General Rules and Instructions

Plagiarism Note and Late Policy

Copying code (either from other students or from external sources) is strictly prohibited! We will be using automatic anti-plagiarism tools, and any violation of this rule will make your submitted work null. It is recommended to respect the submission deadlines. Late submissions will be accepted, though, but no assistance will be provided after the deadline.

Provided Libraries

For each assignment, you will use the geometry processing library libigl, which includes implementations of many of the algorithms presented in class. You are advised to look over the relevant tutorials before starting the implementation for the assignments; you are also encouraged to examine the source code of all the library functions that you use in your code to see how they were implemented. In particular you will be using the python bindings of igl.

Alternatively, if you wish to develop your assignments in C++, this is possible, but you will not receive any assistance. In this case, you shall develop a cmake project based on the template provided with the C++ version of the library, and shall make sure your code compiles and executes under MacOS.

The libigl library includes a set of tutorials, an introduction to which can be found in the two previous links. You are advised to look over the relevant tutorials before starting the implementation for the assignments; you are also encouraged to examine the source code of all the library functions that you use in your code to see how they were implemented.

No libraries apart from libigl, numpy, meshplot, and scipy are permitted unless permission is granted in advance.

Installing igl in Python

- Install Anaconda from https://www.anaconda.com/
- In a terminal, create a Python virtual environment with the following command:

    ```
    conda create -n myenv conda-forge::igl conda-forge::meshplot numpy scipy
    ```

  where you substitute myenv with a name of your choice for this environment. If you work on an ARM architecture (new Mac), use the following command instead:

    ```
    conda create -n myenv --platform osx-64 conda-forge::igl
                    conda-forge::meshplot numpy scipy
    ```

- To develop and run your code, activate the environment with `conda activate myenv`
- To leave the environment, type `conda deactivate`
- You can run your code from an IDE (e.g., Visual Studio Code) by specifying the environment
- We will use jupyter notebooks: you can run the notebook inside an IDE, or launch it from the terminal (`jupyter notebook`) and use it in the web browser. Environment specification is necessary for the browser version, too (top right button in the jupyter window).