LMU

LUDWIG-
MAXIMILIANS-
UNIVERSITÄT
MÜNCHEN

INSTITUT FÜR INFORMATIK
MEDIENINFORMATIK
PROF. DR. ANDREAS BUTZ, CHANGKUN OU, FLORIAN LANG
COMPUTERGRAFIK 1, SOMMERSEMESTER 2021

Graded Assignment 01: Transformation

Submission Period: 30.04.2021 00:00 - 04.05.2021 23:59 (Anywhere on Earth, AoE)

General Information

- This is one of the graded assignments. You need to collect above 50 points to pass with 4.0, and a score of 90 points or greater for a 1.0. In this graded assignment, you can collect a maximum of 10 points.
- Please **sign "Erklärung über die eigenständige Bearbeitung" in this document. A submission without electronic signature will not be graded, and a re-submission is *NOT* possible.** (Hint: You can use the macOS's built-in application Preview or Adobe Acrobat Reader DC on Windows. Handwritten signatures in a re-scanned document will not be processed.)
- It is prohibited to exchange solutions for the graded assignments with other students during the examination period. You must work on the graded assignments alone and independently and submit your own solution. If we discover any fraud or plagiarism in the submission, you will be asked to join an additional oral exam. In the worst case, both parties will be excluded from the course.
- We designed the assignment for and recommend you to invest **1 hour** to work on this assignment. You can work on the assignments in German or English. Mixing both languages is allowed as well.
- If you have any questions regarding technical issues, please contact the assistants of the course immediately. The fastest way to do so, is the discussion forum in Moodle.

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.

04.05.2021 Moritz Kilgenstein

Ort, Datum Unterschrift Moritz Kilgenstein

Ich habe insgesamt etwa _____ Stunden Arbeitszeit für diese Abgabe aufgewendet. (Diese Information ist freiwillig, rein informativ und hat keinen Einfluss auf die Benotung.)

1 Task 1: Theory (5 Points, Easy)

In this task, you are going to answer theoretical questions regarding transformations. Please write your answers into the `ANSWER.md` file. Keep your answers short.

1.1 Formulas (2 points)

Let $\mathbf{A} \in \mathbb{R}^{m \times p}$ and $\mathbf{B} \in \mathbb{R}^{q \times n}$ be matrices and $a_{i,j}, b_{i,j}$ their elements. From the tutorial you know the following formula:

$$c_{i,j} = \sum_{k=1}^{p} a_{i,k} b_{k,j} \text{ for } 1 \leq i \leq m, 1 \leq j \leq n \tag{1}$$

where $c_{i,j}$ are the elements of matrix $\mathbf{C}$ in the i-th row and j-th column. Answer **very briefly**:

a) (0.5p) What does this formula calculate?

b) (0.5p) What is the relation between p and q? What happens if that relation is broken?

c) (0.5p) What is the relation between m and n? What happens if that relation is broken?

d) (0.5p) What is the size of matrix $\mathbf{C}$?

1.2 Transformations (3 points)

Consider the point $p = (x, y, z)^\top = (2, 0, 0)^\top$. We now want to move this point to the origin $(0, 0, 0)^\top$ using the following three transformations:

$$f(p) = \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ -2 \end{pmatrix} \qquad g(p) = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} \qquad h(p) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

Here, $f(p)$ translates the point by $(0, 0, -2)^\top$, $g(p)$ rotates the point by 90° around the z-axis, and $h(p)$ rotates the point by 90° around the x-axis. Each transformation can only be applied once. Answer **very briefly**:

a) (1p) What is the sequence of transformations?

b) (0.5p) Does it matter if the coordinate system is left-handed or right-handed? If yes, which kind do we need?

c) (1.5p) Name one problem that can arise if we combine rotations around all axes using Euler angles and briefly explain why this happens and what the result is.

2 Task 2: Practice (5 Points, Easy)

In this task, you are going to implement a basic linear algebra library for `Vector` and `Matrix` classes. Look for `// TODO:` comments in the `src/linalg/vec.ts` and `src/linalg/mat.ts` files. While doing the task, please ensure that ESLint auto fix is activated. It helps you to find problems in the code and can format the code automatically (as discussed in tutorial 1, page 29-31).

The class `Vector` implements the interface `IVector` and has four components `(x, y, z, w)`. The implementation is a homogeneous representation of a three dimensional point or a three dimensional vector. The library should throw an error if an operation is not applicable for the given parameters, e.g. cross product can only be applied to two vectors. The class `Matrix` implements the interface `IMatrix`, which represents a 4x4 matrix.

- (0.5p) Implement `Vector.add()` to compute point/vector addition
- (0.5p) Implement `Vector.sub()` to compute point/vector subtraction
- (0.5p) Implement `Vector.dot()` to compute the vector dot product
- (0.5p) Implement `Vector.cross()` to compute the vector cross product
- (0.5p) Implement `Vector.unit()` to compute the unit vector of a given vector
- (0.5p) Implement `Vector.apply()` to apply a given matrix transformation to the point/vector
- (0.5p) Implement `Matrix.add()` to compute matrix addition
- (0.5p) Implement `Matrix.sub()` to compute matrix subtraction
- (0.5p) Implement `Matrix.mul()` to compute point/vector matrix multiplication or matrix matrix multiplication
- (0.5p) Implement `Matrix.T()` to compute the transposed matrix

Here are some hints:

- You can run the project by 1) installing all dependencies using `npm i` then 2) start and execute the project using `npm start`. Your browser will open a tab automatically and the console prints the result of the code from `src/main.ts`.
- You can use or change the code in `src/main.ts` for testing purpose
- We recommend you to get started with tasks in `src/linalg/vec.ts`, then `src/linalg/mat.ts`.

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`.
- For example, your folder structure should be exactly like this (except the matriculation number):

```
cg1-assignment1-12345678/
    ├── ANSWER.md             <- your text answers
    ├── .eslintignore
    ├── .eslintrc.json
    ├── jest.config.js
    ├── package.json
    ├── package-lock.json
    ├── .prettierrc.js
    ├── README.md
    ├── README.pdf            <- your signature
    ├── src
    │   ├── linalg
    │   │   ├── mat.ts        <- your code
    │   │   ├── utils.ts
    │   │   └── vec.ts        <- your code
    │   └── main.ts
    ├── tsconfig.json
    ├── .vscode
    │   └── settings.json
    └── webpack.config.js
```