

## Laboratory 5

### – Parallelizing techniques –

#### 1. Goal:

The goal of this lab is to implement a simple but non-trivial parallel algorithm.

#### 2. Requirement:

Perform the multiplication of 2 polynomials. Use both the regular  $O(n^2)$  algorithm and the Karatsuba algorithm, and each in both the sequential and the parallelized form.

#### 3. Computer Specification:

CPU: Intel(R) Core(TM) i7-8750H CPU @ 2.20GHz

RAM: 16 GB

System Type: 64-bit

#### 4. Implementation:

Algorithm:

- a. Regular polynomial multiplication:
  - Complexity:  $O(n^2)$
  - Step 1: Distribute each term of the first polynomial to every term of the second polynomial (multiply the coefficient and add the exponents)
  - Step 2: Combine like terms
- b. Karatsuba algorithm:
  - Complexity:  $O(n^{\log_2 3})$
  - A fast multiplication algorithm that uses divide and conquer approach to multiply two numbers

Parallelization – Java Thread Pool with custom Runnable class to achieve parallelization.

#### 5. Performance Test:

- Note: level 'x' = polynomials of rank  $x \cdot 100$

Algorithm	Level 1	Level 5	Level 10	Level 50	Level 100
Regular sequential	4	19	30	297	996
Regular parallelized	3	30	61	121	505
Karatsuba sequential	12	41	85	450	1373
Karatsuba parallelized	39	55	77	111	376

#### 6. Sum up:

- The parallelized methods run faster than the sequential ones (as expected).
- Karatsuba runs faster than the regular one on bigger polynomials (for smaller ones the splitting takes more time than computing the result).