Institut für Informatik
Lehr- und Forschungseinheit für Theoretische Informatik
Dr. Thorsten Altenkirch, Dr. Ralf Matthes
Thomas Anderl, Alessandro Macri

Programmierpraktikum
WS 1999/00

---

2. Milestone (Siedler von Catan)

24. 11. 1999

Deadline: 8.12.1999

Unser nächstes Ziel ist ein *Datenmodell* für die interne Darstellung der Insel während des Spiels zu entwickeln. Dieses Modell soll alle für das Spiel relevanten Attribute der Insel Catan enthalten bzw. ableiten können. Die Darstellung der Spieler mit Handkarten und Rohstoffen wird z.Z. noch nicht verlangt.

Es soll eine Klasse `Catan` definiert werden, wobei der Konstruktor `Catan()` eine neue Insel gemäß der Spielregel erzeugt. Außerdem sollen u.a. folgende Klassen eingeführt werden:

`Feld` Darstellung eines Feldes u.a. mit folgenden Attributen:

- `typ` Handelt es sich um ein Getreide, Wüste, ...
- `wuerfel` Wann produziert die Landschaft ?
- `Kreuzung [] kreuzungen` Welche Kreuzungen sind diesem Feld benachbart ?

`Kreuzung` Darstellung einer Kreuzung mit den Attributen:

- `Weg [] wege` Verbindungswege, die bei dieser Kreuzung enden.
- `Feld [] felder` Welche Felder sind dieser Krezung benachbart?
- `Siedlung siedlung` Gesetzt, falls diese Kreuzung bebaut ist.

`Weg` Darstellung einer Kante, mit dem Attribut

- `Kreuzung [] kreuzungen` Welche Kreuzungen verbindet dieser Weg?
- `Strasse strasse` Gesetzt falls die Kante mit einer Straße bebaut ist.

`Siedlung` Siedlung oder Stadt mit Attributen:

- `farbe` Besitzer der Siedlung.
- `boolean stadt` Handelt es sich um eine Stadt?

`Strasse` Strasse mit dem Attribut:

- `farbe` Besitzer der Strasse.

Bei einigen Attributen kann der Typ frei gewählt werden. So kann z.B. das Attribut `farbe` durch ein `int` oder durch einen `String` oder durch eine neue Klasse dargestellt werden.

Für diese Klassen sollen gemäß der Konventionen für Java Beans `get`- bzw. `set`-Methoden implementiert werden. `set`-Methoden sind nur dann notwendig, wenn das entsprechende Attribut im Laufe des Spiels verändert wird.

`Catan` selbst soll mindestens über folgende Attribute verfügen:

`boolean eröffnung` Insel ist in der Eröffnungsphase, in der andere Regeln gelten.

`Feld mittelFeld` Gibt das Feld in der Mitte der Insel zurück.

Die Methoden sollen, wo sinnvoll, auch prüfen, ob die entsprechende Modifikation des Zustandes mit den Spielregeln vereinbar ist. Falls nicht soll eine selbstdefinierte `exception` erzeugt werden, z.B.

Abstandsregel Beim versuchten Bau einer Siedlung wurde die Abstandsregel verletzt.

KeineVerbindung Es wurde versucht ein Bauwerk ohne Verbindung zu bestehenden zu bauen. Die soll natürlich in der Eröffnunsphase keine Rolle spielen.

Zum Testen soll die Klasse `Catan` eine Methode

```
String toString()
```

implementieren, die eine übersichtliche, semigraphische ASCII-Darstellung des Inselzustandes erzeugt.

Außerdem sollen folgende Auswertungsmethoden realisiert werden:

```
void ertrag(int wurf)
```

Gibt die Erträge aus, die durch diesen Wurf produziert wurden. Später sollen natürlich die entsprechenden Spielerobjekte aufdatiert werden.

```
int punkte(farbe)
```

berechnet die durch Siedlungen und Städten erzielten Punkte für einen bestimmten Spieler.

```
int laengsterWeg(farbe)
```

berechnet den längsten Weg für einen bestimmten Spieler.

Nach Rücksprache mit dem *Coach* kann in begründeten Fällen von der Spezifikation abgewichen werden. Sinnvolle Erweiterungen sind natürlich immer willkommen.