

**КОНТРОЛНА РАБОТА № 1 ПО ФУНКЦИОНАЛНО ПРОГРАМИРАНЕ**  
**КН, 2-ри курс, 1-ви поток (24.11.2021 г.)**

**Задача 1.**

Да се дефинира функция (`get-distribution n`), която приема неотрицателно цяло число  $n$  и връща списък с елементи от вида (`<цифра> . <брой срещания в  $n^2$ >`), представящ разпределението на цифрите на  $n^2$ . Резултатът да е сортиран спрямо цифрите на  $n^2$ .

*Пример:*

`(get-distribution 123) → '( (1 . 2) (2 . 1) (5 . 1) (9 . 1) )`

**Задача 2.**

Даден е списък от непразни списъци. Елементите на дадения списък са такива, че ако сортирате списъка от техните дължини, ще получите списък с естествени числа. Те биха формирали строго нарастваща редица от поредни естествени числа, в която едно число липсва.

Да се дефинира функция (`get-missing-length xss`), която връща липсващото число за списъка `xss`. Ако някой от елементите на `xss` е празен или списъкът `xss` е празен, да се връща грешката `"Empty list!"`.

*Примери:*

`(get-missing-length '((1 2) (4 5 1 1) (1) (5 6 7 8 9))) → 3`

`(get-missing-length '(("a", "a", "a") ("a", "a") ("a", "a", "a", "a") ("a") ("a", "a", "a", "a", "a", "a"))) → 5`

**Задача 3.**

Да се дефинира процедура от по-висок ред (`trailing-zeros n`), която връща анонимна функция, която приема предикат  $p$  и проверява дали  $p$  е в сила за броя на влаещите нули в  $n!$ . Вместо да се пресмята факториелът, да се използва следното свойство: броят на влаещите нули в  $n!$  е равен на

$$\sum_{k \geq 1} \left\lfloor \frac{n}{5^k} \right\rfloor$$

където естественото число  $k$  пробягва интервала от 1 до най-високата степен на 5, която не надвишава  $n$ .

Например броят на влаещите нули в  $127!$  е:

$$\left\lfloor \frac{127}{5} \right\rfloor + \left\lfloor \frac{127}{25} \right\rfloor + \left\lfloor \frac{127}{125} \right\rfloor = 25 + 5 + 1 = 31$$

*Примери:*

`((trailing-zeros 6) even?)`  $\rightarrow$  `#f` (броят на влаещите нули е 1. 1 не е четно)

`((trailing-zeros 1000) even?)`  $\rightarrow$  `#f` (броят на влаещите нули е 249. 249 не е четно)

`((trailing-zeros 100000) even?)`  $\rightarrow$  `#f` (броят на влаещите нули е 24999. 24999 не е четно)

`((trailing-zeros 1000000000) even?)`  $\rightarrow$  `#t` (броят на влаещите нули е 249999998. 249999998 е четно)

#### Задача 4.

Да се дефинира функция `(persistence n)`, която приема естествено число  $n$  и връща точкова двойка от вида `(ys . x)`. Първият елемент на резултата е списък, чийто първи елемент е равен на произведението на цифрите на  $n$ , а всеки следващ е равен на произведението на цифрите на предходния до получаването на едноцифрено произведение, на което е равен последният елемент на `ys`. Вторият елемент на резултата (т.е. `x`) е равен на дължината на списъка `ys`.

*Примери:*

`(persistence 39)`  $\rightarrow$  `'((27 14 4) . 3)` ;  $3*9=27$ ,  $2*7=14$ ,  $1*4=4$

`(persistence 126)`  $\rightarrow$  `'((12 2) . 2)` ;  $1*2*6=12$ ,  $1*2=2$

`(persistence 4)`  $\rightarrow$  `'((4) . 1)`

`(persistence 999)`  $\rightarrow$  `'((729 126 12 2) . 4)`