

Lista 4

Algebraiczne typy danych

W poniższych zadaniach **nie wolno** wykorzystywać funkcji wbudowanych obliczających **długość listy**, **odwracających listę** oraz **łączących dwie listy**.

Każde zadanie, poza implementacją funkcji, musi posiadać **kompletny zestaw testów**.

Do wykonania zadań należy wykorzystać mechanizmy poznane na wykładzie nr 4.

1) Zdefiniuj:

- a. Typ *tree3* (drzewo trójkowe) o następujących cechach: (OCaml i Scala) (2 pkt.)
 - Drzewo puste jest drzewem trójkowym,
 - Węzeł drzewa składa się z elementu oraz dokładnie trzech poddrzew.
- b. Zdefiniuj funkcjonal *mapTree3* w postaci rozwiniętej, wykonujący operację mapowania dla drzewa trójkowego. (OCaml i Scala) (6 pkt.)

2) Zapoznaj się z poniższym opisem wycinka rzeczywistości i wykonaj zadania:

„Tekst składa się z przynajmniej jednego zdania. Zdanie jest ciągiem zbudowanym z przynajmniej jednego wyrazu. Wyróżniamy trzy rodzaje zdań – oznajmujące (zakończone kropką), pytające (zakończone znakiem zapytania) oraz wykrzyknienia (zakończone wykrzyknikiem). Zdania zaczynają się wielką literą. Każdy wyraz składa się z przynajmniej jednego znaku.”

- a. Bazując na powyższym opisie zdefiniuj algebraiczny typ danych modelujący wycinek rzeczywistości. Pamiętaj, że typy często reprezentują kluczowe byty. Dla uproszczenia przyjmij, że wyrazy mają sens (są poprawnymi wyrazami) oraz wykorzystywane są jedynie małe litery. Typy zdań mają nie zawierać wspomnianych znaków interpunkcyjnych. Zastosuj angielskie nazwy. (OCaml i Scala) (6 pkt.)
- b. Zdefiniuj funkcje: (OCaml lub Scala)
 - *sentenceToString* zamieniającą Zdanie w ciąg znaków. Pamiętaj, że zdania zaczynają się z wielkiej litery i kończą odpowiednim znakiem interpunkcyjnym, (4 pkt.)
 - *textToString* zamieniającą Tekst w łańcuch znaków. Wykorzystaj funkcję *sentenceToString*. (2 pkt.)

Wskazówka:

W języku OCaml przydatne mogą być funkcje *Char.uppercase* oraz *let char_to_string = String.make 1;;*.