



Міністерство освіти і науки України
Національний технічний університет України
“Київський політехнічний інститут імені Ігоря Сікорського”
Факультет інформатики та обчислювальної техніки
Кафедра інформаційні систем та технологій

Лабораторна робота №4
із дисципліни «Основи програмування»
Тема: «**Масиви**»

Виконав:
Студент групи ІА-24
Іскандаров Ельмір
Філімонов Євгеній
Сіденко Дар'я
Яблонський Данило

Перевірив:
Колеснік Валерій Миколайович

Хід роботи:

1. Повторити теоретичні відомості
2. Виконати три завдання з таблиці 2 відповідно до свого варіанту у таблиці 1.
 - В одному з завдань обов'язково має бути використаний цикл «for»
 - В одному з завдань обов'язково має бути використаний цикл «for-each»
 - Кожне завдання має бути реалізовано як окремий клас.
 - Кожен клас має складатись щонайменше з двох методів:
 - `public static void main(String[] args)` - точка входу. Містить код, що кілька разів знаходить результат завдання при різних значеннях аргументів та параметрів. Для перевірки мають бути присутні як дозволені так і заборонені комбінації аргументів та параметрів.
 - Метод, що реалізує задане завдання. Метод має перевіряти аргументи та у разі їх помилковості аварійно закінчувати свою роботу шляхом викидання стандартного виключення `IllegalArgumentException`, `NullPointerException` або `IndexOutOfBoundsException` (дивись л/р №3). В жодному разі цей метод не повинен напряду взаємодіяти з користувачем через консоль або інший UI (ніколи не змішуйте бізнес-логіку та користувацький інтерфейс).
 - Клас може містити інші допоміжні методи.

61	Задано масиви $A(n)$ та $B(n)$. Сформувати масив $C(n)$ у наступний спосіб: елемент масиву C дорівнює сумі відповідних елементів масивів A і B , якщо обидва елементи додатні; елемент масиву C дорівнює добутку відповідних елементів масивів A і B , якщо обидва елементи від'ємні; елемент масиву C дорівнює нулю в усіх інших випадках.
----	--

```
import java.util.Arrays;

public class Task61 {
    public static void main(String[] args) {
        int[] arrayA = {1, 2, 3, 4, 5};
        int[] arrayB = {4, 6, 1, 5, 9};
        int[] arrayA1 = {-1, -2, 3, 4, -5};
        int[] arrayB1 = {4, -6, -1, 5, 9};
        int[] arrayA2 = null;
        int[] arrayB2 = {4, -6, -1, 5, 9};
        int[] arrayA3 = {-1, -2, 3, 4, -5};
        int[] arrayB3 = {4, -6, -1, 5, 9, 8};
        printResult(arrayA, arrayB);
        printResult(arrayA1, arrayB1);
        printResult(arrayA2, arrayB2);
        printResult(arrayA3, arrayB3);
    }
}
```

```

public static int[] calculation(int[] arrayA, int[] arrayB) {
    if (arrayA.length != arrayB.length) {
        throw new IllegalArgumentException("different arrays length");
    }
    int[] arrayC = new int[arrayA.length];
    for (int i = 0; i < arrayC.length; i++) {
        if (arrayA[i] > 0 && arrayB[i] > 0) {
            arrayC[i] = arrayA[i] + arrayB[i];
        } else if (arrayA[i] < 0 && arrayB[i] < 0) {
            arrayC[i] = arrayA[i] * arrayB[i];
        } else {
            arrayC[i] = 0;
        }
    }
    return arrayC;
}

```

4 usages

```

public static void printResult(int[] arrayA, int[] arrayB) {
    try {
        System.out.println(Arrays.toString(calculation(arrayA, arrayB)));
    } catch (IllegalArgumentException e) {
        System.out.println("Exception: " + e.getMessage());
    } catch (NullPointerException e) {
        System.out.println("Exception: array is null");
    }
}

```

```

public static void printResult(int[] arrayA, int[] arrayB) {
    try {
        System.out.println(Arrays.toString(calculation(arrayA, arrayB)));
    } catch (IllegalArgumentException e) {
        System.out.println("Exception: " + e.getMessage());
    } catch (NullPointerException e) {
        System.out.println("Exception: array is null");
    }
}

```

[5, 8, 4, 9, 14]

[0, 12, 0, 9, 0]

Exception: array is null

Exception: different arrays length

```
import java.util.Arrays;

public class Task38 {
    1 usage
    private static double task38(int[] userArray){
        int sum = 0;
        sum += userArray[2];
        Arrays.sort(userArray);
        sum += userArray[userArray.length-1];
        return sum;
    }

    public static void main(String[] args){
        int[] aArray = new int[] {-1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
        int[] bArray = new int[] {-3, 1, 6, -10, 7, 0, -2, 8, -2};
        int[] cArray = new int[0];
        printResults(aArray);
        printResults(bArray);
        printResults(cArray);
    }

    3 usages
    static void printResults(int[] resArray){
        try {
            System.out.println("Сума 3 і найбільшого значень: " +task38(resArray));
        } catch(ArrayIndexOutOfBoundsException e) {
            System.out.println("Exception: index is out of bounds");
        }
    }
}
```

Сума 3 і найбільшого значень: 13.0

Сума 3 і найбільшого значень: 14.0

Exception: index is out of bounds

```

public class Task8 {
    public static void main(String[] args){
        int[] x = {1, 2, 3, 4, 5, 6, 7, 8, 9};
        int[] x1 = null;
        printResults(x);
        printResults(x1);
    }
    1 usage
    public static int calculation(int[] a) {
        int sum = 0;
        for ( int i:a){
            if (i % 3 == 0){
                sum +=i;
            }
        }
        return sum;
    }
    2 usages
    static void printResults(int[] resArray){
        try {
            System.out.println("Сума чисел, кратних 3: " + calculation(resArray));
        } catch (NullPointerException e) {
            System.out.println("Exception: array is null");
        }
    }
}

```

```

Сума чисел, кратних 3: 18
Exception: array is null

```

Висновок: в ході лабораторної роботи ми більш детально розглянули масиви. Дізнались про стек, для чого він потрібен і яка різниця між стеком та купою. Розглянули переваги та недоліки «for» та «for-each» при роботі з масивами.