

Übungsblatt 9

Abgabe: 15.01.2023

Aufgabe 1 Lässig Level laden (100 %)

Die Felddescription soll nicht mehr direkt aus einem *String*-Array im Code stammen, sondern aus einer Datei geladen werden. Die Datei beschreibt auch nicht nur das Bodengitter, sondern auch die Positionen und Orientierungen der Akteure. Dazu wird in der Felddescription der Platz genutzt, der bisher die Gitterknoten ('O') dargestellt hat. Für jeden Akteur werden vier mögliche Symbole benötigt, da er ja in vier verschiedenen Ausrichtungen platziert werden kann. Auch wird erwartet, dass in jeder Level-Beschreibung genau eine (Tastatur-gesteuerte) Spielfigur, also eine Instanz der Klasse *Player*, vorkommen muss. Ihr müsst festlegen, durch welche Symbole eure Spielobjekte in der Level-Beschreibung kodiert werden sollen.

Der Konstruktor der Klasse *Level* soll die Datei mit dem übergebenen Namen einlesen.¹ Deren Inhalt entspricht einfach dem bisherigen *String*-Array, aber mit weiteren Symbolen für die Akteure. Die Einträge des *String*-Arrays sind nun einfach die Zeilen der Textdatei. Aus der eingelesenen Datei wird zum einen wieder ein *String*-Array erzeugt, aus dem dann ein Objekt der Klasse *Field* konstruiert wird, das Teil des Levels ist. Zum anderen werden Objekte für alle Akteure erzeugt, die zusätzlich in einer Liste abgelegt werden, so dass sie später per *getActors()*-Methode abgefragt werden können. Wichtig ist dabei, dass der *Player*, wie schon auf Übungsblatt 8, immer am Anfang der Liste steht.

Alle beim Einlesen eines Levels auftretenden Fehler werden dem Aufrufer durch Erzeugen einer *IllegalArgumentException* mitgeteilt.² Deren Nachricht beschreibt den Grund des Fehlers. Diese Ausnahme braucht nicht vom Hauptprogramm gefangen zu werden. Folgende Fehler sollen zu einer solchen Exception führen:

- Die Level-Datei wurde nicht gefunden.
- Es gab Probleme beim Lesen der Datei.
- Die Datei enthält ungültige Symbole an den Positionen von Gitterknoten.³
- Die Datei enthält nicht genau einen *Player*.

Stellt euer Hauptprogramm auf die Benutzung der Klasse *Level* um.

Aufgabe 2 Bonusaufgabe: Ich bin dann mal weg (10 %)

Implementiert in der Klasse *Level* und hilfsweise in der Klasse *Field* eine Methode *hide()*, die einen Level wieder vom Bildschirm verschwinden lässt. Dazu müsst ihr euch alle Instanzen der Klasse *GameObject* merken, die beim Erzeugen eines Levels erstellt wurden und bei ihnen in der Methode *hide()* *setVisible(false)* aufrufen.

Abgabe. Analog zu Übungsblatt 6, d.h. im selben Repository in einem Ordner *loesung09*.

¹Statt *new FileInputStream(fileName)* könnt ihr auch *Game.Jar.getInputStream(fileName)* aufrufen, das sich ähnlich verhält, aber auch in *jar*-Archiven funktioniert.

²Letztendlich ist ja immer der übergebene Dateiname daran schuld, wenn etwas nicht funktioniert.

³Die Zeichen für die Nachbarschaftsstruktur müssen nicht geprüft werden.