

Lista 1

W zadaniach z tej listy można używać tylko dwóch funkcji bibliotecznych: `List.hd` i `List.tl`.

1. Napisz dowolną funkcję o podanym typie:

- a) `int * int -> int * int`,
- b) `float * float -> bool`,
- c) `'a list * int -> 'a list`

2. Napisz funkcję *ends*: `'a list -> 'a * 'a`, zwracającą pierwszy i ostatni element zadanej listy, np.

`ends [1;2;3;5] => (1,5).`

3. Napisz funkcję *isSorted*: `'a list -> bool`, sprawdzającą, czy dana lista jest posortowana niemalejąco, np.

`isSorted [1;3;3;5;6;7] => true.`

4. Napisz funkcję *powers*: `int * int -> int list`, tworzącą listę wszystkich potęg zadanej liczby mniejszych lub równych od zadanej potęgi, np.

`powers (2,3) => [1;2;4;8].`

5. Napisz funkcję *split*: `'a list * 'a -> 'a list * 'a list` dzielącą daną listę na listę elementów mniejszych bądź równych od zadanego oraz listę elementów większych od zadanego, np.

`split (['a'; 's'; 'h'; 'g'], 'g') => (['a'; 'g'], ['s'; 'h'])`

6. Napisz funkcję *segments*: `'a list * int -> 'a list list` zamieniającą zadaną listę na listę list o zadanej maksymalnej długości, np.

`segments ([1;2;3;4;5;6;7;8;9], 2) => [[1;2];[3;4];[5;6];[7;8];[9]].`

7. Napisz funkcję *swap*: `'a list * int -> 'a list` dzielącą daną listę dwie podlisty w zadanym punkcie i zwracającą listę powstałą z zamiany tych podlist miejscami, np.

`swap (["a";"b";"5";"6"], 2) => ["5";"6";"a";"b"].`