

## Home Exercise Nr 5 - Functional Programming with Python

### Generators

1. The function evenprt

Write a function `evenprt(N1,N2,N3)` that receives three parameters, `N1`, `N2`, and `N3` which must be positive integers, such that  $N1 < N2$ , and `N3` must be less than the absolute difference between `N1` and `N2`. This function will print all the even numbers between `N1` and `N2` (including `N2`), `N3` numbers per line.

- a. Write a version of this function, which is not efficient in memory usage.
- b. Write a version of this function, which is efficient in memory usage.

2. The function primefactors

- a. In one of the previous chapters of this course, we learned about the function “`napa`” which implements Erathostenes’ algorithm for the calculation of prime numbers. Write a new version of the “`napa`” function, by using functional programming tools *only*.
- b. We want to write a function `primefactors(N)` which receives a positive integer `N`, and returns a list containing all the prime divisors of `N`. This function has to use the new version of the “`napa`” function, that you use in paragraph ‘a’ of the question. Write two versions of the function `primefactors`: a version written using Python generators, and the other one don’t.

3. Solve again all the exercises of Homework Nr 4, by using generators and/or generator expressions, if, according to your opinion, it is convenient to use them, especially if you think that those generators optimize place and time.