

Основи програмування
Лабораторна робота №5
Тема: Масиви 2

Виконали студенти групи ІА-23:

Содолинський Вадим
Калина Сергій

Дятченко Іван

Перевірив:

Колеснік В. М.

Хід роботи:

1. Повторити теоретичні відомості
2. Виконати завдання з таблиці 2 відповідно до свого варіанту у таблиці 1.
 - 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.
3. Відповісти на контрольні питання

Завдання 1(1). Знайти суму елементів масиву, у яких індекс кратний 3

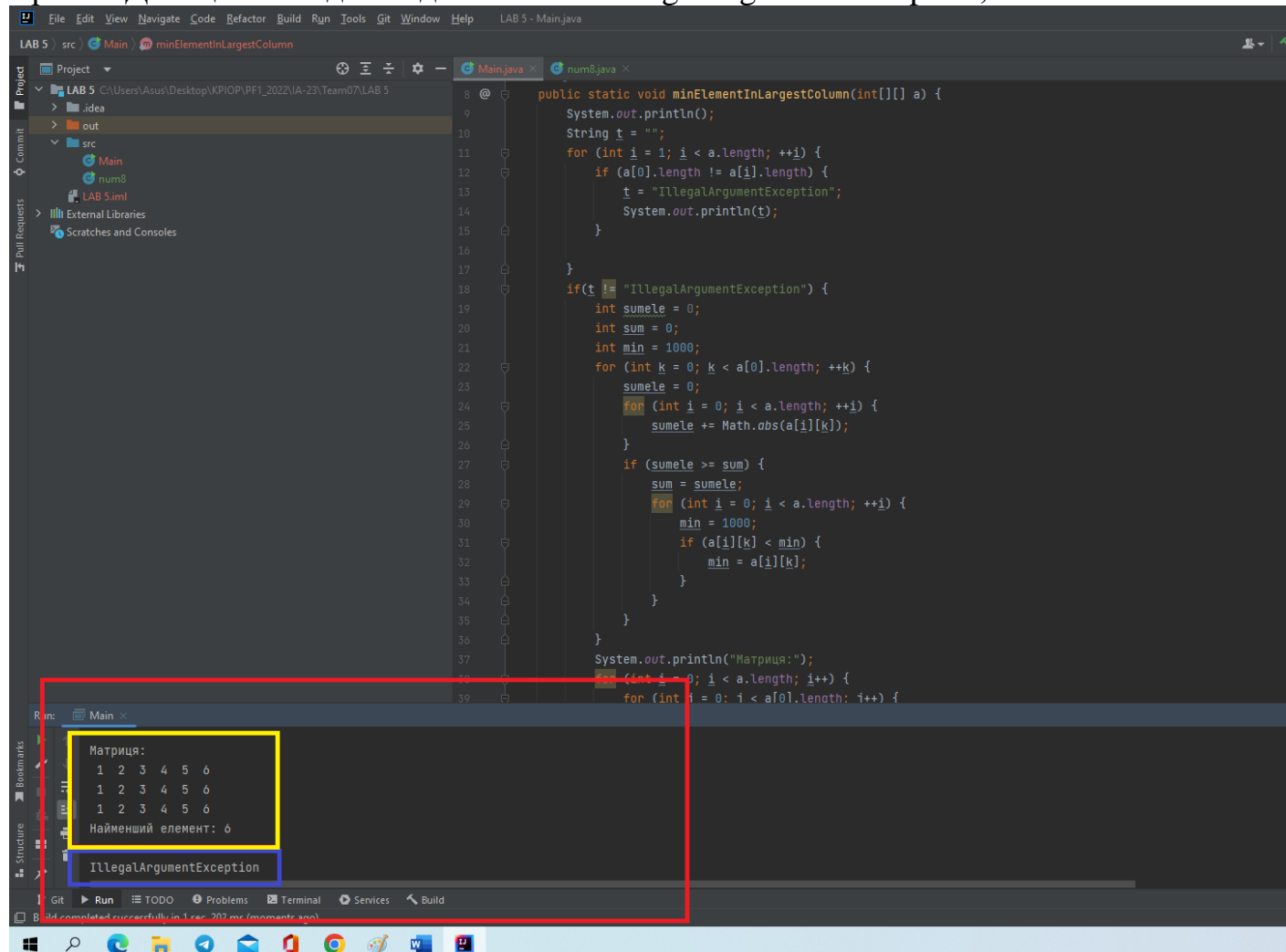
Крок 1. Ми створили метод `minElementLargestColumn`, який має 1 параметр(`int[][] a`). Спочатку ми робимо перевірку матриці за допомогою циклу(перевіряє чи всі рядки мають однакову кількість стовпчиків)

```
String t = "";
for (int i = 1; i < a.length; ++i) {
    if (a[0].length != a[i].length) {
        t = "IllegalArgumentException";
        System.out.println(t);
    }
}
```

і якщо вона не проходить перевірку, то метод виведе `IllegalArgumentException` і закінчить роботу.

Крок 2. Якщо матриця проходить перевірку, то ми створюємо три `int`: `sumele`(сума модулів елементів деякого стовпця), `sum`(найбільше значення суми на даний момент), `min`(найменший елемент стовпця, сума модулів якого є найбільшою). Далі створюємо два цикли, за допомогою яких знаходимо спочатку стовпець, у якого сума модулів найбільша(найбільше значення суми на даний момент), а потім його найменший елемент(`min`).

Крок 3. Далі цей метод виводить нам або `IllegalArgumentException`, або `min`.



Завдання 1(8). Знайти суму елементів масиву, у яких індекс кратний 3

Крок 1. Ми створили метод `multiply`, який має 2 параметри(`int[][] a`, `int[][] b`). Спочатку ми робимо перевірку матриць за допомогою циклу(перевіряє чи всі рядки мають однакову кількість стовпчиків, також перевіряє чи кількість стовпчиків першої матриці дорівнює кількості рядків другої матриці) і якщо вони не проходять перевірку, то метод виведе `IllegalArgumentException` і закінчить роботу.

Крок 2. Якщо матриці проходять перевірку, то ми створюємо нульову матрицю `c`, яка має кількість рядків матриці `a` та кількість стовпчиків `b`, `int num`(змінна буде приймати в себе значення певного елемента і потім підставляти його в матрицю `c`). Далі за допомогою трьох циклів обчислюємо значення елементів матриці `c` і завдяки `num` замінюємо нулі на щойно обчислені значення елементів.

Крок 3. Далі цей метод виводить нам або `IllegalArgumentException`, або матрицю `c`.

The screenshot displays the IntelliJ IDEA IDE with the `multiply` method implemented in `num8.java`. The method signature is `public static void multiply(int[][] a, int[][] b)`. It first checks if the dimensions of the input matrices are compatible for multiplication. If not, it prints `IllegalArgumentException`. If compatible, it calculates the product matrix `c` and prints its elements. The `Run` window at the bottom shows the output of the program, which is a 3x3 matrix of numbers.

```
8 @
9
10 for (int i = 1; i < a.length; ++i) {
11     if (a[0].length != a[i].length) {
12         t = "IllegalArgumentException";
13         System.out.println(t);
14     }
15 }
16 for (int i = 1; i < b.length; ++i) {
17     if (b[0].length != b[i].length) {
18         t = "IllegalArgumentException";
19         System.out.println(t);
20     }
21 }
22
23 if (t != "IllegalArgumentException") {
24     if (a[1].length != b.length) {
25         t = "IllegalArgumentException";
26         System.out.println(t);
27     }
28 }
29
30 if (t != "IllegalArgumentException") {
31     int num = 0;
32     int[][] c = new int[a.length][b[1].length];
33     for (int i = 0; i < a.length; ++i) {
34         for (int k = 0; k < b[0].length; ++k) {
35             for (int q = 0; q < a[1].length; ++q) {
36                 num += a[i][q] * b[q][k];
37             }
38             c[i][k] = num;
39         }
40     }
41     for (int i = 0; i < c.length; ++i) {
42         for (int k = 0; k < c[0].length; ++k) {
43             System.out.print(c[i][k] + " ");
44         }
45         System.out.println();
46     }
47 }
```

Run: num8 x

```
"C:\Program Files\Java\jdk-17.0.4.1\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2022.2.2\lib\idea_rt.jar" 14 32
32 77
50 122
IllegalArgumentException
Process finished with exit code 0
```

Висновок: ми покращили свої навички в роботі з масивами та закріпили свої знання з попередніх тем.