

Wrocław, dn. 04.12.2018

**Paradygmaty programowania - studia stacjonarne, lista 8, 04.12.2018, godzina 09:15**  
**Czas na rozwiązanie: 70 min.**

**WAŻNE1: Do każdego zadania przygotuj 3-5 testów sprawdzających poprawność działania!**

**WAŻNE2: Nie wolno stosować funkcji bibliotecznych i konstrukcji innych niż te przedstawione na wykładzie 7.**

1) Zdefiniuj funkcję "polacz" łączącą naprzemiennie elementy dwóch list. Funkcję napisz w języku Java w sposób imperatywny przy użyciu tablic i w języku Ocaml używając list i stylu funkcyjnego.

Przykład:

[1;3;5;7;9;11] oraz [2;4;6;8] daje [1;2;3;4;5;6;7;8;9;11]

Punkty: 7 (język Java, Ocaml).

2) Napisz wersję funkcji zip (patrz wykład 2, strona 32) dla list leniwych.

Przykład

[1;2;3;4] [1;2;3;4] -> [1;1;2;2;3;3;4;4];

Punkty: 5 (język Ocaml)

3. Napisz funkcję powielającą elementy w kolekcji na podstawie drugiej kolekcji określającej ile razy elementy mają być powielone.

Funkcję napisz w języku Java w sposób imperatywny przy użyciu tablic i w języku Scala używając list i stylu funkcyjnego.

przykład: duplicate [1;2;3] oraz [0;3;1;4] daje wynik [2;2;2;3]

Punkty: 8 (język Java, Scala)