

**Прізвище:** Дорош  
**Ім'я:** Володимир  
**По-батькові:** Юрійович  
**Група:** КН-405  
**Варіант:** 5  
**Github:** [https://github.com/sQverful/nulp\\_tpr\\_labs](https://github.com/sQverful/nulp_tpr_labs)



**Кафедра:** САПР  
**Дисципліна:** Теорія прийняття рішень  
**Перевірів:** Кривий Р.З.

## **ЗВІТ**

до лабораторної роботи №3 на  
тему "Метод експертної оцінки"

**Мета роботи:** Отримати практичні навички використання різних методів колективного голосування.

### **Короткі теоретичні відомості:**

Парадокс Кондорсе — відомий парадокс теорії суспільного вибору, вперше описаний Кондорсе у 1785 році.

Полягає в тому, що правило простої більшості не в змозі забезпечити транзитивність бінарного відношення громадської переваги серед варіантів, що обираються. В силу нетранзитивності, результат може залежати від порядку голосування, що дає можливість маніпуляції вибором більшості.

Узагальнений теоремою «про неможливість» Ерроу в 1951 році.

На практиці ідея про необхідність ранжування кандидатів реалізована в голосуванні за методом Шульце.

Метод Борда (правило Борда) - система голосування, запропонована в 1770 році Жан-Шарлем де Борда з метою більш ретельного обліку переваг виборців в умовах безлічі кандидатів. Широко використовується і в сучасності.

Відповідно до цього методу результати голосування виражаються у вигляді числа балів, набраних кожним з кандидатів. Так, під час виборів з  $n$  кандидатів кожен голосує ранжує всіх кандидатів строго по спадаючій переваги, за перше місце за бажанням кандидата присуджується  $n$  балів, за друге —  $n-1$  балів і т. д. (за останнє місце - 1 бал), всі набрані бали кандидатами підсумовуються. Відповідно, переможцем виборів вважається кандидат, який набрав найвищий сумарний бал.

### **Індивідуальне завдання:**

Який кандидат виграє А,Б,С по методу Кондорсе і по методу Борда для заданого розподілу голосів?

Порядок вирішення завдання:

- 1) Провести розрахунок для кожного методу колективного голосування.
- 2) Описати порядок виконання роботи і вказати переможці за кожним методом
- 3) Реалізувати програмне забезпечення, яке б розв'язувало дану задачу. Мова програмування неважлива. Обов'язково: дані мають зчитуватись з файлу і виводитись у табличній формі.

5.	24	A -> Б -> С
	27	A -> С -> Б
	26	С -> Б -> А
	26	Б -> С -> А
	16	С -> А -> Б

Рис.1. Вхідні дані.

### Виконання індивідуального завдання:

Метод Борда. Задавши коефіцієнти пріоритетності для кожного кандидата від 2 до 0, отримаємо:

- 1)  $A = 24 * 2 = 48$ ;  $B = 24 * 1 = 24$ ;  $C = 24 * 0 = 0$
- 2)  $A = 27 * 2 = 54$ ;  $C = 27 * 1 = 27$ ;  $B = 27 * 0 = 0$
- 3)  $C = 26 * 2 = 52$ ;  $B = 26 * 1 = 26$ ;  $A = 26 * 0 = 0$
- 4)  $B = 26 * 2 = 52$ ;  $C = 26 * 1 = 26$ ;  $A = 26 * 0 = 0$
- 5)  $C = 16 * 2 = 32$ ;  $A = 16 * 1 = 16$ ;  $B = 16 * 0 = 0$

В результаті сумарного підрахунку голосів отримаємо:  $A = 118$ ;  $B = 102$ ;  $C = 137$ .

Метод Кондорса.

Виконавши попарне порівняння одного кандидата відносно іншого, отримаємо:

$$A > B = 24 + 27 + 16 = 67$$

$$A > C = 24 + 27 = 51$$

$$B > A = 26 + 26 = 52$$

$$B > C = 24 + 26 = 50$$

$$C > A = 26 + 26 + 16 = 68$$

$$C > B = 27 + 26 + 16 = 69$$

Складемо фінальну пропорцію:  $C > A > B$

### Код програми:

#### 1. Метод Борда:

```
public class Bord {

    public static void main(String[] args) {

        String input = Util.readFile("part3.txt");

        System.out.println(input);

        System.out.println(bord(input));

    }

    public static String bord(String input) {
        StringBuilder sbResult = new StringBuilder();
        Pattern pattern = Pattern.compile(".+");
        Matcher matcher = pattern.matcher(input);

        String[][] values = new String[5][4];
        int pointer = 0;

        String temp;
```

```

while (matcher.find()) {
    temp = matcher.group();
    values[pointer] = temp.split(",");
    pointer++;
}

int totalA = 0;
int totalB = 0;
int totalC = 0;

Element[] elValues = new Element[5];

for (int i = 0; i < 5; i++) {
    elValues[i] = new Element();
    elValues[i].value = Integer.valueOf(values[i][0]);
    for (int j = 0; j < values[i].length; j++) {
        int tp = j;
        if (values[i][j].contains("A")) {
            elValues[i].index_A = (tp - 3) * -1;
        } else if (values[i][j].contains("B")) {
            elValues[i].index_B = (tp - 3) * -1;
        } else if (values[i][j].contains("C")) {
            elValues[i].index_C = (tp - 3) * -1;
        }
    }
}

for (int i = 0; i < elValues.length; i++) {
    totalA = totalA + elValues[i].index_A * elValues[i].value;
    totalB = totalB + elValues[i].index_B * elValues[i].value;
    totalC = totalC + elValues[i].index_C * elValues[i].value;
}

sbResult.append("A = " + totalA + " B = " + totalB + " C = " + totalC + "
Winner: ");

if (totalA > totalB && totalA > totalC) {
    sbResult.append("A");
}
if (totalB > totalA && totalB > totalC) {
    sbResult.append("B");
}
if (totalC > totalB && totalC > totalA) {
    sbResult.append("C");
}

return sbResult.toString();
}

public static class Element extends Object {

    Element() {
    }
    int value;
    int index_A;
    int index_B;
    int index_C;
}
}

```

## 2. Метод Кондорса:

```
3. public class Condorce {
    public static void main(String[] args) {

        String input = Util.readFile("part3.txt");
        System.out.println(condorce(input));

    }

    public static String condorce(String input) {
        StringBuilder sbResult = new StringBuilder();
        Pattern pattern = Pattern.compile(".+");
        Matcher matcher = pattern.matcher(input);

        String[][] values = new String[5][4];
        int pointer = 0;

        String temp;

        while (matcher.find()) {
            temp = matcher.group();
            values[pointer] = temp.split(",");
            pointer++;
        }

        int AB = 0;
        int AC = 0;
        int BA = 0;
        int BC = 0;
        int CA = 0;
        int CB = 0;

        Bord.Element[] elValues = new Bord.Element[5];

        for (int i = 0; i < 5; i++) {
            elValues[i] = new Bord.Element();
            elValues[i].value = Integer.parseInt(values[i][0]);
            for (int j = 0; j < values[i].length; j++) {
                int tp = j;
                if (values[i][j].contains("A")) {
                    elValues[i].index_A = (tp - 3) * -1;
                } else if (values[i][j].contains("B")) {
                    elValues[i].index_B = (tp - 3) * -1;
                } else if (values[i][j].contains("C")) {
                    elValues[i].index_C = (tp - 3) * -1;
                }
            }
        }

        for (int i = 0; i < elValues.length; i++) {
            if (elValues[i].index_A > elValues[i].index_B) {
                AB += elValues[i].value;
            } else {
                BA += elValues[i].value;
            }

            if (elValues[i].index_A > elValues[i].index_C) {
                AC += elValues[i].value;
            } else {
                CA += elValues[i].value;
            }
        }
    }
}
```

```

        if (elValues[i].index_B > elValues[i].index_C) {
            BC += elValues[i].value;
        } else {
            CB += elValues[i].value;
        }
    }

    sbResult.append("A>B = " + AB + " A>C = " + AC + " B>A = " + BA + " B>C = " + BC
+ " C>A = " + CA + " C>B = " + CB);

    int totalA = AB + BC;
    int totalB = BC + BA;
    int totalC = CB + CA;

    if (totalA > totalB && totalA > totalC) {
        sbResult.append(" Winner: A");
    }
    if (totalB > totalA && totalB > totalC) {
        sbResult.append(" Winner: B");
    }
    if (totalC > totalB && totalC > totalA) {
        sbResult.append(" Winner: C");
    }

    return sbResult.toString();
}

public static class Element extends Object {

    Element() {

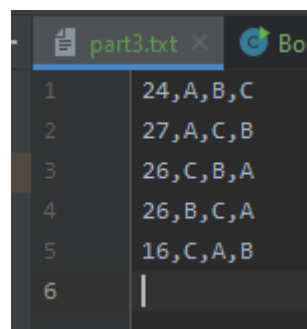
    }
    int value;
    int index_A;
    int index_B;
    int index_C;

}

}

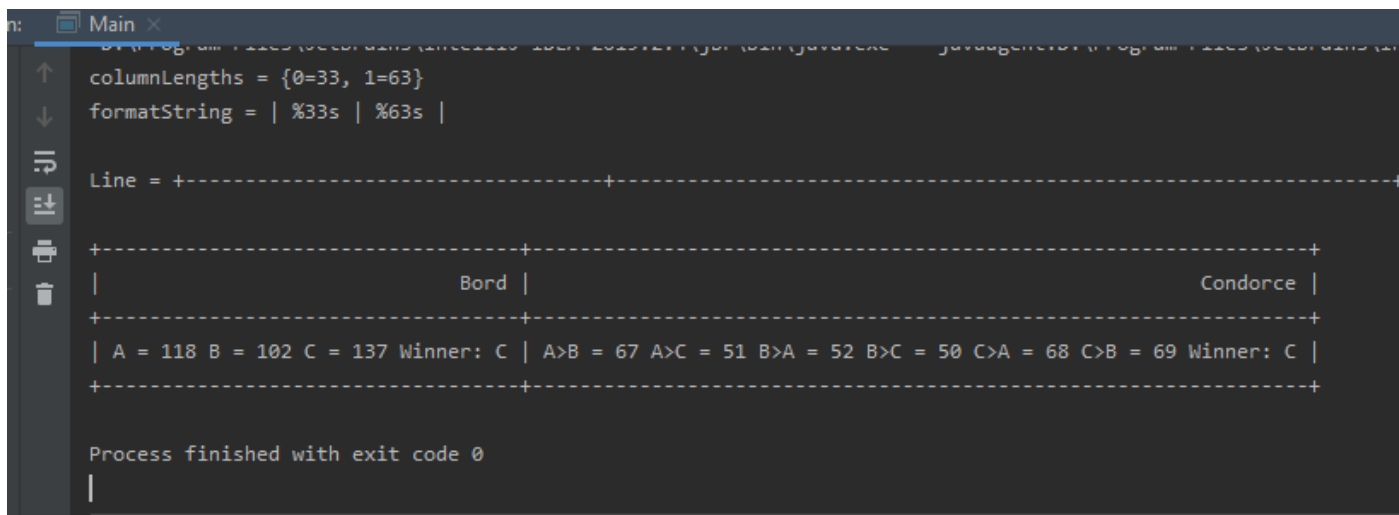
```

**Результат виконання програми:**



Line	Data
1	24,A,B,C
2	27,A,C,B
3	26,C,B,A
4	26,B,C,A
5	16,C,A,B
6	

Рис.2. Файл із вхідними даними part3.txt.



```
columnLengths = {0=33, 1=63}
formatString = | %33s | %63s |

Line = +-----+-----+
+-----+-----+
|          Bord          |          Condorce          |
+-----+-----+
| A = 118 B = 102 C = 137 Winner: C | A>B = 67 A>C = 51 B>A = 52 B>C = 50 C>A = 68 C>B = 69 Winner: C |
+-----+-----+

Process finished with exit code 0
```

Рис.3. Результат виконання програми.

**Висновок:** Під час виконання даної лабораторної роботи, я опрацював 2 методи підрахунку голосів - Борда та Кондорса. Метод Борда та метод Кондорсе часто не дають інтуїтивно очікуваних результатів при підрахунку. При практичному застосуванні методу часто виникає парадокс Кондорсе, коли переваги протилежних груп виборців суперечать одна одній. Походження недоліків методу обумовлено, перш за все, тим, що до об'єктів нечислової природи (місцях в порядку переваги) застосовуються арифметичні операції.