

Paradygmaty programowania - ćwiczenia

Lista 2

1. Jak głęboki będzie stos w Scali dla wywołania *evenR(3)* (funkcja zdefiniowana na wykładzie)?

Poniższe funkcje należy napisać w obu językach: OCaml i Scala (wykorzystując mechanizm dopasowania wzorca w zadaniach 4-6).

2. Liczby Fibonacciego są zdefiniowane następująco:

$$\begin{aligned}f(0) &= 0 \\f(1) &= 1 \\f(n+2) &= f(n) + f(n+1)\end{aligned}$$

Napisz dwie funkcje, które dla danego n znajdują n -tą liczbę Fibonacciego: pierwszą opartą bezpośrednio na powyższej definicji i drugą, wykorzystującą rekursję ogonową. Porównaj ich szybkość wykonania, obliczając np. 37-mą liczbę Fibonacciego.

3. Dla zadanej liczby rzeczywistej a oraz dokładności ε można znaleźć pierwiastek trzeciego stopnia z a wyliczając kolejne przybliżenia x_i tego pierwiastka:

$$\begin{aligned}x_0 &= a/3 \quad \text{dla } a \geq 1 \\x_0 &= a \quad \text{dla } a < 1 \\x_{i+1} &= x_i + (a/x_i^2 - x_i)/3\end{aligned}$$

Dokładność jest osiągnięta, jeśli $|x_i^3 - a| \leq \varepsilon * |a|$.

Napisz efektywną (wykorzystującą rekursję ogonową) funkcję *root3*, która dla zadanej liczby a znajduje pierwiastek trzeciego stopnia z dokładnością $1e-15$.

Uwaga. Przypominam, że OCaml nie wykonuje automatycznie żadnych koercji typów.

4. Zwiąż zmienną x z wartością 0 konstruując wzorce, do których mają się dopasować następujące wyrażenia:

$$\text{a) } [-2; -1; 0; 1; 2] \qquad \text{b) } [(1,2); (0,1)]$$

Np. dla wyrażenia $(\text{true}, "hello", 0)$ wymagany wzorem jest $(_ , _, x)$.

5. Zdefiniuj funkcję *initsegment* : $'a \text{ list} * 'a \text{ list} \rightarrow \text{bool}$ sprawdzającą w czasie liniowym, czy pierwsza lista stanowi początkowy segment drugiej listy. Każda lista jest swoim początkowym segmentem, lista pusta jest początkowym segmentem każdej listy.

6. Zdefiniuj funkcję *replace_nth* : $'a \text{ list} * \text{int} * 'a \rightarrow 'a \text{ list}$, zastępującą n -ty element listy podaną wartością (pierwszy element ma numer 0), np.

$$\text{replace_nth } ([1;2;3], 1, 5) \Rightarrow [1;5;3]$$