1 Parallel processing of vectors and matrices

Restructure the implementation of exercises 1,3,4,5 of PL4 (working with vectors and matrices) with Rust threads and concurrency mechanisms in the standard (or parking_lot) libraries.

Keep the same interface to reuse the code in main to test the functions.

2 Area of Mandelbrot Set (1)

Implement the parallel function to calculate the area of the Mandelbrot Set with tasks and the Threadpool library.

3 Working with parallel iterators and closures

Restructure the implementation of exercise 7 of PL4, with parallel iterators, using Rayon.

4 Cholesky decomposition

a. Parallelize the Cholesky decomposition of PL4 with Rayon.

b. Using Instant and Duration from the std::time module, compare the elapsed time of the with the sequential implementation of PL4. What can you conclude?

5 Area of Mandelbrot Set (2) (Group Evaluation #2)

a. Implement the parallel function to calculate the area of the Mandelbrot Set, with Rayon libraries.

b. Using Instant and Duration from the std::time module, compare the elapsed time of the implementation (also with the Threadpool implementation of exercise 2 and the sequential implementation of PL4). What can you conclude?

c. Compare also with the OpenMP implementations. What can you conclude?

Credits:

- Version 1.0, Luis Miguel Pinho, with inputs from:
  - The Rust Programming Language, by Steve Klabnik and Carol Nichols, with contributions from the Rust Community, https://doc.rust-lang.org/stable/book/title-page.html
  - Rust by Example, https://doc.rust-lang.org/stable/rust-by-example/index.html
  - Learning Rust, https://learning-rust.github.io/
  - Rust Cookbook, https://rust-lang-nursery.github.io/rust-cookbook/
  - Code examples for the book Programming Rust, https://github.com/ProgrammingRust