**Matura-Arbeit**

**Einleitung**

Ein kleiner oder besser gesagt ein jüngerer Bruder ist in vielen Hinsichten etwas Tolles. Was ich aber nicht erwartet hätte, ist, dass er mir indirekt hilft ein Thema für die Matur-Arbeit zu finden. In der Quarta erhielt er von meinem Onkel einen Rubik’s Cube zu Weihnachten geschenkt, der ihn nicht sonderlich interessierte. Erst als ich ihn versuchte zu lösen, begann das wetteifern, denn er war der festen Überzeugung, dass man das nicht lösen kann.

Ich muss zugeben, meine ersten Versuche scheiterten kläglich. Als ich mich etwas von meinem Frust erholt hatte, wollte ich unbedingt dieses Rätselspiel „besiegen“. Ich tat, was ich immer tue, wenn ich ein Problem habe: Ich fragte das Internet um Hilfe. Da fand ich auch die ersten Lösungsansätze. An diesem Tag schaffte ich es das erste Mal den Würfel zu lösen. Ich benötigte neben den mehreren ausgedruckten A4-Seiten eine Unmenge an Zeit. Es müssen gegen zwanzig Minuten gewesen sein. Durch das mehrfache Wiederholen der gleichen Algorithmen habe ich sie immer besser verinnerlicht bis ich sie schliesslich auswendig konnte. Ohne die Ausdrucke habe ich bereits eine Menge Zeit gespart. Die benötigte Zeit konnte ich etwa um die Hälfte der Anfangszeit reduziert.

Als ich den Rubik’s Cube zum ersten Mal mit in die Schule nahm, war die Begeisterung gering. Als sich jedoch die ersten selbst am Rätselspiel ausprobierten, stieg das Interesse. Binnen weniger Wochen lernte die halbe Klasse die Algorithmen auswendig und das Phänomen der „Würfel-Krankheit“ übertrug sich auf die Parallelklassen. Zumindest die der MN-Abteilung.

**Geschichte des Rubik’s Cube**

Der Würfel wurde von Erno Rubik erfunden. Es sollte Studenten Verständnis für die dreidimensionale

**Vorgehen**

1. Da in meiner Matura-Arbeit der Rubik’s Cube eine zentrale Rolle spielt habe ich mich zu Beginn mit dem Aufbau dieses Würfels beschäftigt. Meine erste Erkenntnis war, dass der Würfel aus unterschiedlichen Steinen besteht. Es gibt insgesamt drei Steintypen. Dabei hat jeder Typ andere Eigenschaften.

Der Mittelstein befindet sich in der Mitte einer Fläche. Er hat nur eine farbige Oberfläche. Die sechs Mittelsteine sind miteinander verbunden und halten den Würfel zusammen. Daher können die Mittelsteine beim Drehen der Flächen in ihrer Position nicht verändert werden. Beim meinem Würfel heisst das konkret, dass der weisse Mittelstein immer gegenüber des gelben, der rote gegenüber des orangen, der blaue gegenüber des grünen Mittelsteines stehen wird, egal wie oft der Würfel gedreht wird. Insgesamt gibt es sechs Mittelsteine.

[Foto des offenen Cubes]

Der Kantenstein ist bereits etwas spannender. Er befindet sich zwischen zwei Mittelsteinen und hat zwei farbige Oberflächen. Dieser Steintyp ist wesentlich freier, da er seine Position im Würfel verändern kann. Er kann mit jedem der elf anderen Kantenstein vertauscht werden.

Der Eckstein liegt, wie der Name schon sagt, in den Ecken des Würfels. Er hat drei farbige Oberflächen. Die Ecksteine können ihre Position untereinander vertauschen. Insgesamt gibt es acht Ecksteine.

Für mein Programm habe ich mir gedacht, dass ich aus dieser Erkenntnis eine Klasse „Stein“ bilde, von der ich danach drei Klassen ableite. Diese Klassen sollen die oben genannten Steintypen darstellen. Den Rubik’s Cube würde ich aus diesen drei Klassen zusammensetzen, indem ich mehrere Objekte erstelle.

Für die Vaterklasse benötigte ich zur Erstellung des Würfels noch zwei Methoden. Eine Methode sollte die Stein-Objekte an ihre richtige Position verschieben und die zweite sollte die Stein-Objekte an einer bestimmten Position ausrichten.

1. Der zweite wichtiger Punkt war die Herausforderung der dritten Dimension. Bei der Zielfindung habe ich mich lange schwer getan, weil ich nicht wusste, wie ich dieses Problem angehen sollte. Ich hatte ein Buch, welche das Problem durch Matrizen löst. Da mir dieses Thema den Rahmen meiner Arbeit zu sprengen drohte, habe ich den Tipp meiner Betreuungsperson beachtet und mich dafür entschieden mit der Java 3D-Bibliotheke zu arbeiten.  
     
   Die Benutzung der Bibliothek benötigte einen grösseren Zeitaufwand, weil es sich um die erste Bibliothek handelt, die ich in grossen Umfang benutzte. Im Internet habe ich eine umfassende Dokumentation über die Klasse gefunden, die ich als Selbststudium las. Dabei habe ich folgendes System entwickelt: Nach den Kapiteln habe ich versucht das gelesenen in Beispielprogrammen umzusetzen. Zu Beginn habe ich Beispiele im Internet gesucht, direkt abgeschrieben und versucht zu interpretieren, was die einzelnen Zeilen genau aussagen. In einem späteren Zeitpunkt habe ich die Beispielprogramme meinem Projekt angepasst. Mit diesem Vorgehen habe ich mich langsam an mein tatsächliches Ziel angenähert.  
     
   Während dem Selbststudium habe ich gemerkt, dass es sich nicht auszahlt die drei Steintypen zu vererben. Weshalb ich alle Steine aus einer Klasse bildete.

Aufbau

Mein Programm besteht aus fünf Klassen. Die Klasse Stone wird nur am Anfang aufgerufen und zeichnet die

Die Java 3D Bibliothek

Um mein Projekt zu verwirklichen, habe ich die Java 3D Bibliothek verwendet. Es handelt sich dabei um „Application Programming Interface“ (API), welche 3D-Modelle generieren und verändern kann.

Es ist als Hierarchie aufgebaut und sieht aus, wie ein Familienstammbaum. Eine untere Hirarchiestuffe wird als Kind-Objekt der oberen bezeichnet. Die obere heisst Eltern-Objekt.

Ein Node [„Knoten“] ist ein Punkt in diesem Modell. Man unterscheidet zwischen Leaf Node [„offener Knoten“] und dem Group Node. Der leaf Node ist ein Punkt im Stammbaum ohne ein Kind-Objekt. Das heisst, dass er sich unterhalb nicht weiter fortsetzt. Der Group Node degegen besitzt mindestens ein Kind-Objekt.

Wichtig bei diesem Modell ist, dass ein Eltern-Objekt mehrere Kind-Objekte haben kann, jdeoch ein Kind-Objekt nur ein Eltern-Objekt.

Es handelt sich um ein Baum-Modell. Wobei immer ein Kind-Objekt immer nur ein Kind-Objekt beinhalten kann.

Objekte der Bibliothek

Branch Group

Transform Group

Shape3D

Eigene Objekte