

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра інформаційних систем та технологій

Лабораторна робота №5

з дисципліни «Основи програмування»

Тема: Масиви 2

Виконали:
Студенти групи ІА-23
Мозоль Владислав,
Куроч Владислав,
Лядський Дмитро
Дата здачі: 04.11.2022

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 напяму 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

Task 5

```
public class task5 {  
    public static void main(String[] args){  
        printResult(a: 4, b: 6);  
        System.out.println("=====");  
        printResult(a: 4, b: 8);  
        System.out.println("=====");  
        printResult(a: 8, b: 8);  
        System.out.println("=====");  
        printResult(a: 4, b: 4);  
    }  
    1 usage  
    public static void clearNE(int a, int b) {  
        if (a == b){  
            throw new IllegalArgumentException("Це не прямокутник!!!");  
        }  
        int[][] aa = new int[a][b];  
        for(int i = 0; i < a; i++){  
            for(int j = 0; j < b; j++){  
                aa[i][j] = i < j ? 0 : i > j ? 5 : 0;  
                System.out.print(aa[i][j]);  
            }  
            System.out.println(" ");  
        }  
    }  
    4 usages  
    static void printResult(int a,int b){  
        try {  
            clearNE(a,b);  
        } catch (IllegalArgumentException e){  
            System.out.println("Ybara!" + e.getMessage());  
        }  
    }  
}
```

```

000000
500000
550000
555000

=====
00000000
50000000
55000000
55500000

=====
Увага!Це не прямокутник!!!
=====
Увага!Це не прямокутник!!!

Process finished with exit code 0
|

```

Task 11

```

1  ▶ public class task11 {
2
3  ▶  public static void main(String[] args) {
4      int[][] oneWin = { {0, 1, 2, 0, 2}, {1, 0, 2, 0, 2}, {1, 1, 0, 0, 1}, {1, 0, 2, 0, 2}, {2, 2, 1, 2, 0}};
5      printResult(oneWin);
6      int[][] twoWin = { {0, 1, 2, 0, 2}, {1, 0, 2, 0, 2}, {1, 1, 0, 1, 1}, {1, 0, 2, 0, 2}, {2, 2, 1, 2, 0}};
7      printResult(twoWin);
8      int[][] onlyWin = { {1, 1, 2, 0, 2}, {1, 0, 2, 1, 2}, {1, 1, 0, 1, 1}, {1, 2, 2, 0, 2}, {2, 2, 1, 2, 0}};
9      printResult(onlyWin);
10     int[][] onlyLose = { {0, 1, 2, 0, 2}, {1, 0, 2, 0, 2}, {1, 0, 0, 1, 1}, {1, 0, 2, 0, 2}, {2, 0, 1, 2, 0}};
11     printResult(onlyLose);
12 }
13 @ 1 usage
14 public static void getFlowless(int[][] results){
15     int i=0,j=0;
16     for(int[] l : results) {
17         for (int s : l)
18             i = s == 2 ? i + 1 : s == 0 ? i : s == 1 ? i + 1 : i;
19         j = i >= 4 ? j + 1 : j;
20         i = 0;
21     }
22     if (j == 0){
23         throw new IllegalArgumentException("Жодна з команд не закінчила сезон без програвів!");
24     }
25     System.out.println("В цьому сезоні стільки команд закінчило без програвів: " + j);
26 }
27 4 usages
28 public static void printResult(int[][] results){
29     try {
30         getFlowless(results);
31     }
32 }

```

```

26     }
    4 usages
27     public static void printResult(int[][] results){
28         try {
29             getFlawless(results);
30         } catch (IllegalArgumentException e){
31             System.out.println("Увага! " + e.getMessage());
32         }
33     }
34 }

```

```

"F:\For programming\Java\bin\java.exe" "-javaagent:F:\For p
В цьому сезоні стільки команд закінчило без програвів: 1
В цьому сезоні стільки команд закінчило без програвів: 2
В цьому сезоні стільки команд закінчило без програвів: 5
Увага! Жодна з команд не закінчила сезон без програвів!

Process finished with exit code 0

```

Контрольні питання:

1. Яким чином двовірні масиви розташовуються у пам'яті?

Пояснення для одновимірних масивів дуже просте – всі вони розташовані в пам'яті підряд. Двовимірні масиви розташовуються дещо інакше - спочатку елементи пер-шого рядка, потім другого і т. д. Розташування масивів більшої розмірності пояснюється аналогічно.

2. Чи можна створити двовірний масив непрямокутної форми?

Двовимірні масиви можуть бути двох видів: прямокутний масив у вигляді таблиці; ступінчатий масив. У цьому масиві кількість елементів кожного рядка (масиву) може бути різною

3. При створенні масиву його зміст заповнюється спеціальними значеннями за-замовчуванням або випадковими значеннями, що були у пам'яті, яка була виділена під масив?

Спеціальними значеннями за-замовчуванням

4. Як дізнатися розмір двовірного масиву, якщо відомо, що він прямокутний?

За допомогою команди `length`.

5. До якого результату призведе виконання наступного фрагменту коду:

```

int[][] a = {{1,2,3}, {4,5,6},{7,8,9}};
System.out.println(a.length);

```

3 Команда `System.out.println(a.length);` виводить довжину масиву.

6. До якого результату призведе виконання наступного фрагменту коду:

```
int[][] b = {{1}, {2, 3}, {4, 5, 6}};  
System.out.println(b.length);  
System.out.println(b[0].length);
```

3 `System.out.println(b.length);` виводить довжину масиву

1 `System.out.println(b[0].length);` виводить кількість елементів в першому елементі масиву b

7. Є наступний фрагмент коду:

```
int[] a = {1, 2};  
int[][] b = {{3,4}, {5,6}};  
int[] c;  
int[][] d;
```

Опишіть результат, до якого призведе кожна з наступних операцій:

```
c = a;  
c = b;  
d = a;  
d = b;
```

В першому варіанті `c = a` масив c стає посиланням на масив a, а тому коли звернемось до масиву c він буде складатися з чисел які є в масиві a. Аналогічно з `d = b`. А от якщо `c = b`, `d = a` то воно просто не буде компілюватися тому що розмірності масивів різні тому компілятор вкаже на помилку.

Висновок: виконуючи цю лабораторну роботу ми закріпили свої знання, попрактикувалися в розробці програм, які вирішують невеликі задачі за допомогою масивів.