

# kurs języka Java

## rozkład liczby na czynniki pierwsze

Instytut Informatyki  
Uniwersytetu Wrocławskiego

Paweł Rzechonek

---

Zadanie.

Napisz program, który wypisze na standardowym wyjściu `System.out` rozkład na czynniki pierwsze liczb całkowitych zadanych poprzez argumenty wywołania. Rozkład każdej z nich ma być wypisany w osobnym wierszu: najpierw liczba, potem znak `=` a potem czynniki pierwsze poodzielane znakiem `*`.

Program należy napisać w wybranym IDE (na przykład *IntelliJ IDEA*). Następnie należy go skompilować i uruchomić, definiując w środowisku uruchomieniowym argumenty wywołania. Można założyć, że parametrami wywołania programu będą liczby całkowite nieprzekraczające typu `long`. Jeśli program wywołano bez żadnego parametru, to należy wypisać na standardowym strumieniu dla błędów `System.err` instrukcję obsługi programu.

Programując to zadanie zdefiniuj klasę usługową `LiczbyPierwsze`. Klasa ta powinna posiadać tylko dwie publiczne metody statyczne:

- metoda `czyPierwsza(long)` testująca pierwszość liczby (zwraca wartość typu `boolean`);
- metoda `naCzynnikiPierwsze(long)` generująca rozkład liczby na czynniki pierwsze (zwraca tablicę typu `long[]`).

Implementując te metody wykorzystaj *sito Eratostenesa*, modyfikując je w taki sposób, że dla każdej liczby pamiętaj jej najmniejszy dzielnik pierwszy (będzie to trik pomocny przy wyliczaniu rozkładu liczby na czynniki pierwsze). Wymienione funkcje powinny udzielać poprawnych odpowiedzi dla wszystkich liczb typu `long` (weź pod uwagę, że nie możesz utworzyć tak dużego sita, więc zastanów się jak obejść to ograniczenie algorytmicznie). Klasa `LiczbyPierwsze` powinna być tak zdefiniowana, aby nie można było stworzyć jej instancji.

```
public final class LiczbyPierwsze {
    private final static int POTEGA2 = 21;
    private final static int[] SITO = new int[1 << POTEGA2];
    // potrzebny jest statyczny blok inicjalizacyjny dla sita
    // [0, 1, 2, 3, 2, 5, 2, 7, 2, 3, 2, 11, ... ]
    public static boolean czyPierwsza (long n) { ... }
    public static long[] naCzynnikiPierwsze (long n) { ... }
    // ...
}
```

Uwaga.

Rozkład liczby ujemnej na czynniki pierwsze ma się rozpoczynać od czynnika  $-1$ . Rozkład liczb  $-1$ ,  $0$  i  $1$  ma być tożsamościowy. Twój program powinien sobie skutecznie poradzić z liczbą  $-9223372036854775808$  oraz  $9223372036854775783$ .

Wskazówka.

Liczba złożona  $n$  posiada co najmniej jeden dzielnik pierwszy, który jest  $\leq \sqrt{n}$ .