

Computer Grafik 2019 - Übungsbetrieb

In den Übungen zur Vorlesung Computer Grafik bauen Sie Ihren eigenen Grafik Renderer in Java. Damit Sie sich auf das Implementieren von einzelnen Kern-Algorithmen konzentrieren können wird eine Testumgebung vorgegeben. Diese gibt das Design des Renderers vor. Basierend auf dieser Ausgangslage werden Sie im Laufe des Semesters aufeinander aufbauende Aufgaben lösen. Es ist daher wichtig dass Sie alle Übungsblätter bearbeiten. Es werden keine Musterlösungen heraus gegeben.

Generelle Informationen

Lesen Sie dieses Informationsblatt aufmerksam durch.

Ausgabe, Abgabe und Vorzeigen

Die Übungsblätter werden donnerstags auf der Vorlesungswebseite veröffentlicht. Die Lösungen müssen im Tutorat vorgezeigt werden. Programme die nicht kompilieren oder nicht lauffähig sind werden generell mit 0 Punkten bewertet. Der Sourcecode muss jeweils am Tag vor dem Tutorat (23:59) über das courses System abgegeben werden. Nur Lösungen die abgegeben wurden dürfen im Tutorat vorgezeigt werden.

Fragestunde

Für die meisten Übungsblätter stehen zwei Wochen für die Bearbeitung zur Verfügung. In der Zwischenwoche wird es jeweils eine Fragestunde geben. Bei Fragen ausserhalb dieser Stunden bitte frühzeitig per Email bei den Tutoren melden.

Gruppen

Am Anfang des Semesters werden Zweiergruppen für die Bearbeitung der Übungen gebildet. Jeder Gruppe wird ein Zeitslot für das Besprechen der Lösung zugewiesen (Fr. 13:15 - 16:00). Diese Slots sind fix, es können keine Termine ausserhalb dieser Zeiten vereinbart werden. Wichtig: Die gesamte Gruppe muss für das Vorzeigen präsent sein! Nichterscheinen wird mit 0 Punkten bewertet. Alle in der Gruppe müssen in der Lage sein die Aufgaben zu erklären, die Bewertung ist individuell.

Zulassung zur Prüfung

Um an die Prüfung zugelassen zu werden, müssen 75% der insgesamt möglichen Übungspunkte erreicht werden. Die Bewertung ist individuell. Es wird kein Bonusblatt geben. Achtung: Nicht jedes Blatt ergibt gleich viele Punkte.

Plagiate

Kopieren von Code anderer Gruppen (auch aus den Vorjahren) ist untersagt. Bei Plagiatsfällen wird die gesamte Übung generell mit 0 Punkten bewertet, im Wiederholungsfall darf nicht an der Prüfung teilgenommen werden. Da wir nicht überprüfen können wer von wem kopiert hat, gelten diese Sanktionen jeweils für beide Gruppen. Im Zweifelsfall vor der Abgabe bei den Tutoren melden.

Information zur Übungsumgebung

Beachten Sie folgende Regeln für die Bearbeitung der Übungen. Nichtrespektieren kann zu Punkteabzug führen.

Abhängigkeiten

Für die Übungsumgebung benötigen Sie JDK 8 oder höher. Den Source Code finden Sie in folgendem Bitbucket Repository:

<https://bitbucket.org/gravis-unibas/computer-grafik-2019>

Eine Einladung zum Repository wird Ihnen in der ersten Vorlesungswoche zugeschickt.

Entwicklungsumgebung

Als IDE empfehlen wir IntelliJ IDEA.

<https://www.jetbrains.com/idea/>

Es steht Ihnen jedoch frei andere IDEs zu benutzen. Beachten Sie dass Sie in der Lage sein müssen die laufenden Programme im Tutorat vorzuführen.

Entwicklung

Stellen, an denen Sie Code einfügen müssen, sind stets markiert durch:

```
//TODO: Blatt X, Aufgabe Y
```

Nehmen Sie keine Änderungen an den mitgelieferten Klassen ausserhalb der markierten Stellen vor.

Für jedes Aufgabenblatt gibt es im package `exercises` eine Klasse mit einer `main` Methode. Rufen Sie die `main` Methode dieser Klasse ohne zusätzliche Argumente auf, wenn Sie ihre Implementierung testen wollen.

Courses Abgabe

Bei der Abgabe über das Courses System nur die veränderten Quellcodedateien als Archiv hochladen. Dabei alle Dateien ohne Unterordner in ein Archiv packen. Nur ZIP Archive hochladen.

Punkteverteilung

Aufgaben, die nicht mit einem Stern \star markiert sind, sind grundsätzlich freiwillig. Die markierten Aufgaben sind solche, auf denen zukünftige Übungsblätter aufbauen werden. Die \star -Aufgaben decken jedoch nicht unbedingt die Mindestpunktzahl ab. Wenn dies so ist, können Sie aus den verbleibenden Aufgaben auswählen, um die Mindestpunktzahl zu erreichen.

Utils

In den Übungen werden Sie häufig mit Vektoren und Matrizen arbeiten. Im mitgelieferten package `utils` finden Sie einige nützliche Klassen dafür, bspw. `Vector2` für einen Vektor $v \in \mathbb{R}^2$ oder `Matrix3` für eine Matrix $M \in \mathbb{R}^{3 \times 3}$.