

## Übung 09: Swing, MVC-Prinzip

Abgabetermin: 5. 6. 2007, 8:15

Name: \_\_\_\_\_

Matrikelnummer: \_\_\_\_\_

Informatik:  G1 (Prähofer)    G2 (Prähofer)    G3 (Wimmer)    G4 (Wimmer)

WIN:         G1 (Ibrahim)    G2 (Ibrahim)    G3 (Schwinger)

Aufgabe	Punkte	abzugeben schriftlich	abzugeben elektronisch	korr.	Pkte
Übung 9	24	Prosabeschreibung, Java-Programm, Screenshots	Java-Programm	<input type="checkbox"/>	

### Graphische Oberfläche für Spiel

In Übung 6 haben wir das Logikrätsel „Sudoku“ mit einer textuellen Schnittstelle implementiert. In dieser Übung soll diese durch eine graphische Swing-Oberfläche ersetzt werden. Die Kommunikation zwischen Oberfläche und Datenmodell soll strikt dem MVC-Prinzip folgen.

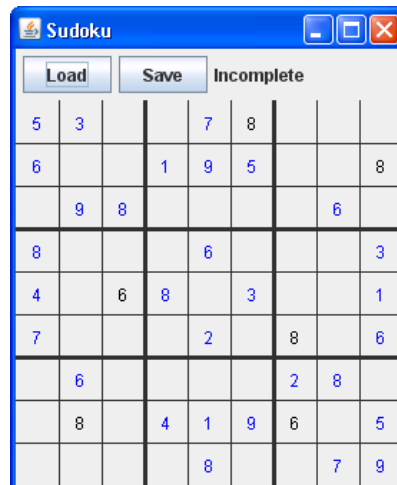
Falls Sie nicht auf Ihrer Lösung von Übung 6 aufbauen können oder wollen, können Sie stattdessen eine graphische Oberfläche für TicTacToe (Quellcode auf der Homepage von Übungsstunde 6) entwickeln. Alle Teile der Aufgabenstellung gelten dann sinngemäß, insbesondere die Einhaltung des MVC-Prinzips.

Das Datenmodell von Übung 6 war noch nicht nach dem MVC-Prinzip realisiert. Ändern Sie das Modell, sodass Listener beim Datenmodell registriert werden können und diese benachrichtigt werden, sobald sich etwas am Modell ändert, d.h. wenn sich ein Zellenwert ändert oder wenn sich der Spielstatus ändert.

Die graphische Darstellung erfolgt in einer selbst gezeichneten Komponente. Der Wert einer Zelle kann geändert werden, indem der Benutzer in die Zelle klickt und dann den neuen Wert in einem Dialog eingibt. Verwenden Sie dazu einen `MouseListener` und die Methode `OptionPane.showInputDialog`.

Der aktuelle Spielzustand kann in eine Datei gespeichert und daraus wieder geladen werden. Implementieren Sie den Dateizugriff über entsprechende `Reader` und `Writer`. Als Vereinfachung müssen Sie in der Datei nicht zwischen fixen und änderbaren Zellen unterscheiden. Alle eingelesenen Zellen sind automatisch fix. Es müssen daher nur 81 Ziffern (= Zeichen) gelesen werden.

Die Anwendung kann beispielsweise so aussehen:



Hinweise:

- Halten Sie sich strikt an das MVC-Muster, d.h. alle Aktualisierungen der Views werden vom Model angestoßen. Registrieren Sie dazu die nötigen Listener beim Datenmodell.
- Die graphische Darstellung und das Layout des Fensters sind nicht fix vorgegeben und müssen daher nicht genau dem obigen Screenshot entsprechen. Sie können auch ein Menü verwenden.