_name());

}

sr\_ball = 0;

}

System.out.println("Высокий балл: ");

int viss\_ball = console.nextInt();

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

mass\_ocenki = abiturients[i].getOcenki();

for (int j = 0; j < abiturients[i].getOcenki().length; j++) {

sr\_ball += mass\_ocenki[i];

}

sr\_ball /= abiturients[i].getOcenki().length;

if (sr\_ball > viss\_ball){

System.out.println(abiturients[i].getFirst\_name() + " " + abiturients[i].getSecond\_name());

}

sr\_ball = 0;

}

System.out.println("Проходной балл: ");

double pr\_ball = console.nextInt();

System.out.println("Имеют полупроходной балл: ");

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

mass\_ocenki = abiturients[i].getOcenki();

for (int j = 0; j < abiturients[i].getOcenki().length; j++) {

sr\_ball += mass\_ocenki[i];

}

sr\_ball /= abiturients[i].getOcenki().length;

if (sr\_ball > pr\_ball/2){

System.out.println(abiturients[i].getFirst\_name() + " " + abiturients[i].getSecond\_name());

}

sr\_ball = 0;

}

}

}

Код класса Abiturient:

package Zadanie\_4;

import java.util.Arrays;

public class Abiturient {

public int id;

public String first\_name;

public String second\_name;

public String third\_name;

public String address;

public int number\_phone;

public int[] ocenki;

@Override

public String toString() {

return "Abiturient{" +

"id=" + id +

", first\_name='" + first\_name + '\'' +

", second\_name='" + second\_name + '\'' +

", third\_name='" + third\_name + '\'' +

", address='" + address + '\'' +

", number\_phone=" + number\_phone +

", ocenki=" + Arrays.toString(ocenki) +

'}';

}

public Abiturient(int id, String first\_name, String second\_name, String third\_name, String address, int number\_phone, int[] ocenki) {

this.id = id;

this.first\_name = first\_name;

this.second\_name = second\_name;

this.third\_name = third\_name;

this.address = address;

this.number\_phone = number\_phone;

this.ocenki = ocenki;

}

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getFirst\_name() {

return first\_name;

}

public void setFirst\_name(String first\_name) {

this.first\_name = first\_name;

}

public String getSecond\_name() {

return second\_name;

}

public void setSecond\_name(String second\_name) {

this.second\_name = second\_name;

}

public String getThird\_name() {

return third\_name;

}

public void setThird\_name(String third\_name) {

this.third\_name = third\_name;

}

public String getAddress() {

return address;

}

public void setAddress(String address) {

this.address = address;

}

public int getNumber\_phone() {

return number\_phone;

}

public void setNumber\_phone(int number\_phone) {

this.number\_phone = number\_phone;

}

public int[] getOcenki() {

return ocenki;

}

public void setOcenki(int[] ocenki) {

this.ocenki = ocenki;

}

}

Работа программы показана на рисунке 4.

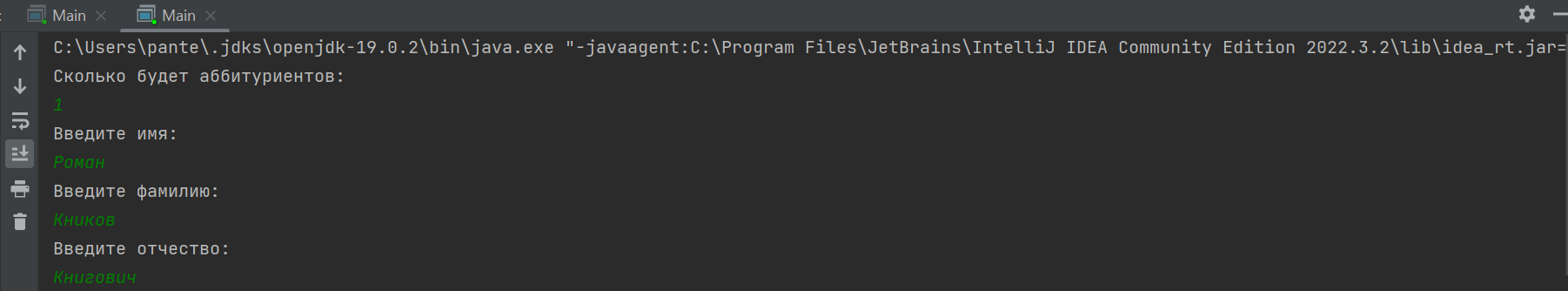


Рисунок 4 – Работа программы

**Задание 5:** создать приложение, удовлетворяющее требованиям, приведенным в задании. Аргументировать принадлежность классу каждого создаваемого метода и корректно переопределить для каждого класса методы equals(), hashCode(), toString(). Создать объект класса Простая дробь, используя класс Число. Методы: вывод на экран, сложение, вычитание, умножение, деление.

Код класса Main:

package Zadanie\_5;

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner console = new Scanner(System.in);

System.out.println("Введите числитель 1: ");

double ch11 = console.nextDouble();

System.out.println("Введите знаменатель 1: ");

double ch12 = console.nextDouble();

System.out.println("Введите числитель 2: ");

double ch21 = console.nextDouble();

System.out.println("Введите знаменатель 2: ");

double ch22 = console.nextDouble();

chislo chislos11 = new chislo(ch11);

chislo chislos12 = new chislo(ch12);

chislo chislos21 = new chislo(ch21);

chislo chislos22 = new chislo(ch22);

eze\_drob eze\_drob1 = new eze\_drob(chislos11, chislos12);

eze\_drob eze\_drob2 = new eze\_drob(chislos21, chislos22);

eze\_drob1.sloj(eze\_drob2);

eze\_drob1.vich(eze\_drob2);

eze\_drob1.del(eze\_drob2);

eze\_drob1.umnoj(eze\_drob2);

}

}

Код класса Chislo:

package Zadanie\_5;

import java.util.Objects;

public class chislo {

private double ch1;

public chislo(double ch1){

this.ch1 = ch1;

}

public double getCh1() {

return ch1;

}

public void setCh1(double ch1) {

this.ch1 = ch1;

}

@Override

public boolean equals(Object o) {

if (this == o) return true;

if (!(o instanceof chislo chislo)) return false;

return Double.compare(chislo.getCh1(), getCh1()) == 0;

}

@Override

public int hashCode() {

return Objects.hash(getCh1());

}

@Override

public String toString() {

return "chislo{" +

"ch1=" + ch1 +

'}';

}

}

Код класса Eze\_drob:

package Zadanie\_5;

import java.util.Objects;

public class eze\_drob {

private chislo chisl;

private chislo znam;

protected void sloj(eze\_drob eze\_drob2){

double chislitel = chisl.getCh1() \* eze\_drob2.znam.getCh1() + znam.getCh1() \* eze\_drob2.chisl.getCh1();

double znamen = znam.getCh1() \* eze\_drob2.znam.getCh1();

System.out.println("Сложение: ");

System.out.println("Числитель: " + chislitel);

System.out.println("Знаменатель: " + znamen);

}

protected void vich(eze\_drob eze\_drob2){

double chislitel = chisl.getCh1() \* eze\_drob2.znam.getCh1() - znam.getCh1() \* eze\_drob2.chisl.getCh1();

double znamen = znam.getCh1() \* eze\_drob2.znam.getCh1();

System.out.println("Вычитание: ");

System.out.println("Числитель: " + chislitel);

System.out.println("Знаменатель: " + znamen);

}

protected void umnoj(eze\_drob drob2){

double chislitel = chisl.getCh1() \* drob2.chisl.getCh1();

double znamen = znam.getCh1() \* drob2.znam.getCh1();

System.out.println("Умножение: ");

System.out.println("Числитель: " + chislitel);

System.out.println("Знаменатель: " + znamen);

}

protected void del(eze\_drob drob2){

double chislitel = chisl.getCh1() \* drob2.znam.getCh1();

double znamen = znam.getCh1() \* drob2.chisl.getCh1();

System.out.println("Умножение: ");

System.out.println("Числитель: " + chislitel);

System.out.println("Знаменатель: " + znamen);

}

public eze\_drob(chislo chisl, chislo znam) {

this.chisl = chisl;

this.znam = znam;

}

public chislo getChisl() {

return chisl;

}

public void setChisl(chislo chisl) {

this.chisl = chisl;

}

public chislo getZnam() {

return znam;

}

public void setZnam(chislo znam) {

this.znam = znam;

}

@Override

public boolean equals(Object o) {

if (this == o) return true;

if (!(o instanceof eze\_drob eze\_drob)) return false;

return Objects.equals(getChisl(), eze\_drob.getChisl()) && Objects.equals(getZnam(), eze\_drob.getZnam());

}

@Override

public int hashCode() {

return Objects.hash(getChisl(), getZnam());

}

@Override

public String toString() {

return "eze\_drob{" +

"chisl=" + chisl +

", znam=" + znam +

'}';

}

}

Работа программы показана на рисунке 5.

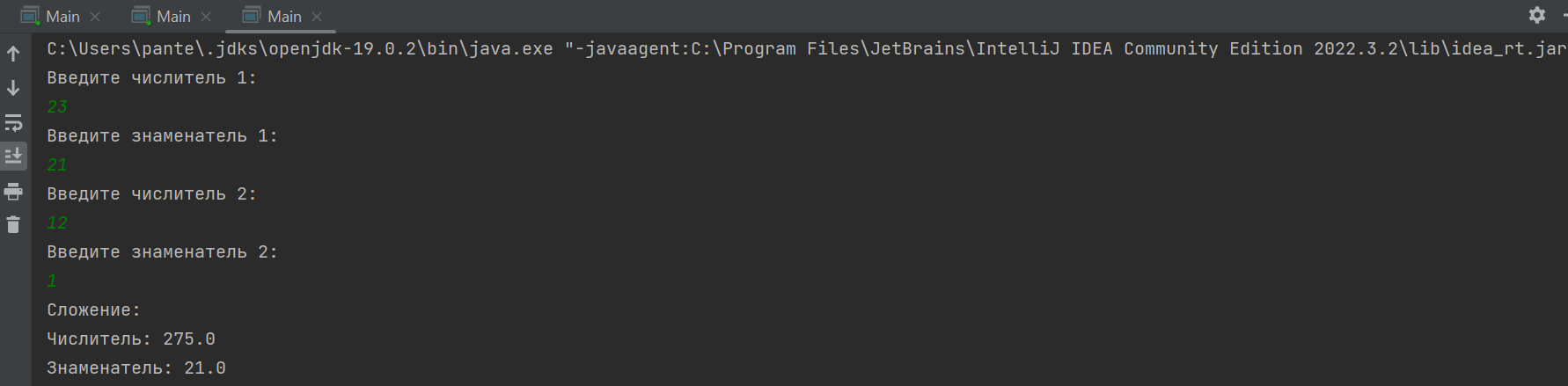


Рисунок 5 – Работа программы

**Задание 6:** создать приложение, удовлетворяющее требованиям, приведенным в задании. Аргументировать принадлежность классу каждого создаваемого метода и корректно переопределить для каждого класса методы equals(), hashCode(), toString(). Создать объект класса Дом, используя классы Окно, Дверь. Методы: закрыть на ключ, вывести на консоль количество окон, дверей.

Код класса Main:

package Zadanie\_6;

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner console = new Scanner(System.in);

System.out.println("сколько дома дверей?");

int n1 = console.nextInt();

door[] doors = new door[n1];

doors[0] = new door(false);

System.out.println("сколько дома окон?");

int n2 = console.nextInt();

window[] windows = new window[n2];

windows[0] = new window(false);

house houses = new house(windows, doors);

doors[0].lock\_on\_key();

if(doors[0].getLock()){

System.out.println("дверь закрыта");

} else {

System.out.println("дверь открыта");

}

windows[0].lock();

if(doors[0].getLock()){

System.out.println("окно закрыто");

} else {

System.out.println("окно открыто");

}

System.out.println("В этом доме столько дверей: " + houses.getDoors().length);

System.out.println("В этом доме столько окон: " + houses.getWindows().length);

}

}

Код класса House:

package Zadanie\_6;

public class house {

private window[] windows = new window[100];

private door[] doors = new door[100];

public house(window[] windows, door[] doors) {

this.windows = windows;

this.doors = doors;

}

public window[] getWindows() {

return windows;

}

public void setWindows(window[] windows) {

this.windows = windows;

}

public door[] getDoors() {

return doors;

}

public void setDoors(door[] doors) {

this.doors = doors;

}

}

Код класса Door:

package Zadanie\_6;

import java.util.Objects;

import java.util.Scanner;

public class door {

Scanner console = new Scanner(System.in);

private boolean lock;

public door(boolean lock) {

this.lock = lock;

}

public boolean lock\_on\_key(){

System.out.println("Что сделать с дверью?");

System.out.println("1 - открыть, 2 - закрыть на ключ");

String doing = console.next();

if (doing.equals("1")){

lock = true;

} else {

lock = false;

}

return lock;

}

public boolean getLock() {

return lock;

}

public void setLock(boolean lock) {

this.lock = lock;

}

@Override

public String toString() {

return "door{" +

"lock=" + lock +

'}';

}

@Override

public boolean equals(Object o) {

if (this == o) return true;

if (!(o instanceof door door)) return false;

return getLock() == door.getLock();

}

@Override

public int hashCode() {

return Objects.hash(getLock());

}

}

Код класса Window:

package Zadanie\_6;

import java.util.Objects;

import java.util.Scanner;

public class window {

Scanner console = new Scanner(System.in);

private boolean lock;

public window(boolean lock) {

this.lock = lock;

}

public boolean lock(){

System.out.println("Что сделать с окном?");

System.out.println("1 - открыть, 2 - закрыть");

String doing = console.next();

if (doing.equals("1")){

lock = true;

} else {

lock = false;

}

return lock;

}

@Override

public boolean equals(Object o) {

if (this == o) return true;

if (!(o instanceof window window)) return false;

return lock == window.lock;

}

@Override

public int hashCode() {

return Objects.hash(lock);

}

@Override

public String toString() {

return "window{" +

"lock=" + lock +

'}';

}

}

Работа программы показана на рисунке 6.

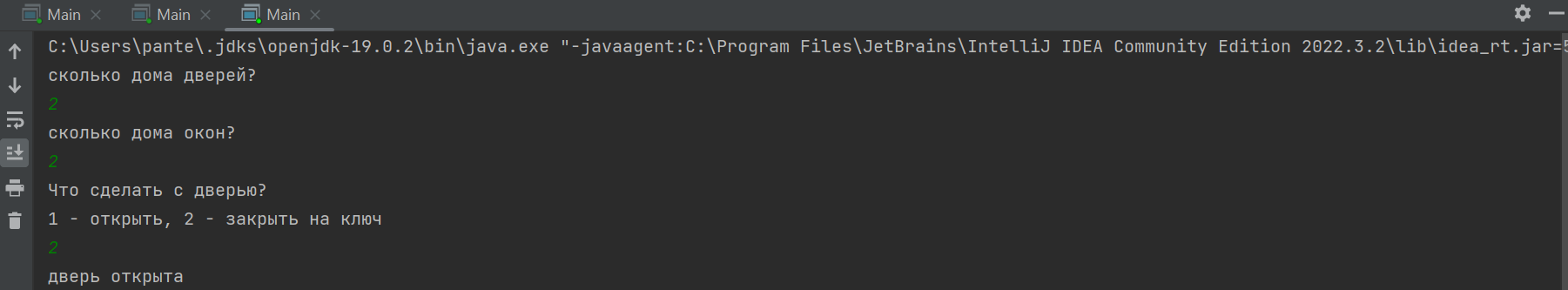


Рисунок 6 – Работа программы

**Задание 7:** построить модель программной системы. Система Больница. Пациенту назначается лечащий Врач. Врач может сделать назначение Пациенту (процедуры, лекарства, операции). Медсестра или другой Врач выполняют назначение. Пациент может быть выписан из Больницы по окончании лечения, при нарушении режима или при иных обстоятельствах.

Код класса Main:

package Zadanie\_7;

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner console = new Scanner(System.in);

Pacient pacients1 = new Pacient( "Роман", "Тетраков",

"Таблеткович", 1, "Шиза", 1, false, false);

lech\_vrach lechVrach1 = new lech\_vrach (1, "Роман", "Тетраков",

"Таблеткович", "от шизи", "пить таблетки");

med\_sister medSister1 = new med\_sister("Мария", "Кариева",

"Оранжевна", "Тетраков");

Pacient[] pacients = new Pacient[100];

lech\_vrach[] lechVrach = new lech\_vrach[100];

med\_sister[] medSister = new med\_sister[100];

int id\_cart = 2;

int id\_vrach = 2;

int id\_vracha = 0;

int id\_pacienta = 0;

int id\_sister = 0;

System.out.println("С кем будет работа?");

System.out.println("1 - пациент");

System.out.println("2 - врач");

System.out.println("3 - медсестра");

int work = console.nextInt();

if(work == 1){

System.out.println("Пациент");

System.out.println("добавить пациента?");

String otv = console.next();

if(otv.equals("да")){

String first\_name = console.next();

String second\_name = console.next();

String third\_name = console.next();

String lechnie = console.next();

int vrach = console.nextInt();

id\_cart++;

pacients[id\_pacienta] = new Pacient(first\_name, second\_name, third\_name, id\_cart,

lechnie, vrach, false, false);

id\_pacienta++;

}

System.out.println("с каким пациоентом рабоать?");

int id\_cart\_pac = console.nextInt();

System.out.println("Назначить врача?");

String da = console.next();

if(da.equals("да")){

pacients[id\_cart\_pac].setLech\_vrach\_naz(lechVrach[id\_vracha].getId());

}

if(pacients[id\_cart\_pac].getEnd\_lech() || pacients[id\_cart\_pac].getNarush()){

System.out.println("пациента выписан");

System.out.println("Причина");

if(pacients[id\_cart\_pac].getEnd\_lech()){

System.out.println("Закончено лечение");

} else {

System.out.println("Буянит");

}

}

} else if (work == 2) {

System.out.println("Врач");

System.out.println("добавить врача?");

String otv = console.next();

if(otv.equals("да")){

String first\_name = console.next();

String second\_name = console.next();

String third\_name = console.next();

String spec = console.next();

String lechnie = console.next();

id\_vrach++;

lechVrach[id\_vracha] = new lech\_vrach(id\_vrach, first\_name, second\_name, third\_name, spec, lechnie);

id\_vracha++;

}

System.out.println("с каким врачом рабоать?");

int id\_vrach\_work = console.nextInt();

System.out.println("какому пациенту назначать?");

int id\_cart\_pac = console.nextInt();

System.out.println("Назначить процедуры?");

String da = console.next();

if(da.equals("да")){

pacients[id\_cart\_pac].setProc(lechVrach[id\_vrach\_work].procedur());

} else {

System.out.println("Назначить лекарство?");

da = console.next();

if(da.equals("да")){

pacients[id\_cart\_pac].setProc(lechVrach[id\_vrach\_work].new\_lech());

} else {

System.out.println("Назначить операцию?");

da = console.next();

if(da.equals("да")){

pacients[id\_cart\_pac].setProc(lechVrach[id\_vrach\_work].operasion());

}

}

}

} else {

System.out.println("Медсестра");

System.out.println("добавить медсестру?");

String otv = console.next();

if(otv.equals("да")){

String first\_name = console.next();

String second\_name = console.next();

String third\_name = console.next();

String boss = console.next();

medSister[id\_sister] = new med\_sister(first\_name, second\_name, third\_name, boss);

id\_sister++;

}

System.out.println("с какой медсестрой рабоать?");

int id\_med\_pac = console.nextInt();

medSister[id\_med\_pac].naznach();

}

}

}

Код класса Lech\_vrach:

package Zadanie\_7;

import java.util.Scanner;

public class lech\_vrach {

Scanner console = new Scanner(System.in);

public int id;

public String first\_name;

public String second\_name;

public String third\_name;

public String specialnost;

public String lechenie;

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String new\_lech(){

System.out.println("Новое лечение");

String lech = console.next();

return lech;

}

public String operasion(){

System.out.println("Какую операцию назначить: ");

String op = console.next();

return op;

}

public String procedur(){

System.out.println("Какую процедуру назначить: ");

String pr = console.next();

return pr;

}

public lech\_vrach(int id, String first\_name, String second\_name, String third\_name,

String specialnost, String lechenie) {

this.id = id;

this.first\_name = first\_name;

this.second\_name = second\_name;

this.third\_name = third\_name;

this.specialnost = specialnost;

this.lechenie = lechenie;

}

public String getFirst\_name() {

return first\_name;

}

public void setFirst\_name(String first\_name) {

this.first\_name = first\_name;

}

public String getSecond\_name() {

return second\_name;

}

public void setSecond\_name(String second\_name) {

this.second\_name = second\_name;

}

public String getThird\_name() {

return third\_name;

}

public void setThird\_name(String third\_name) {

this.third\_name = third\_name;

}

public String getSpecialnost() {

return specialnost;

}

public void setSpecialnost(String specialnost) {

this.specialnost = specialnost;

}

public String getLechenie() {

return lechenie;

}

public void setLechenie(String lechenie) {

this.lechenie = lechenie;

}

}

Код класса Med\_sister:

package Zadanie\_7;

import java.util.Scanner;

public class med\_sister {

Scanner console = new Scanner(System.in);

public String first\_name;

public String second\_name;

public String third\_name;

public String boss;

public med\_sister(String first\_name, String second\_name, String third\_name, String boss) {

this.first\_name = first\_name;

this.second\_name = second\_name;

this.third\_name = third\_name;

this.boss = boss;

}

public void naznach(){

System.out.println("выполнить назначение");

}

public String getFirst\_name() {

return first\_name;

}

public void setFirst\_name(String first\_name) {

this.first\_name = first\_name;

}

public String getSecond\_name() {

return second\_name;

}

public void setSecond\_name(String second\_name) {

this.second\_name = second\_name;

}

public String getThird\_name() {

return third\_name;

}

public void setThird\_name(String third\_name) {

this.third\_name = third\_name;

}

public String getBoss() {

return boss;

}

public void setBoss(String boss) {

this.boss = boss;

}

}

Код класса Pacient:

package Zadanie\_7;

public class Pacient {

public String first\_name;

public String second\_name;

public String third\_name;

public int id\_medic\_cart;

public String lechenie;

public int vrach;

public boolean end\_lech;

public boolean narush;

public String oper;

public String proc;

public int lechushii;

public int lech\_vrach\_naz;

public boolean getEnd\_lech() {

return end\_lech;

}

public boolean getNarush() {

return narush;

}

public void setNarush(boolean narush) {

this.narush = narush;

}

public void setEnd\_lech(boolean end\_lech) {

this.end\_lech = end\_lech;

}

public int getLech\_vrach\_naz() {

return lech\_vrach\_naz;

}

public void setLech\_vrach\_naz(int lech\_vrach\_naz) {

this.lech\_vrach\_naz = lech\_vrach\_naz;

}

public int getLechushii() {

return lechushii;

}

public void setLechushii(int lechushii) {

this.lechushii = lechushii;

}

public String getOper() {

return oper;

}

public void setOper(String oper) {

this.oper = oper;

}

public String getProc() {

return proc;

}

public void setProc(String proc) {

this.proc = proc;

}

public Pacient(String first\_name, String second\_name, String third\_name,

int id\_medic\_cart, String lechenie, int vrach, boolean end\_lech, boolean narush) {

this.first\_name = first\_name;

this.second\_name = second\_name;

this.third\_name = third\_name;

this.id\_medic\_cart = id\_medic\_cart;

this.lechenie = lechenie;

this.vrach = vrach;

this.end\_lech = end\_lech;

this.narush = narush;

}

public String getFirst\_name() {

return first\_name;

}

public void setFirst\_name(String first\_name) {

this.first\_name = first\_name;

}

public String getSecond\_name() {

return second\_name;

}

public void setSecond\_name(String second\_name) {

this.second\_name = second\_name;

}

public String getThird\_name() {

return third\_name;

}

public void setThird\_name(String third\_name) {

this.third\_name = third\_name;

}

public int getId\_medic\_cart() {

return id\_medic\_cart;

}

public void setId\_medic\_cart(int id\_medic\_cart) {

this.id\_medic\_cart = id\_medic\_cart;

}

public String getLechenie() {

return lechenie;

}

public void setLechenie(String lechenie) {

this.lechenie = lechenie;

}

}

Работа программы показана на рисунке 7.

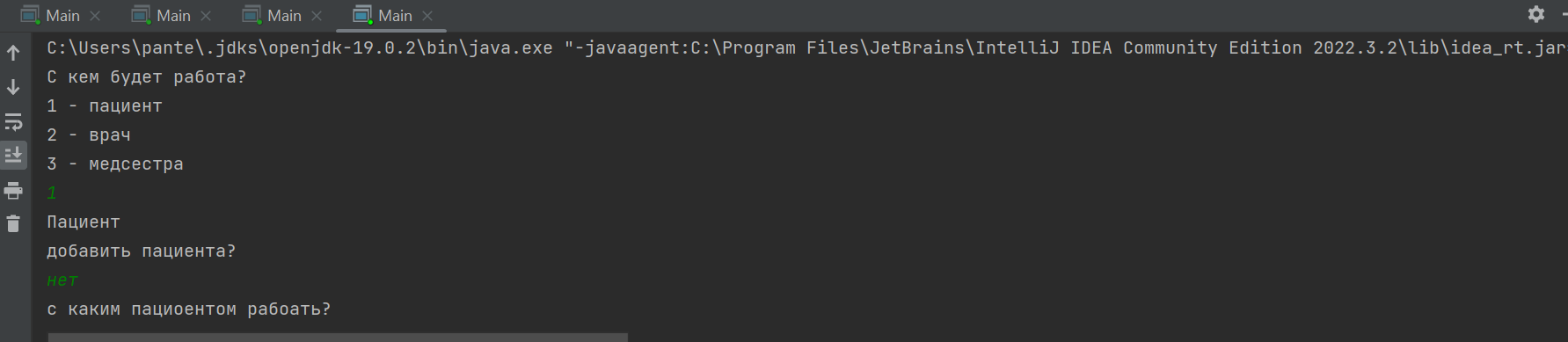


Рисунок 7 – Работа программы

**Задание 8:** построить модель программной системы. Система Вступительные экзамены. Абитуриент регистрируется на Факультет, сдает Экзамены. Преподаватель выставляет Оценку. Система подсчитывает средний балл и определяет Абитуриентов, зачисленных в учебное заведение.

Код класса Main:

package Zadanie\_8;

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner console = new Scanner(System.in);

abiturient[] abiturients = new abiturient[100];

facultet[] facultets = new facultet[10];

Prepodovatel[] prepodovatels = new Prepodovatel[40];

System.out.println("Сколько будет аббитуриентов: ");

int col1 = console.nextInt();

for (int i = 0; i < col1; i++) {

System.out.println("Введите имя: ");

String first\_name = console.next();

System.out.println("Введите фамилию: ");

String second\_name = console.next();

System.out.println("Введите отчество: ");

String third\_name = console.next();

abiturients[i] = new abiturient(i, first\_name, second\_name, third\_name);

}

System.out.println("Сколько будет учителей: ");

int col2 = console.nextInt();

for (int i = 0; i < col2; i++) {

String first\_name = console.next();

String second\_name = console.next();

String third\_name = console.next();

prepodovatels[i] = new Prepodovatel(i, first\_name, second\_name, third\_name);

}

System.out.println("с каким аббитуриентом работать?");

int id\_abb = console.nextInt();

System.out.println("на какой факультет поступает?");

int fak = console.nextInt();

abiturients[id\_abb].setFak(fak);

int n = 0;

System.out.println("с каким преподавателем работать?");

int id\_pr = console.nextInt();

if(abiturients[id\_abb].ekz()){

facultets[abiturients[id\_abb].fak].sr += prepodovatels[id\_pr].ocenka();

abiturients[id\_abb].setOcenka(prepodovatels[id\_pr].ocenka());

n++;

facultets[abiturients[id\_abb].fak].setN(n);

}

System.out.println("Средний бал какого факультета?");

int num\_fak = console.nextInt();

System.out.println(facultets[num\_fak].pr\_ball());

if (abiturients[id\_abb].getOcenka() > facultets[num\_fak].pr\_ball()){

System.out.println("аббитуриент прошел");

} else {

System.out.println("аббитуриент не прошел");

}

}

}

Код класса Abiturient:

package Zadanie\_8;

import java.util.Scanner;

public class abiturient {

Scanner console = new Scanner(System.in);

public int id;

public String first\_name;

public String second\_name;

public String third\_name;

public int fak;

public int ocenka;

public int getOcenka() {

return ocenka;

}

public void setOcenka(int ocenka) {

this.ocenka = ocenka;

}

public boolean ekz(){

boolean sdal;

System.out.println("Экзамен написан?");

String da = console.next();

if(da.equals("да")){

sdal = true;

} else {

sdal = false;

}

return sdal;

}

public int getFak() {

return fak;

}

public void setFak(int fak) {

this.fak = fak;

}

public abiturient(int id, String first\_name, String second\_name, String third\_name) {

this.id = id;

this.first\_name = first\_name;

this.second\_name = second\_name;

this.third\_name = third\_name;

}

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getFirst\_name() {

return first\_name;

}

public void setFirst\_name(String first\_name) {

this.first\_name = first\_name;

}

public String getSecond\_name() {

return second\_name;

}

public void setSecond\_name(String second\_name) {

this.second\_name = second\_name;

}

public String getThird\_name() {

return third\_name;

}

public void setThird\_name(String third\_name) {

this.third\_name = third\_name;

}

}

Код класса Facultet:

package Zadanie\_8;

public class facultet {

public int id;

public double sr;

public int n;

public String name;

public int getN() {

return n;

}

public void setN(int n) {

this.n = n;

}

public double pr\_ball(){

double ball = 0;

ball = sr / n;

return ball;

}

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public double getSr() {

return sr;

}

public void setSr(double sr) {

this.sr = sr;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

}

Код класса Prepodovatel:

package Zadanie\_8;

public class Prepodovatel {

public int id;

public String first\_name;

public String second\_name;

public String third\_name;

public int ocenka(){

int n2 = 5;

int n1 = 2;

int oc = (int) ((Math.random() \* ((n2 - n1))) + n1);

return oc;

}

public Prepodovatel(int id, String first\_name, String second\_name, String third\_name) {

this.id = id;

this.first\_name = first\_name;

this.second\_name = second\_name;

this.third\_name = third\_name;

}

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getFirst\_name() {

return first\_name;

}

public void setFirst\_name(String first\_name) {

this.first\_name = first\_name;

}

public String getSecond\_name() {

return second\_name;

}

public void setSecond\_name(String second\_name) {

this.second\_name = second\_name;

}

public String getThird\_name() {

return third\_name;

}

public void setThird\_name(String third\_name) {

this.third\_name = third\_name;

}

}

Работа программы показана на рисунке 8.

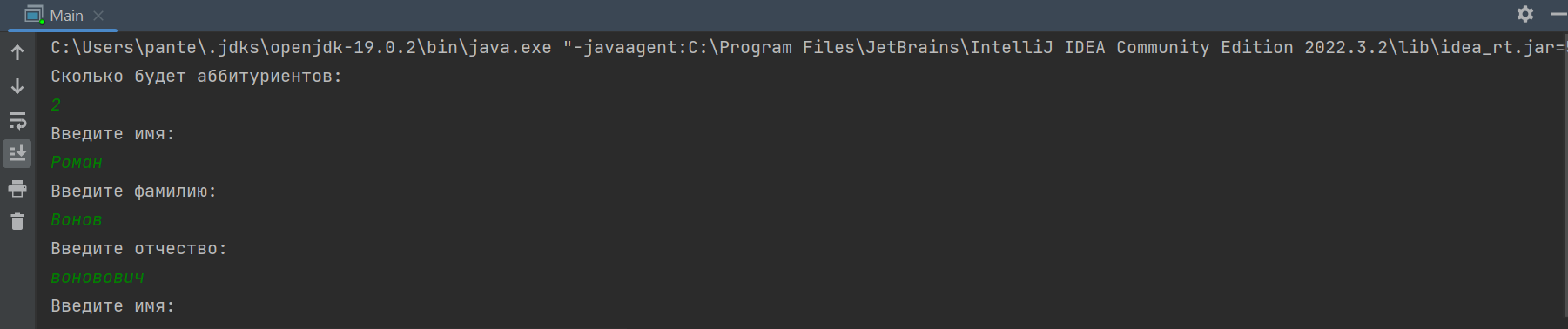


Рисунок 8 – Работа программы

**Вывод:** были освоены принципы ООП на языке программирования Java