

Übungszettel 2 (10 Punkte)

Abgabe: 3.11. 14h als weblink

Pattern Teil 2:

In dieser Aufgabe geht es wieder um das Erzeugen von 2-dimensionalen Mustern. Diesmal sollen systematisch mathematische Funktionen angewendet und untersucht werden, sowie die in der Vorlesung besprochenen Modulationen von Parametern (Amplitude, Frequenz, Phase) und Transformationen (Quantisierung, Blending, Clamping).

Es sollen wieder 3 verschiedene Pattern erzeugt werden. Legt zur Vereinfachung für jedes pattern eine eigene Klasse an, die festgelegte und berechnete Parameter als Felder enthält. Jede Pattern-Klasse soll zudem folgende Methoden implementieren: `update(time)`, `draw(N)`, `incremental_draw(x,y)`.

Dazu ein einfaches Beispiel:

```
var Squares = function() {
    this.size = 0;
}

Squares.prototype.update = function (time) {
    this.size = Math.round(time);
}

Squares.prototype.draw = function (N) {
    for (var i=0; i<N; i++) {
        for (var j=0; j<N; j++) {
            this.incremental_draw (i*canvas.width/N,
                                   j*canvas.height/N);
        }
    }
}

Squares.prototype.incremental_draw = function (x,y) {
    context.strokeRect(x,y,this.size,this.size);
}
```

Die Aufgaben:

a) Implementiert die Klassen nach dem oben genannten Muster. In der Funktionen `incremental_draw` werden die einzelnen Formen, aus denen das Pattern zusammengesetzt ist in Ortsabhängigkeit gezeichnet, zB Rechtecke, deren Parameter wie Größe, Farbe, Seitenverhältnis von der Position abhängig sind. Andere Beispiele sind Linien, mit ortsabhängiger Länge, Wikel usw. Auch denkbar sind Polygone mit unterschiedlicher Anzahl an Ecken. Es ist durchaus denkbar, dass pro Aufruf mehrere Formen (zB 10 Punkte in bestimmter Verteilung) gezeichnet werden. Ebenso kann an dieser Stelle ein Zeitverhalten implementiert werden, wie zB eine Rotation der einzelnen Formen (die wiederum Ortsabhängig sein kann, zB drehen sich Formen in der Mitte des Bildes schneller als außen). In dieser Aufgabe ist ein gewisses Finetuning und Experimentieren mit den Parametern, wie der Frequenz von periodischen Funktionen, notwendig um schöne Ergebnisse zu erzielen.

b) Nutzt die `update(time)` Funktion um langsamere Änderungen des Patterns zu erzeugen. Wir können hier Parameter global für das ganze Pattern berechnen, die dann in den einzelnen Aufrufen von `incremental_draw` verwendet werden.

c) Erstellt eine kleine Animation, in der die einzelnen Pattern ineinander überblendet werden. Experimentiert mit verschiedenen Möglichkeiten der Zeit- und Ortsabhängigkeit und findet ein Verhalten, welches für Euch ästhetisch ansprechend ist und welches mit den Pattern harmoniert. Es gibt zahllose Möglichkeiten: der Übergang kann stellenweise abrupt sein, wie Fliesen, die nach und nach umgedreht werden. Er kann sich in einem Zeitraum von mehreren Sekunden vom linken zum Rechten Bildrand bewegen, wie eine Schiebeblende, der Übergang kann an vielen zufälligen Stellen im Bild starten und sich ausbreiten wie Feuer, eine Teilform kann sich in eine andere morphen (ein Pattern aus Rechtecken wird zu vielen Punkten), die einzelnen Teilformen können, wie Scherben, herunterfallen und die neuen kommen von oben ins Bild hinein...