

Übungsblatt 7 – Multimediaprogrammierung

Achtung: Zur Lösung dieser Übungsaufgabe dürfen ausschließlich die Module verwendet werden, die mit der Standardinstallation von Python 2.7.3 und Pygame 1.9.1 kommen. Abgaben, die zusätzliche Module enthalten oder nicht in Python 2.7.3 geschrieben sind erhalten automatisch null Punkte.

Zur Lösung der Programmieraufgaben können Sie Ihre Lösung aus Übungsblatt 6 verwenden. Alternativ können Sie auch die Beispiellösung von der Webseite verwenden.

Aufgabe 1: Eine Klasse für Sprite-Animationen (6 Punkte)

Erstellen Sie ein Modul „spriteanimation.py“. Definieren Sie in diesem eine Klasse „Animation“. Die Klasse soll folgende Eigenschaften erfüllen. Achten Sie auf die korrekte Schreibweise aller Methoden etc.

- a) Der Konstruktor erwartet als Argument eine beliebig lange Liste bzw. ein beliebig langes Tupel von Surface-Objekten. Jedes Surface steht für ein Frame der Animation (in der definierten Reihenfolge).
- b) Die Klasse hat folgende Methoden:
 - *set_frames(list)* ersetzt die aktuellen Frames durch eine neue Liste (bzw. Tupel).
 - *add_frame(Surface, pos)* Setzt ein neues Frame (Surface) an einer Position (ein Integerwert mit $0 \leq pos \leq \text{Länge der Liste}$) in die Animation ein. Entspricht die Position der Länge der Liste, dann wird ein weiteres Frame an die Animation angehängt. Ist sie kleiner, dann wird das Frame an dieser Position eingefügt und die folgenden Frames um eine Position verschoben. Wird pos nicht angegeben, wird das Frame automatisch hinten an die Liste angehängt.
 - *replace_frame(Surface, pos)* ersetzt das Frame an einer angegebenen Position durch ein neues Surface.
 - *remove_frame(pos)* entfernt das Frame der angegebenen Position.
 - *set_framerate(rate)* bestimmt wie viele Frames pro Sekunde angezeigt werden sollen. Wurde diese Methode nicht ausgeführt, dann wird als Standardwert 30 Frames pro Sekunde angenommen.
 - *get_frame(time_passed)* liefert ein Surface zurück, das dem aktuellen Frame entspricht. Das Argument *time_passed* gibt an, wie viele Sekunden seit dem letzten Aufruf der Methode vergangen sind.
 - *reset_animation()* setzt die Animation zurück auf das erste Frame.
- b) Animieren Sie nun die Explosionen der Kreise in Übungsblatt 6 mit Sprites. Suchen Sie eine beliebige Explosionsanimation im Internet oder erstellen Sie diese selber. Das Sprite-Sheet muss aus mindestens 3 Frames bestehen. Diese wird jedes Mal abgespielt, wenn ein Kreis durch eine Bombe zerstört wird.

Sprite Sheets für Ihr Spiel finden Sie z.B. hier:

<http://sdb.drshnaps.com/>

http://www.themysticalforestzone.com/Sprite_section.htm

Beachten Sie, dass Sie die Sprite Sheets eventuell nachbearbeiten müssen, um einheitliche Größen und Abstände der einzelnen Bilder zu erhalten.

Achten Sie darauf, dass sich alle Bilddateien in Ihrer Abgabe befinden. Um Probleme auf verschiedenen Systemen zu vermeiden sollten sich die Bilddateien im gleichen Verzeichnis wie Ihr Skript befinden. Geben Sie außerdem die Quellen zu Ihren Sprite Sheets und Sounds in einer Datei „copyright.txt“ ab.

Erstellen Sie in Ihrer Abgabe einen Ordner „aufgabe1“, der die Lösung zu dieser Aufgabe enthält.

Aufgabe 2: Sound (4 Punkte) (für alle nicht-KuMler)

Erweitern Sie nun Ihr Spiel (Aufgabe 1) um Sound. Folgende Sounds sollen abgespielt werden:

- Während dem Spiel läuft Musik in einer Endlosschleife.
- Die Explosionen erzeugen ein Explosionsgeräusch.
- Das Abfeuern der Bomben erzeugt ein passendes Geräusch.

Erstellen Sie in Ihrer Abgabe einen Ordner „aufgabe2“, der die Lösung zu dieser Aufgabe enthält.

Aufgabe 3: Schlüsselbildanimation (4 Punkte) (nur Kunst und Multimedia)

Sie kennen den Begriff Schlüsselbild (keyframe) schon aus der Videokompression. Hierbei werden nur einige der Teilbilder eines Videos vollständig gespeichert. Zwischenbilder werden anhand bestimmter Regeln berechnet.

Eine ähnliche Technik wird auch bei der so genannten Schlüsselbildanimation (keyframe animation) verwendet. Erklären Sie den Begriff Schlüsselbildanimation. Gehen Sie hierbei auch auf den Begriff der Interpolation ein. Bei welcher Art von Animationen bietet sich diese Technik besonders an? Bei welchen Animationen stößt sie schnell an ihre Grenzen?

Skizzieren Sie ein Beispiel, bei dem sich diese Art der Animation besonders gut anbietet. Zeichnen Sie dieses Beispiel (die Schlüsselbilder).

Nennen Sie ein Beispiel aus der Übung, bei dem diese Art der Animation angewendet wurde.

Erstellen Sie in Ihrer Abgabe einen Ordner „aufgabe3“, der die Lösung zu dieser Aufgabe enthält. Die Abgabe der Lösung muss als Textdatei (Endung .txt) bzw. als PNG-Datei erfolgen. Andere Abgaben werden nicht berücksichtigt.

Abgabe

„nur Kunst und Multimedia“ bedeutet, dass diese Aufgaben nur von Studierenden des Faches Kunst und Multimedia abgegeben werden müssen. „für alle nicht-KuMler“ bedeutet, dass diese Aufgaben von allen Studenten abgegeben werden muss, die nicht Kunst und Multimedia studieren.

Bitte geben Sie Ihre Lösung als ZIP-Datei bis zum 18.06.2012 10:00 Uhr im UniWorx Portal (<https://uniworx.ifi.lmu.de/?action=uniworxCourseWelcome&id=51>) ab. Bitte fügen Sie Ihrer Abgabe eine Datei studiengang.txt hinzu, welche Ihren Studiengang enthält. Dies dient der einfacheren Korrektur.