Assignment 1: Transformation

Submission Period: 10.05.2022 12:00:00 - 24.05.2022 12:00:00 (Central European Summer Time)

General Information

- This is one of the graded assignments. In this assignment, one can collect a maximum of 10 points. **A submission that cannot be compiled will not be graded.**
- There are two different aids for that may be helpful for accomplishing the assignment. 1) a certain visual effect that is described in detail in the assignment, and can be verified visually using `npm start`; or 2) provided tests that can be executed using command `npm test`.
- Note that **passing all provided tests, or achieving a similar visual effect does not represent one can collect all points**. The evaluation of submissions will also run additional tests that are not provided to check the general robustness and soundness of a submission, such as possible edge cases. We recommend carefully considering an implementation if desire more points.
- **Dependent tasks assume previous results to be correct**. This means a subsequent implementation that depends on a previous incorrect implementation is also considered incorrect.
- It is prohibited to exchange solutions for the assignments with other students during the examination period. You must work on the assignments alone and independently and submit your own solution. If we discover any fraud or plagiarism in the submission, **both parties will be excluded** from the exam.
- If you found the task description ambiguity or potential mistakes in the provided code skeleton, please contact cg1ss22@medien.ifi.lmu.de or ask in the tutorial class for further clarification.

Erklärung über die eigenständige Bearbeitung

Ich erkläre hiermit, dass ich die vorliegende Arbeit vollständig selbstständig angefertigt habe. Quellen und Hilfsmittel über den Rahmen der Vorlesungen/Übungen hinaus sind als solche markiert und angegeben. Ich bin mir darüber im Klaren, dass Verstöße durch Plagiate oder Zusammenarbeit mit Dritten zum Ausschluss von der Veranstaltung führen.

Task: Write A Simple Math Library (10 Points, Easy)

In this task, you are going to implement a basic linear algebra library for `Vec4` and `Mat4` classes. Look for `// TODO:` comments in the `src/math/vec4.ts` and `src/math/mat4.ts` files.

The class `Vec4` has four components `(x, y, z, w)`. The implementation is a homogeneous representation of a 3D point or a 3D vector. The library should throw an error if an operation is not applicable for given parameters, e.g. cross product can only be applied to two vectors. Similarly, the class `Mat4` represents a 4x4 matrix and uses row major representation[1] internally.

- (0.5p) Implement `Vec4.add()` to compute point/vector addition
- (0.5p) Implement `Vec4.sub()` to compute point/vector subtraction
- (1p) Implement `Vec4.dot()` to compute the vector dot product
- (1p) Implement `Vec4.cross()` to compute the vector cross product
- (1p) Implement `Vec4.len()` to compute the length of a given vector
- (1p) Implement `Vec4.unit()` to compute the unit vector of a given vector
- (1p) Implement `Vec4.apply()` to apply a given matrix transformation to the point/vector
- (0.5p) Implement `Mat4.add()` to compute matrix addition
- (0.5p) Implement `Mat4.sub()` to compute matrix subtraction
- (1p) Implement `Mat4.mulM()` to compute matrix multiplication
- (1p) Implement `Mat4.mulV()` to compute point/vector matrix multiplication
- (1p) Implement `Mat4.T()` to compute the transposed matrix

You can run the project by 1) installing all dependencies using `npm i` then 2) start and test your implementation using `npm test`.

[1] See https://en.wikipedia.org/wiki/Row-_and_column-major_order

Submission Instructions

Please use the provided submission template and follow the submission instruction below to submit your solution to Uni2Work.

- Delete the two folders: **`node_modules`**, and **`build`**.
- Rename your folder to `cg1-assignment1-<your matriculation number>`, and compress everything as a single `.zip` file. For example, if your matriculation number is 12345678, then the zip-file's filename should be `cg1-assignment1-12345678.zip`.

  ✔ `cg1-assignment1-12345678.zip`
  ✘ `cg1-assignment1-<12345678>.zip`

  Your folder structure should be exactly like this (except the matriculation number):

```
cg1-assignment1-12345678/
├── .eslintignore
├── .eslintrc.json
├── .prettierrc.js
├── .vscode
│   └── settings.json
├── README.pdf
├── jest.config.js
├── package-lock.json
├── package.json
├── src
│   └── math
│       ├── mat4.ts       <- your code
│       ├── utils.ts
│       └── vec4.ts       <- your code
├── test
│   ├── math.test.ts
│   └── utils.ts
├── tsconfig.json
└── webpack.config.js
```

  If you could not find files starting with a dot, e.g., `.eslintignore`, please configure your operating system to show hidden files[2].

[2] See https://www.google.com/search?q=how+to+show+hidden+files