

KURS PROGRAMOWANIA W JAVIE

ŁAŃCUCHY FIBONACCIEGO

Instytut Informatyki Uniwersytetu Wrocławskiego

Paweł Rzechonek

Zadanie.

Łańcuchy Fibonacciego dla napisów startowych a i b można zdefiniować rekurencyjnie:

$$S_n = \begin{cases} \epsilon & : n = 0 \\ a & : n = 1 \\ b & : n = 2 \\ S_{n-2} S_{n-1} & : n > 2 \end{cases}$$

Tak więc łańcuch S_n jest konkatenacją swoich dwóch poprzedników S_{n-2} i S_{n-1} (dla $n > 2$). Kilka początkowych łańcuchów Fibonacciego dla napisów startowych a i b to:

n	n -ty łańcuch Fibonacciego
0	
1	a
2	b
3	ab
4	bab
5	abbab
6	bababbab
7	abbabbababbab
8	bababbababbabbababbab
9	abbabbababbabbababbababbabbababbab

Napisz program, który wygeneruje i stablicuje n początkowych łańcuchów Fibonacciego a na końcu wydrukuje ostatni z nich S_n . Parametr n oraz napisy startowe mają być przekazane do programu poprzez argumenty wywołania. Sprawdź w swoim programie, czy wywołano go z trzema argumentami, przy czym pierwszy argument ma być niedużą liczbą całkowitą > 2 a ostatnie dwa argumenty mają być różnymi słowami jednoliterowymi. Oto kilka przykładowych wywołań takiego programu o nazwie **Fibo**:

```
$> java Fibo 3 p q
pq
$> java Fibo 10 a b
bababbababbabbababbabbababbabbababbabbababbababbababbab
$> java Fibo s t 5
błędne argumenty wywołania programu
$> java Fibo x y
błędne argumenty wywołania programu
```

Uwaga.

Program należy skompilować i uruchomić z wiersza poleceń! Podczas kompilacji użyj opcji `-encoding`, aby kompilator umiał poprawnie przetłumaczyć literały znakowe zaszyte w programie do *Unikodu*. Jeśli twój program był napisany w pliku kodowanym zgodnie ze standardem *ISO-8859-2* to wywołanie kompilatora będzie następujące:

```
$> javac -encoding iso-8859-2 Fibo.java
```

Przed kompilacją sprawdź jakiego kodowania użyłeś w swoim programie. Jeśli pisałeś pod Windowsem, może to być `windows-1250`.

Wskazówka 1.

Napis `String` można przekonwertować liczbę `int` korzystając z klasy opakowującej `Integer`:

```
int x = new Integer(args[0]);
```

Wskazówka 2.

Do porównywania napisów `String` użyj metody `compareTo`:

```
String a=null, b=null;
...
if (a.compareTo(b)==0) throw new IllegalArgumentException();
```

Wskazówka 3.

Do wyznaczenia długości napisu `String` użyj metody `length`:

```
String a=null, b=null;
...
if (a.length()!=1||b.lenght()!=1) throw new IllegalArgumentException();
```

Wskazówka 4.

Nie możemy założyć, że program zostanie wywołany z prawidłowymi parametrami, dlatego należy zadbać o obsługę sytuacji wyjątkowych za pomocą instrukcji `try-catch`:

```
try {
    ...
}
catch (NumberFormatException ex) {
    ...
}
catch (IllegalArgumentException ex) {
    ...
}
```