

# Inhaltsverzeichnis

|          |   |          |
|----------|---|----------|
| <b>1</b> | <b>Einleitung</b>                               | <b>1</b> |
| 1.1      | Spielziel und Spielfunktionen . . . . .         | 1        |
| 1.2      | Effekte im Spiel . . . . .                      | 1        |
| <b>2</b> | <b>Netzwerkmodus</b>                            | <b>2</b> |
| 2.1      | Limitierungen des Netzwerkmodus . . . . .       | 2        |
| <b>3</b> | <b>MarioMod</b>                                 | <b>2</b> |
| 3.1      | Schwierigkeitsgrad . . . . .                    | 2        |
| <b>4</b> | <b>Objektorientierte Architektur des Spiels</b> | <b>3</b> |
| 4.1      | Klassenaufteilung . . . . .                     | 3        |
| <b>5</b> | <b>Bekannte Fehler</b>                          | <b>4</b> |

## Timtris Projektbeschreibung

### 1 Einleitung

Das Spiel Timtris ist eine Java Implementierung der beiden Spiele Tetris und Dr. Mario welche auch über das Netzwerk spielbar ist. Es ist aus einem Universitätsprojekt entstanden.

Name der Klasse mit der Main-Methode: Board.class

#### 1.1 Spielziel und Spielfunktionen

Das Spiel basiert auf dem bekannten Spiel Tetris. 7 verschiedene Steine sollen auf dem Spielfeld so angeordnet werden, dass sich volle horizontale Reihen ergeben. Diese werden nach Vervollständigung gelöscht und Punkte werden gutgeschrieben. (1 Punkt pro gelöschtem GeometricObject, bei 2 oder mehr Reihen +4Punkte pro zusätzlich gelöschter Reihe) Verloren ist das Spiel sobald keine Figuren mehr auf das Spielfeld passen. Im Singleplayermodus ist das Spiel theoretisch endlos, wird jedoch von Level zu Level schneller und durch Effekte erschwert.

#### 1.2 Effekte im Spiel

Die Effekte sind im einzelnen:

- Verschobene Startposition des herunterfallenden Objektes
- Rotation des Objektes beim Erstellen
- Das sofortige Herunterfallen des gerade erstellten Objektes an der Erstellungsposition

- Löschen eines zufälligen GeometricObjects in der Liste
- Hinzufügen einer neuen Zeile mit einer Lücke von unten
- Das Herunterfallen einer Figur mit 2 horizontalen GeometricObjects (siehe MarioMod)

Das Auftreten der Effekte ist von den Schwierigkeitseinstellungen unter Options abhängig. Der Defaultwert bei Start des Spiels ist Medium.

## 2 Netzwerkmodus

Es existiert ein Netzwerkspielmodus mit den Möglichkeiten das Spiel über Netzwerk mit einem Mitspieler ähnlich wie im Singleplayermodus zu spielen (theoretisch unendliches Spiel bis ein Spieler keinen Stein mehr setzen kann). Möglich ist auSSerdem ein Spiel auf Punkte in dem der Server vorher eine Punktehöchstgrenze festsetzt. Wird diese von einem der Spieler erreicht steht er als Gewinner fest. Zudem kann das Netzwerkspiel auf Zeit gespielt werden. Hierbei erfolgt die Ermittlung des Gewinners nach einer bestimmten Zeitspanne anhand der erreichten Punkte.

### 2.1 Limitierungen des Netzwerkmodus

Der Netzwerkmodus ist nur zwischen 2 Mitspielern möglich. Wird im Netzwerkmodus von einem Spieler eine Reihe gelöscht, so erscheint eine neue fast vollständige Reihe auf dem Spielfeld des Gegners. Theoretisch wäre das Spiel auf beliebig viele Clienten erweiterbar gewesen, jedoch erschien das aufgrund der zu erwartenden Spielreihen die zu den Clienten hinzugefügt werden ohne ein neues Konzept der Verteilung der Reihen auf die Mitspieler unvorteilhaft, da es zu viele Reihen auf einmal pro Spieler bedeutet hätte.

## 3 MarioMod

Ein weiterer eingebauter Mod ist an das Spiel Dr. Mario angelehnt. Hier gilt es die auf dem Spielfeld verteilten Kreise durch vertikale Anordnung von mindestens 4 Spielsteinen gleicher Farbe (inklusive einem Kreis) zu löschen. Erschwert wird dies dadurch dass, anders wie bei dem klassischen Dr. Mario Spiel evtl. übrig bleibende Steine bei Löschung einer 4-er Reihe nicht herunterfallen und somit in der Luft schweben bleiben. Werden die Steine so gesetzt das 2 Reihen gelöscht werden müssten, wird nur eine Reihe gelöscht um das Spiel noch zusätzlich zu erschweren.

### 3.1 Schwierigkeitsgrad

Der Schwierigkeitsgrad erhöht sich in dieser Modifikation durch die Geschwindigkeit der fallenden Steine, die sich nach einer bestimmten Anzahl generierter Blöcke erhöht. (abhängig von der gewählten Schwierigkeit im Optionenmenü) Sind alle Kreise auf dem Spielfeld gelöscht werden setzt sich die Geschwindigkeit wieder zurück und ein neues Spielfeld mit mehr zu vernichtenden Kreisen wird generiert (das nächste Level). Verloren ist das Spiel auch hier wenn der nächste Stein nicht mehr auf das Spielfeld passt.

## 4 Objektorientierte Architektur des Spiels

Die objektorientierte Architektur des Spieles begründet sich zum einen in den schon im Semester gefertigten GeometricObjects, die weiterverwendet und zu z.B. den Tetrisfiguren zusammengesetzt werden. Diese basieren auf der Klasse Vertex, die u.a. zur Bewegung der GeometricObjects verwendet wird. Auch die Unterklasse der GeometricObjects Circle wurde im Mod des Spieles implementiert. Weiterhin wurde die im Praktikum eingeführten Schnittstellen MoveableObject und Paintable sowie ein KeyListener für die Steuerung implementiert. Viele weitere Objekte wurden zudem aus den Standard java Libraries importiert (z.B: Socket uva.).

### 4.1 Klassenaufteilung

Auch eine sinnvolle Aufteilung der Funktionalitäten des Programmes nach Klassen ist erfolgt.

- So wird die Klasse Board für das Spielfeld und seine Funktionen genutzt (z.b. das Zeichnen der Objekte, den Aufbau des Pausescreeens u.a.).
- Die Deleteklasse kümmert sich um die Frage ob GeometricObjects gelöscht werden sollen und wenn es so ist übernimmt sie auch das Löschen der Objekte.
- Die Gameklasse startet und beendet die in Timtris implementierten Spiele und schreibt den GameLog mit den erreichten Punkteständen der jeweiligen Spiele in das Verzeichnis des Spieles.
- Die Klasse KeyMove beschäftigt sich mit den Links-Rechts-Bewegungen der Tetrisfiguren und eventuellen Crashes die durch die Bewegungen verursacht werden. AuSSerdem enthält sie die Funktion down mit der die Figuren soweit wie möglich nach unten geschoben werden können.
- Die Klasse Makeo erzeugt die für das Tetrisspiel nötigen Figuren und je nach gewähltem Schwierigkeitsgrad etwaige Zusatzeffekte.
- In der MarioModklasse stehen die angepassten Funktionen für den MarioMod (z.B. zum generieren der Objekte, zum Löschen usw.)
- Die Klasse Menu beinhaltet einzig den Aufbau des Hauptmenüs (der MenuBar) und 2 Frames (für about, Spielanleitung und Optionen) des Spieles und gibt die aufgebaute Menüleiste zum einbinden an die Klasse Board zurück.
- In der Networkklasse werden sowohl die Funktionen zum Starten des Clients als auch zum Starten des Servers und zum Erstellen der verschiedenen Frames und zum schlieSSen der Netzwerkverbindungen gespeichert.
- Die Klasse Rotate enthält Funktionen um die Tetrisfiguren rotieren zu lassen, angepasst auf jede Tetrisfigur.
- Eine Unterklasse von JLabel stellt die StatusBar Klasse dar, die dazu dient die Statusanzeige bez. Level und Punkten sowie im Netzwerkmodus die Punkte des Gegners, die verbleibende Zeit und die Punktegrenze anzuzeigen.

Alle Funktionen der Klassen sind nicht statisch und können somit jederzeit ersetzt werden.

## **5 Bekannte Fehler**

Leider war es mir nicht möglich folgenden Fehler zu korrigieren: Er tritt im Netzwerkmodus auf wenn 2mal hintereinander der gleiche Spieler den Server stellt und verhindert die Ausführung des Timers auf der Clientseite. Wird der Server jedoch abwechselnd gestellt, so funktionieren beide Seiten.