Structure

This project is split in two environments, and consequently two folders: client and server. Both environments are configured with Cmake, and will work out of the box on any linux distribution (windows and macOS untested) with an up-to-date Cmake installation.

Usage

Client

Start the client by running `{projectRoot}/client/build/client` with no arguments, assuming the app is already built. This will open a command prompt with whom you can interact to initiate client-side actions.

Type in `commands` to get a list of all available commands and their description. When it comes to our use-case, we won't need anything other than the following four directives:

```
test
connect
send
disconnect
```

Here are some examples of usage:

```
test 127.0.0.1 8080 // test <ip> <port>
connect 127.0.0.1 8080 // connect <ip> <port>
send one-word-message // if the message doesn't contain any spaces
send "multi word message or command"
disconnect // disconnect from the server you are currently connected to
```

Server

Once you fully understand how the client works, you'll need to get the server up and running. Again, assuming it is already built, run `{projectRoot}/server/build/server <ip> <port>` , `<ip>` and `<port>` being the IP address and port number you want your server to be reachable at.

You should see a short log message similar to the one below if everything did start properly:

```
Server started on XXX.XXX.XXX.XXX:XXXX
```

In case of failure, try another port, it's often caused by another application already using it.

Requests

Currently, the server supports three command requests:

```
std::vector<Command> _commands = {
    {"erato", runEratosthenes},
    {"matrix", runMatrixMult},
    {"sort", runBubbleSort}
  };
```

Use the keyword on the left, followed by the correct number of arguments to get the result of the computation as a response.
Here are again some examples:

```
Remote@127.0.0.1:8080> send "sort 1,2,1,9,4"
Response: [1, 1, 2, 4, 9]
```

```
Remote@127.0.0.1:8080> send "erato 10"
Response: [2, 3, 5, 7]
```