

- I. Utwórz statyczną funkcję `findMax` przyjmującą jako argument tablicę wartości typu `int` i zwracającą największą z tych wartości. Następnie przedstaw przykład mechanizmu przeciążania funkcji, na przykładzie `findMax` i typu `double`.
- II. Utwórz statyczną funkcję `splitToDigits` przyjmującą jako argument wartość całkowitą dodatnią i zwracającą jednowymiarową tablicę wartości całkowitych dodatnich. Celem funkcji będzie utworzenie tablicy o rozmiarze tożsamym z ilością cyfr liczby dostarczonej jako argument, a następnie wypełnienie jej cyframi z jakich składa się dostarczona liczba.
- III. Utwórz statyczną funkcję `isArmstrongNumber` przyjmującą jako argument wartość typu `int` i dostarczającą jako rezultat wartość logiczną. Przyjmij że działanie tej funkcji będzie adekwatne do nazwy funkcji i będzie sprawdzało czy dostarczona jako argument wartość jest liczbą *Armstronga* (*narcystyczną*).

IV. 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 .