

Paradygmaty programowania

Lista 9

Zad. 1. [4+3p] Napisać w OCaml funkcję, typu $'a\ \textit{llist} \rightarrow (int \rightarrow int) \rightarrow 'a\ \textit{llist}$ przekształcającą listę leniwą do nowej listy leniwej, w której każdy element listy wejściowej jest powtórzony $f(i)$ razy, gdzie f jest funkcją a i jest pozycją elementu w liście wejściowej. Zadanie wykonać następnie w Scala. Wszystkie operacje należy wykonywać bezpośrednio na listach leniwych. Listę zwykłą można jedynie wykorzystać do utworzenia na jej podstawie listy leniwej lub ewentualnie do wyświetlenia listy leniwej.

Zad. 2. [3p] Zaimplementować słownik zawierający wyrazy, na które składa się liczba całkowita (wartość słowa) i częstość jej wystąpienia. Słownik ma mieć postać uporządkowaną względem wartości słowa. Napisać w OCaml funkcję typu $int \rightarrow 'a \rightarrow (int, int)\ \textit{słowa\ list}$ (gdzie *słowa* jest nazwą typu opisującego wyraz) dodającą wyraz do słownika (w przypadku istnienia wyrazu w słowniku należy tylko zwiększyć jego częstość).