APiE Exercise - Complexity, Debugging, Optimizing

Write your answers into a report, and send it to `t.weinhart@utwente.nl`. Note the report guidelines on Blackboard.

Exercise 1 - Debugging (3pts)

The Sieve of Erathostenes is an algorithm that returns all primes from 2 to N. It was proposed by Eratosthenes in approx. 300 BC. It requires the following steps:

1. List all numbers from 2 to n in a sequence.
2. Take the smallest uncrossed number from the sequence, put it in your primes list and cross out all its multiples.
3. If n is uncrossed and the smallest uncrossed number is greater than $\sqrt{n}$, then n is prime, otherwise not.

Below you can find an implementation of the Sieve of Erathostenes that contains many coding errors (the file, `listofprime.m`, is available on Blackboard). Find and correct the coding errors and list them in your report.

Help: M-Lint (the thin colored bar to the right of your MATLAB editor) contains warnings and error messages. However, some errors can only be found by carefully reading the code. Further, you find a correct (and slightly optimised version) of the Sieve of Erathostenes by typing `edit primes` *in the MATLAB command window.*

```
function listofprimes(N)
%   listofprimes(N) Generates a list of prime numbers from 2 to N.
%
%   Input: scalar value N
%   Output: a row vector of prime numbers from 2 to N.

% check if input is a scalar
if length(N)==1
  error('N must be a scalar');
end

% check if input is greater or equal to 2; else return empty array
if N < 2,
  primes = [];
end

% create an array for the numbers from 2 to N
```

```
primes = 2:N;

% loop through all integers from 2 to sqrt(N)
for k = 2:sqrt(N)
  % setting a value of the array to 0 indicates the value is crossed out
  % check if the number k is not crossed out
  if primes(k)!=0
     % cross out all multiples of k starting from k^2, i.e.
     % cross out k^2, k^2+k, k^2+2*k, ...
     primes((k*k):k:length(primes)) = 0
  end
end
% return all values that are not crossed out
primes = primes(primes>0);
return primes;
```

Exercise 2 - Complexity (3pts)

Below you see a simple algorithm for the matrix product.

1. Evaluate the order of complexity for the case of square matrices in both time and memory for Algorithm 1. Explain your reasoning in your report.
2. Now compute the complexity of the matrix product in MATLAB: For two random square matrices $M, N \in \mathbb{R}^{n\times n}$ (`M=rand(n);`), calculate the matrix product (`P=M*N;`), and find the computing time t using `cputime`. Repeat this step for different values of n and plot t as a function of n in a log-log scale plot. Do you see the dependence on n as you expected? How large can n be s.t. you can still compute the product within 10 seconds?

Algorithm 1 A simple algorithm for the Matrix product

Input: $M \in \mathbb{R}^{a\times b}, N \in \mathbb{R}^{b\times c}$
Output: $P \in \mathbb{R}^{a\times c}$
  for $i = 1$ to a **do**
    for $j = 1$ to c **do**
      $P_{i,j} \leftarrow 0$
      for $k = 1$ to b **do**
        $P_{i,j} \leftarrow P_{i,j} + M_{i,k} \cdot N_{k,j}$
      end for
    end for
  end for

Exercise 3 - Optimization (2pts)

The code below returns the minimum distance to the origin of a set of points $(x_i, y_i, z_i), i = 1, \ldots, n$. Its speed has already been improved by preallocating the array d. The code however can still be improved.

1. Use the MATLAB profiler to find out which lines require most of the computing time.
2. Try to optimize the code by using matrix-vector operations as much as possible. Report on the speed improvement.

```
function d = minDistance(x,y,z)
% Given a set of 3D points specified by column vectors x,y,z, this
% function computes the minimum distance to the origin

nPoints = length(x);
% preallocate
d = zeros(nPoints,1);

% compute distance for every point
for k = 1:nPoints
      d(k) = sqrt(x(k)^2 + y(k)^2 + z(k)^2);
end

% get minimum distance
d = min(d)
```

Exercise 4 - Optional (only for higher grades) (2pts)

Repeat exercise 2 for the MATLAB `perms` algorithm. You can find the source code by typing `edit perms` into your command window.