

ЕКЗАМЕНАЦІЙНИЙ БІЛЕТ № 3

- (10 балів) а). Опишіть суть методу Терстоуна – методу отримання кількісних оцінок тверджень експерта. б). Сформулюйте властивості альтернатив, які домінують за Слейтером. с). Позиційні правила голосування: правило Хара.
- (15 балів) а). Обчисліть об'єднання бінарних відношень P та Q . б). Обчисліть дисперсію для розподілу Гаусса с). Знайти мажоранти для бінарного відношення Q .

$$P = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}, \quad Q = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

- (15 балів) а). Означення нечіткої множини б). Сформулювати властивості антирефлексивного бінарного відношення. с). Означення бінарного відношення.
- (10 балів) Написати програму, котра знаходить оптимальну альтернативу за методом ідеальної точки. Критерії якості задачі оптимізації задані наступним чином:

$$Q_1(x_1, x_2) = 2x_1 + x_2^2 \Rightarrow \max,$$

$$Q_2(x_1, x_2) = -2x_1^2 + x_2 \Rightarrow \max.$$

Верхов А. Фе/42 ТПР

Екзменаційний білет №3

② а) Об'єднання бінарних відношень:

$$P = \begin{vmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{vmatrix} \quad Q = \begin{vmatrix} 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{vmatrix}$$

$$P \cup Q = \begin{vmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{vmatrix}$$

В) ~~Рассчитать~~ Рассчитать дисперсию для плотности Гаусса

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

$$\mu_f = \int_{-\infty}^{\infty} x f(x) dx = a$$

$$D_f = \int_{-\infty}^{\infty} (x - \mu_f)^2 f(x) dx = a^2$$

с) Найти миноранты для априорного распределения Q

$$Q = \begin{vmatrix} 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{vmatrix}$$

$$\bar{Q} = \begin{vmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \end{vmatrix}$$

$$\textcircled{3} \textcircled{1} \textcircled{0} \textcircled{2}$$

Миноранта : 3

1) а) Опишіть суть методу Терстоуна — методу
прийомна к-сний оцінок твердження

Метод Терстоуна — метод, що дозволяє оцінити
випрати зміни в шкалі інтервалів
психологічних: суїцідальних характеристик

Спосіб розробляють певні твердження,
дані відомі експертами, що дозволяє
визначити к-мі значення для кожного
твердження

Терстоун розділив всю суму позитивно-негативного
ставлення на 11 категорій, розділених рівними
інтервалами

б) Формулюйте властивості альтернатив, які
домінують за Свейтером

Альтернатива може вважатися домінуючою за
Свейтером, коли не існує іншої
альтернативи для якої:

$$f_i(x) \geq f_i(x_0), i \in I_1$$

$$f_i(x) < f_i(x_0), i \in I_2$$

с) Полиційні правила голосування: правила Хара

Спочатку шукається варіант, що набув більше 50 % перших місць в профілі у всіх \bar{P} . Якщо такий варіант знайдено, то даний варіант вважається колективним рішенням. В іншому випадку видається найменший варіант, і процедура знову починається.

③ а) Означення нечіткої множини:

Нечітка множина — це множина, що може набувати значень в інтервалі $[0, 1]$, а не тільки 0 чи 1.

б) Сформулювати властивості антирефлексивного бінарного відношення

Антирефлексивне відношення — це таке бінарне відношення, у якому жоден елемент не співвідноситься із собою

$$\forall a \in X, \neg (aRa)$$

в) Означення бінарного відношення:

Бінарне відношення — окремий випадок відношення заданого на множині M , яке встановлюється між двома елементами множини

④ Алгоритм:

Спочатку обчислюємо значення двох критеріїв для кожної з шести ~~можливих~~ альтернатив, потім для пошуку можливих альтернатив послідовно виключаємо з поведінки можливих доміновані альтернативи, поки не дійдено до оптимальної. Будемо лінійну зборку, використовувати оцінки альтернатив у просторі критеріїв

```
import java.util.Arrays;
import java.util.ArrayList;
import java.text.DecimalFormat;

public class Main {
    private static final int N = 11;
    private static int[] x = new int[10];
    private static int[] y = new int[10];
    private static DecimalFormat df = new DecimalFormat("###.##");

    public static void main(String[] args) {
        setCoordinates();
        show(x, y);
    }

    private static void PerfectPoint(int[] x, int[] y) {
        int xMax = Arrays.stream(x).max().getAsInt();
        int yMax = Arrays.stream(y).max().getAsInt();

        System.out.println("\n\nІдеальна точка це (" + xMax + ", " + yMax + ")");
        List<Double> distances = new ArrayList<>();
        List<Integer> indexes = new ArrayList<>();
        for (int i = 0; i < x.length; i++) {
            if ((x[i] == xMax) && (y[i] == yMax)) {
                System.out.println("Ідеальна точка належить до множини допустимих розв'язків");
                return;
            } else {
                indexes.add(i);
                distances.add(getDistance(xMax, yMax, x[i], y[i]));
            }
        }

        double minDistance = distances.stream().min(Comparator.comparing(Double::doubleValue)).get();
        int index = distances.indexOf(minDistance);
        index = indexes.get(index);
        System.out.print("Q  |");
        for (double j : distances) {
            System.out.print(" " + df.format(j) + "  |");
        }
        System.out.println();
        System.out.println("За методом ідеальної точки найменша віддаль = " + df.format(minDistance) +
            " при x = " + x[index] + ", y = " + y[index]);
    }
}
```

```

private static void setCoordinates() {
    for (int i = 0; i < 10; i++) {
        x[i] = getRandomNumber();
        y[i] = getRandomNumber();
    }
}

private static int[] getXCriterion(int[] x, int [] y) {
    int [] XC = new int[x.length];
    for (int i = 0; i < x.length; i++) {
        XC[i] = 2*x[i] + y[i] * y[i];
    }
    return XC;
}

private static int[] getYCriterion(int[] x, int [] y) {
    int [] XY = new int[x.length];
    for (int i = 0; i < x.length; i++) {
        XY[i] = -2*x[i]*x[i] + y[i];
    }
    return XY;
}

private static int getRandomNumber() {
    int min = N - 3;
    int max = N + 3;
    return (int) (Math.random() * ((max - min) + 1)) + min;
}

private static void show(int[] x, int[] y) {
    System.out.println(" | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |");
    System.out.print("x  ");
    for (int j : x) {
        System.out.print(" " + j + " ");
    }
    System.out.println();
    System.out.print("y  ");
    for (int j : y) {
        System.out.print(" " + j + " ");
    }
    System.out.println();
    System.out.println("Критерії якості: ");
    int [] XC = getXCriterion(x, y);
    int [] YC = getYCriterion(x, y);
    System.out.print("Q1  ");
    for (int j : XC) {
        System.out.print(" " + j + " ");
    }
    System.out.println();
    System.out.print("Q2  ");
    for (int j : YC) {
        System.out.print(" " + j + " ");
    }
    System.out.println();

    PerfectPoint(XC, YC);
}
}

```