

- I. Utwórz metodę `jestRowna` sprawdzającą czy dostarczone jako argumenty dwie dwuwymiarowe tablice a , b (niekoniecznie prostokątne)

- mają takie same wymiary,
- spełniają warunek, że $a[i][j] = b[i][j]$ dla dowolnych, dostępnych indeksów i , j .

Metoda zwróci rezultat w postaci wartości logicznej `true` tylko gdy oba powyższe są spełnione jednocześnie.

- II. Dany jest nagłówek funkcji:

```
1 public static int [][] calculateSquares(int screenWidth, int ←  
    screenHeight, int side)
```

przyjmujący jako argumenty odpowiednio: szerokość i wysokość ekranu, oraz długość boku kwadratu. Uzupełnij ciało tej metody algorytmem wyliczającym ile kwadratów o zadanej długości boku można umieścić na ekranie. Następnie wylicz współrzędne (górnego lewego rogu) x i y wszystkich kwadratów i zwróć je w postaci dwu wymiarowej tablicy w której pierwsza kolumna opisuje współrzędne x a druga y .

- III. Zaimplementuj statyczną metodę iteracyjną

```
1 int gcdI(int a, int b)
```

zwracającą NWD, czyli największy wspólny dzielnik, liczb a i b .

- IV. Zaimplementuj statyczną metodę rekurencyjną

```
1 int gcdR(int a, int b)
```

zwracającą NWD, czyli największy wspólny dzielnik, liczb a i b .

- V. Zaimplementuj statyczną metodę rekurencyjną

```
1 int maxElem(int [] arr, int from)
```

zwracającą największy element tablicy `arr` spośród jej elementów od tego o indeksie `from`.