

Lista 3

Funkcje wyższych rzędów

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

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

Do wykonania zadań należy wykorzystać mechanizmy poznane na wykładzie nr 3.

- 1) Napisać operator $><$ przyjmującą listy $[x_1; x_2; x_3; \dots]$ oraz $[y_1; y_2; y_3; \dots]$, i zwracającą funkcję argumentu f postaci:

$$f \rightarrow [f\ x_1\ y_1; f\ x_2\ y_2; f\ x_3\ y_3; \dots]$$

Jeżeli $n < 0$ zwrócić listę pustą. (OCaml) (25pkt.)

- 2) Napisać funkcję *coposites* generującą listę wszystkich liczb złożonych z przedziału $[2, n]$. Wykorzystać jedynie wyrażenia *for* oraz *List.range* tzn. nie można korzystać z rekurencji, ani funkcjonałów. (Scala) (25pkt.)

Wskazówka: Aby przekonwertować liczbę rzeczywistą na całkowitoliczbową w Scali należy wywołać metodę *.toInt* np. `2.57.toInt == 2`