

# CMake Developer Test Task

This task consists of 3 parts. Parts 1 and 2 should take less than 8 hours to complete. Part 3 should also take you another 8 hours. Regardless of your final application status, you will be compensated for your effort if it meets the specification criteria.

You will be judged by the final output (see Deliverables).

## 1. Project

Setup a simple C++ project that has the following elements:

- `cgcustommath` dynamic library with one function and dependency on Eigen3 library managed through Conan.
- `mat` executable.

Don't spend too much time on this, as writing the code will not be part of your job. Bare minimum will suffice, you don't need to handle user input errors or write unit tests.

### 1.1 `cgcustommath` library

This library should contain a single function:

```
Eigen::Array22f matrix(int a, int b)
```

This function should:

- Create a 2x2 identity matrix.
- Multiply the matrix by `a`.
- Add `b` to the matrix.
- Return the resulting matrix.

### 1.2 `mat` executable

This executable should

- Take 2 command line parameters.
- Pass them to the `matrix` function as `a` and `b`.
- Print the resulting matrix on screen.

## 2. CMake and Debugging

- Create a CMake project that will compile and build both elements.
- Create a script that will build an XCode project containing both elements and allow for debugging each target.
- Create build configurations and cmake presets that MS Visual Studio will recognize when opening as a folder.
- You must use `msvc` on Windows and `clang` on the Mac.
- Provide `readme.md` file with instructions how to build/open project on Win and Mac.

### 3. Packaging and installers

Installer needs to install the framework/dll alongside other system libraries and the executable in a folder selected by the user.

You need to:

- Create a Python script that will build an installer on Windows using WixToolset.
- Create a Python script that will build a dmg installer on the Mac.
- Use Python 3.10 with type hints and doc strings.
- Code documentation doesn't need to be extensive, but it needs to be legible and meaningful.
- You can use 3rd party tools in your scripts but they must be easily installed using pip, brew or choco. If you do use them, please provide a `setup` script that will make sure all required tools are installed.
- Provide readme.md file with instructions how to build installers on Win and Mac.

### 4. Deliverables

You need to deliver the following:

- Git repo with code.
- Screencast of debugging `matrix` function on each platform - add a breakpoint inside the function and show that you can hit it when running the program in XCode and Visual Studio.
- Screencasts of building installers and running installation on both machines (Win/Mac).
- We must be able to run the installer scripts on our machines by following your instructions.

For screencasts you may use QuickTime recording on the Mac and OBS Studio on Win, unless you have other preferences.

If you have questions, ask them to [bart@creativeimpatience.com](mailto:bart@creativeimpatience.com).

Good luck!