Assignment 0: Setting up Nori (1 point)

Deadline: 2021-03-31 23:59

In this task, you will take your first steps with Nori and create your first integrator.

Setting up Nori (1 point)

Download the framework provided in the submission GitLab and set up a working environment. You will at least need `git`, `cmake` and a reasonably recent version of a good C++ compiler (MSVC, gcc or similar). Get to know Nori by following the steps for Assignment 1 at `https://wjakob.github.io/nori/`. You can skip the instructions regarding GitHub and GitHub classroom since we use our own GitLab (`https://submission.cg.tuwien.ac.at`). Your first steps will look something like:

```
git clone git@submission.cg.tuwien.ac.at:Your/Repository.git assignments
cd assignments
git submodule update --init --recursive
cd ..
mkdir assignments_build # or continue with your IDE of choice..
cd assignments_build
cmake -S ../assignments -B ./
make -j8
```

Make sure to implement the `NormalIntegrator` class as instructed. However, instead of using the `abs` function for the integrated color value, compute it as $\frac{n}{2} + 0.5$. This makes debugging easier, but obviously the resulting image will look different. Note that our test files are in slightly different locations. The bunny test files are located in `scenes/assignment0`.

Submission format

To store or submit your code, please use our own, institute-hosted submission Gitlab `https://submission.cg.tuwien.ac.at`. You will receive a mail with your account and assignment repository as soon as they are ready. **The `master` branch is for development only.** You should push there while you are experimenting with the assignment and don't want to lose your work. Once your solution works and you believe it is ready to be graded, please use the branch `submission<X>` where `<X>` is the assignment number. **E.g., in order to submit your solution for the intro assignment (0), push to `submission0`.**

If you push to a `submission` branch, the server will trigger automatic compilation and some testing for your code. You can track the state of new submissions being processed on the GitLab page for your repository under "CI/CD > Pipelines". If a stage fails, click on it to receive additional output and system information from the executing server. If everything worked, you will shortly find a report with your test results in the "CI/CD" pipeline section, when checking the artifacts of the "report" stage. You can submit multiple times until the deadline, but don't clog the system by, e.g., using the submission server for debugging. The last submission that was pushed before the deadline counts, regardless of the results from automatic testing. They are only meant for your convenience and to provide some automated feedback.

Please make sure to NOT add unnecessary files (project folders, temporary compiler results), as your application will be created from your code and CMake setup only. Examples of files that are usually relevant:

- **changed or added** CMakeLists.txt files
- **changed or added** code files (.h, .cpp)
- **changed or added** test cases if you want to show off advanced solutions

Make sure to keep the directory structure in your submitted archive the same as in the framework.

Words of wisdom

- The framework is using Eigen under the hood for vectors and matrices etc. Be careful when using `auto` in your code (Read here why).

- If you have questions, please use TUWEL, but refrain from posting critical code sections.

- To reduce the waiting time, Nori runs multi-threaded by default. To make debugging easier, you will want to set the number of threads to 1. To do so, simply execute Nori with the additional arguments `-t 1`.