

Übungsblatt 5

Abgabe bis Dienstag, den **5. Juni 2018** um **12:00 Uhr**

Bei diesem Übungsblatt geht es darum, den Code vom letzten Übungsblatt objekt-orientiert zu machen. Es muss dafür sehr wenig neuer Code geschrieben, sondern hauptsächlich umstrukturiert werden.

Aufgabe 1 (10 Punkte)

Deklarieren Sie eine Klasse *GameOfLife* in einer Datei *GameOfLife.h*.

Die Klasse sollte Variablen für den Inhalt der beiden Felder enthalten, für die beiden Zeiger auf diese Felder, für die Größe des Displays, die Koordinaten der zuletzt geklickten Position, den aktuellen Spielmodus (schrittweise oder automatisch), die aktuelle Anzahl der lebendigen Zellen und was immer Ihr Programm noch braucht. Insbesondere sollten die Variablen dabei sein, die für das Ü4 noch global waren (sofern Sie sie jetzt noch brauchen). Für dieses Übungsblatt sollte es keine globalen Variablen mehr geben.

Die Klasse sollte außerdem Methoden *initialize*, *showState*, *updateState*, *aliveNeighbors*, *processUserInput* und *play* haben. Bis auf die letzte Methode sind das genau die Funktionen vom Ü4. Die Methode *play* ist für den Teil, der für das Ü4 noch in der main Funktion stand, siehe dazu auch Aufgabe 4. Die Methoden, die den Inhalt des Bildschirms modifizieren (*initialize*, *showState*, *play*) brauchen Sie nicht zu testen, die anderen wie gehabt.

Schreiben Sie einen Konstruktor, in dem die Klassenvariablen initialisiert werden (darüber hinaus sollte dort kein Code stehen). Schreiben Sie auch einen Destruktor, der geeignet „aufräumt“. Überlegen Sie selber, welcher Teil des Codes vom Ü4 dafür jeweils in Frage kommt.

Aufgabe 2 (5 Punkte)

Implementieren Sie die Methoden (inklusive Konstruktor und Destruktor) Ihrer Klasse aus Aufgabe 1 in einer Datei *GameOfLife.cpp*. Verstehen Sie, dass der Code praktisch derselbe ist wie der für das Ü4. Sie müssen ihn nur geeignet an das neue Klassendesign anpassen. Wenn Ihr eigener Code nicht funktioniert hat, können Sie sich auch an der Musterlösung für das Ü4 bedienen.

Aufgabe 3

Implementieren sie optional eine Klasse *GameOfLifeFigures* (in einer zugehörigen *.h* und *.cpp* Datei) für den Glider und die Gliderfabrik vom Ü4. Überlegen Sie selber, wie man die Klasse am besten designed, damit der aufrufende Code möglichst einfach die betreffende Figur übernehmen kann. Es gibt auch noch unzählige andere interessante Figuren im Game Of Life, zum Beispiel das *Acorn* oder *Kok's galaxy*. Lassen Sie Ihrer Implementierungslust freien Lauf.

Aufgabe 4 (5 Punkte)

Schreiben Sie ein Hauptprogramm in einer Datei *GameOfLifeMain.cpp*, so dass zusammen mit Ihrem Code aus den vorherigen Aufgaben alles wie nach den Anforderungen für das Ü4 läuft. In dem Hauptprogramm sollte nur minimaler Code stehen: Deklaration/Erzeugung des Objektes, Initialisierung des Bildschirms, Starten des Spiels.

Laden Sie wie gehabt alle Code-Dateien und das Makefile in unser SVN hoch, in einem neuen Unterverzeichnis *blatt-05*. Es gelten weiterhin die 10 Gebote von der letzten Seite des Ü1 und nehmen Sie die Ratschläge Ihres Tutors / Ihrer Tutorin ernst.

Laden Sie wie gehabt auch eine Datei *erfahrungen.txt* hoch (im Unterordner *blatt-05*), in der sie kurz Ihre Erfahrungen mit dem Ü5 und der Vorlesung dazu beschreiben.

Sie sind eine komplexe Figur in einem zellulären Automaten. Was bedeutet das für Ihre Lebenseinstellung?