# **Упражнения:** *Списъци*

## **Зад. 1** *Обръщане на списък*

Дефинирайте функция, която приема списък и връща като резултат списъкът в обратен ред.

### **Пример**

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| [1,2,3,4,5] | [5,4,3,2,1] |
| [1] | [1] |
| [] | [] |

### **Подсказки**

1. Проучете как работи (x : xs) pattern matching-а в Haskell
2. Използвайте рекурсивно `++` оператора между опашката и главата на списъка

## **Зад. 2** *Дължина на списък*

Дефинирайте собствена функция, която приема списък и връща като резултат броят на елементите в списъка.

### **Пример**

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| [123,456] | 2 |
| [1] | 1 |
| [] | 0 |
| "Hello, world!" | 13 |

### **Подсказки**

1. Обхождайте рекурсивно опашката на списъка като за всяко извикване връщайте 1 + резултата от рекурсията

## **Зад. 3** *Дупликиране на елементи от списък*

Дефинирайте функция, която приема списък и връща като резултат списъкът с дупликирани елементи.

### **Пример**

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| "abc" | "aabbcc" |
| [1,2,3] | [1,1,2,2,3,3] |
| [1] | [1,1] |
| [] | [] |

## **Зад. 4** *Премахване на всеки n-ти елемент*

Дефинирайте функция, която приема списък и число n и връща като резултат списък с премахнат всеки n-ти елемент от началния списък

### **Пример**

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| [1,2,3,4,5,6,7,8,9] 3 | [1,2,4,5,7,8] |
| [1,2,3,4,5,6,7,8,9] 1 | [] |
| [1,2,3] 3 | [1,2] |
| [] 3 | [] |

### **Подсказки**

1. Проучете как работят вградените в Haskell функции `take` и `drop`

## **Зад. 5** *Фибоначи (списък)*

Дефинирайте функция, която приема число - n връща като резултат списък с елементите от редицата на Фибоначи от 1 до n.

### **Пример**

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| 10 | [1,1,2,3,5,8,13,21,34,55] |
| 15 | [1,1,2,3,5,8,13,21,34,55,89,144,233,377,610] |
| 1 | [1] |
| 0 | [] |
| -1 | [] |

## **Зад. 6** *Факториел (списък)*

Дефинирайте функция, която приема число - n връща като резултат списък с елементи съответно факториел на числата от 1 до n.

### **Пример**

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| 10 | [1,1,2,6,24,120,720,5040,40320,362880] |
| 5 | [1,1,2,6,24] |
| 0 | [] |
| 1 | [1] |