1 Working with vectors

Given a vector (Vec) of integers:

a. Implement functions that return the max, min, average
b. Implement a function that returns the median (when sorted, the value in the middle position), it should fail if the vector is not sorted
c. Implement a function to create a vector by adding the elements in two other vectors v3[i] = v1[i] + v2[i]
d. Implement a function that sorts the array using quick sort
e. Implement a function that sorts the array using bubble sort
f. Change the median function to calculate the median by sorting a copy of the array, if it is not sorted

For each function implement code in main to test its use.

2 Working with structs

Implement a library that provides complex numbers defined as a struct of two floating point values. Implement functions to initialize, add, subtract, multiply, divide.

Implement a program to test the library.

3 Working with matrices

Implement a function that performs the multiplication of 2 matrices (2-D arrays) of fixed sizes.

4 Working with vectors (2)

Add to the vectors exercise a function to calculate the mode (the value that occurs most often) using a hash map.

5 Working with matrices (2)

Implement a function that performs the multiplication of 2 dynamic matrices, using the DMatrix type from the nalgebra library.

6 Working with iterators and closures (1)

a. Using iterators, implement a function that calculates the sum of all numbers from 1 to 100
b. Using iterators and closures, implement a function that calculates the sum of all squares lower than a “upper_bound” value

7 Working with iterators and closures (2)

Reusing the Complex struct of exercise 2, implement a library of functions to work with a vector of Complex, with functions:

a. Implement functions that return the max, min, average
b. Implement a function to return the sum of the elements in the vector
c. Implement a function to create a vector with the modules of the complex numbers

Use iterators and closures in the implementation.

8 Area of Mandelbrot Set

Implement a function to calculate the estimate of the area of the Mandelbrot set, with iterators.

9 Analyzing and Evaluating

Using Instant and Duration from the std::time module, compare the elapsed time of the exercises 3 and 5 (with the same matrix sizes). Compare also with the sequential C implementation. What can you conclude?

Extra Lab (Students may advance further outside of lab)

Using the the DMatrix type from the nalgebra library, implement a function that performs the Cholesky decomposition.

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