

Programmieren eines Frogger-Games in Greenfoot



Spielmechanik:

- Ein Objekt Frosch soll eine Straße von unten nach oben überqueren.
- Auf der Straße fahren Objekte (Auto, Lastwagen ...) von links nach rechts und/oder von rechts nach links. Wenn ein solches Objekt an die Weltgrenze stößt, kehrt es entweder um oder taucht auf der anderen Seite wieder auf.
- Hat der Frosch den oberen Rand erreicht, erhält er einen Punkt. Er wird wieder auf seine Ausgangsposition gesetzt.
- Hat der Frosch 10 Punkte, hat er gewonnen. Ausgabe einer entsprechenden Nachricht + Spielende (**Greenfoot.stop()**).
- Kollidiert der Frosch mit einem feindlichen Objekt, wird er an seine Ausgangsposition gesetzt und verliert ein Leben.
- Hat der Frosch keine Leben mehr, endet das Spiel. Ausgabe einer entsprechenden Nachricht.
- Punktestand, Leben werden permanent mit **showText(...)** angezeigt.

Powerups

- Ab und zu tauchen Powerups auf. Verwenden Sie dazu in der Weltklasse ein int-Attribut **timer** in der **act()**-Methode:

```
timer = timer + 1
if(timer == 1000)
    this.addObject(.....)
    timer = 0
```

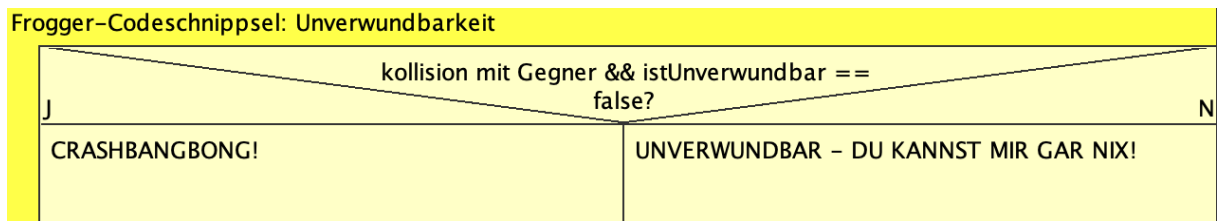
oder Zufallsereignisse:

```
if(Greenfoot.getRandomNumber(100) > 90)
    this.addObject(new Powerup(), zufällige-x-Pos, zufällige-y-Pos)
```

Alternativ können Sie auch, falls Sie Greenfoot in Version > 3.8 installiert haben, die **sleepFor(...)**-Methode verwenden, siehe unten.
- Die Powerups stehen entweder still, oder bewegen sich (sie könnten z.B. mit den Gegnern mitfahren oder von oben nach unten fallen). Dazu gibt es ein wunderbares Video, auf das ich Sie hinweise (oder informatikZentrale -> Greenfoot3 -> Tutorials -> Tricks -> Timer)
- Wenn der Frosch ein solches Powerup berührt, verschwindet das Powerup und der Frosch profitiert.

- Es gibt zwingend! folgende Powerups (Sie können das erweitern):
 1. Leben - Erhöht die Anzahl der Leben des Frosches.
 2. Punkte - Erhöht die Anzahl der Punkte, die der Frosch hat.
 3. Unverwundbarkeit - Macht den Frosch für eine bestimmte Zeit unverwundbar.
 4. Geschwindigkeit - Erhöht die Geschwindigkeit des Frosches für eine bestimmte Zeit

Für 3, 4 brauchen Sie eine boolean-Variablen + Timer (**istUnverwundbar** wird auf **true** gesetzt, **timer** beginnt runterzuzählen, sobald **timer == 0**, wird **istUnverwundbar** auf **false** gesetzt; bei Kollision mit einem Gegner gehen wir so vor:



Getter, Setter

Falls Sie schon bei dieser Einheit waren: Verwenden Sie konsequent IMMER Getter und Setter für den Zugriff auf **private** Variablen (außer im Konstruktor)!

Methoden

- Verwenden Sie konsequent Methoden (z.B. **kollisionPruefen()**, **spielBeenden()**, **bewegen()**, **punkteHochzaehlen()** etc.)

Ersatz für Timer: sleepFor(...)-Methode

Seit der Greenfoot 3.8.1 gibt es eine sehr nützliche Methode, die heißt **sleepFor(int actZyklen)**. Damit werden viele Dinge einfacher, für die man bisher einen Timer gebraucht hat.

Angenommen, der Frosch soll bei Kollision mit einem Lastwagen für kurze Zeit ein anderes Bild bekommen, dann könnte man das so machen:

```

if(this.isTouching(Lastwagen.class)) {
    this.setImage("splatter.png");
    this.sleepFor(20);
    this.setImage("originalbild.png");
}
  
```

Profi: Color-Check

Vielleicht wollen Sie prüfen, ob der Frosch in einer der grauen Einfahrten oben angekommen ist. Schauen Sie sich dazu die World-Methode **getColorAt(...)** an.