

## Funktsionaalprogrammeerimise meetod

### proovieksami näidisülesanded

1. Defineeri funktsioon `split2`, mis jagab argumentlisti kaheks listiks nii, et esimeses listis on argumentlisti esimene, kolmas, viies jne. element ning teises on argumentlisti teine, neljas, kuues jne element:

```
split2 :: [a] -> ([a],[a])
```

Näiteks:

```
split2 [1..10] ==> ([1,3,5,7,9],[2,4,6,8,10])
```

2. Defineeri funktsioon `duplicate`, mis etteantud mittenegatiivsete täisarvude listis asendab iga elemendi iseenda arv kordi iseendaga:

```
duplicate :: [Int] -> [Int]
```

Näiteks:

```
duplicate [3,0,2,4] ==> [3,3,3,2,2,4,4,4,4]
```

3. Olgu antud järgmised Haskell'i definitsioonid:

```
s f g x = f x (g x)
k x y   = x
```

Redutseeri avaldis `s (k (+1)) (k 2) 4` normaalkujule kasutades *aplikatiivset järjekorda* (või normaaljärjekorda, või laiska väärtutamist).

4. Mis on avaldise `head (foldr map [3] [(+1),(+2)])` väärtus (vali *üks* alljärgnev variant):

(a) 3                      (b) 4                      (c) 5                      (d) 6

5. Mis on avaldise `(((+4) .) . (.(*3))) (+2) 1` väärtus (vali *üks* alljärgnev variant):

(a) 9                      (b) 13                      (c) 17                      (d) 21