

Ergänzungen

Bälle erzeugen

Du hast dich sicherlich geärgert, dass jedes Mal, wenn dein Ball verschwindet, du umständlich einen neuen Ball erzeugen musst. Wie wäre es, wenn bei Betätigen z.B. der Leertaste ein neuer Ball erzeugt würde?

- In welcher Klasse wird ein neuer Ball erzeugt, welche Methode muss man benutzen?
- Erstelle in der Klasse Spielfeld eine neue Methode `act()`.

Der Quellcode lautet:

```
public void act() {
    if (Greenfoot.isKeyDown("space")) {
        Ball ball = new Ball();
        addObject(ball, 320, 240);
    }
}
```

Kannst du erklären, was hier passiert? Benutze die Klassendokumentation von Greenfoot.

Leider entstehen jetzt immer viele Bälle, besonders wenn du die Leertaste lange gedrückt hältst - es sollte nur dann ein Ball erzeugt werden, wenn es auf dem Spielfeld keinen weiteren gibt.

Die World-Methode `getObjects(java.lang.Class cls)` liefert eine Liste aller vorhandenen Objekte einer bestimmten Klasse. Du musst nur prüfen, ob diese Liste noch keine Einträge hat. Die Anzahl der Listenelemente liefert die Methode `size()`. Ergänze dazu die Bedingung in `act()`:

```
if (Greenfoot.isKeyDown("space") &&
    getObjects(Ball.class).size() == 0) {
    ...
}
```

Übersetze das Programm und überprüfe deine Änderungen.

Spielfeld füllen

Auch das Füllen des Spielfeldes ist mühsam. Diese Aufgabe könnte doch die Klasse Spielfeld selbst erledigen?

Erstelle in der Klasse `Spielfeld` eine Methode `fuellen()`. Bei Aufruf von `fuellen()` sollen am oberen Spielfeldrand mehrere Reihen (z.B. drei Reihen) von nebeneinander liegenden Blöcken erzeugt werden.

Hinweis: Das Spielfeld ist 640 x 480 Pixel groß. Ein Block hat die Abmessungen 30 x 15 Pixel. Lässt man jeweils zwischen zwei Blöcken 5 Pixel Platz, passen genau 16 Blöcke in eine Reihe. Bei diesen

Werten macht es sich gut, wenn du den ersten Block an die Position (20, 10) platziert. Verwende zwei ineinander geschachtelte for-Schleifen zum Erzeugen von 3 Reihen mit jeweils 16 Blöcken (vgl. Roboter AB-10):

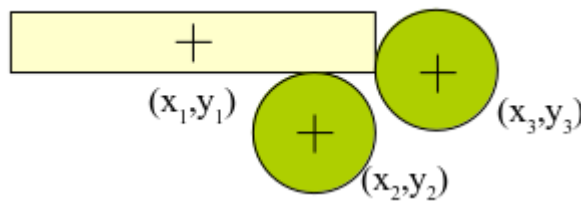
```
Wiederhole für y von 0 bis 2
  Wiederhole für x von 0 bis 15
    Erzeuge ein neues Block-Objekt
    Füge das Block-Objekt an der Stelle (20+x*35 | 15+y*15) ein
```

Außerdem soll automatisch ein Paddel erzeugt werden. Was hältst du von der Position (320, 460)?

Rufe die Methode `fuellen()` im Konstruktor von Spielfeld statt der Methode `prepare()` auf.

Erweiterungsideen

- Werden Blöcke oder das Paddle von der Seite getroffen, ist die Reflexion nicht korrekt. Um zu erkennen, ob der Ball von der Seite kommt, müsste man die x-Koordinate des Balls mit der des



Paddles vergleichen.

Wenn x-Koordinate des Balls $> x_1 + \text{halbe Paddlebreite}$, dann ist der Ball rechts vom Paddle (ist bei x_3 der Fall bei x_2 nicht). Dies kann man ausnutzen. Gleiches gilt für die linke Seite.

- Es gibt Gravitation, d.h. der Ball wird bei der Bewegung nach unten schneller, nach oben langsamer.
- Es sind pro Spiel maximal drei Bälle erlaubt. Damit muss das Spiel um einen Ball-Zähler ergänzt werden.

Jetzt macht es Sinn, am Ende des Spiels eine Meldung auszugeben, ob das Spiel gewonnen wurde (alle Blöcke sind abgeräumt und ein Ball ist noch im Spiel) oder verloren (es gibt noch Blöcke und alle drei Bälle sind verloren).

- Es gibt einen weiteren Punktezähler für getroffene Blöcke.
- Wenn das Paddle in Bewegung ist, wenn der Ball getroffen wird, verändert sich die Flugrichtung.
- Wenn das Paddle am Rand getroffen wird, dann verändert sich die Flugrichtung des Balls.
- Am oberen Rand wird der Ball beschleunigt reflektiert. Das erhöht den Schwierigkeitsgrad, wenn die meisten Blöcke unterhalb des Spielfeldrands schon abgeräumt sind.
- Es gibt verschiedenartige Blöcke mit unterschiedlichen Eigenschaften. Wird ein solcher Block getroffen ...
 - muss er noch einmal/noch zweimal getroffen werden, bis er verschwindet.
 - wird dabei der Ball beschleunigt/verlangsamt.
 - wird dabei die Bewegung des Paddels beschleunigt/verlangsamt.
 - wird dabei die Größe des Paddels verändert.
 - wird dabei ein Bonusball erzeugt.
 - ...
- Es gibt verschiedene Spiel-Levels. Dabei werden die Blöcke mit verschiedenen Eigenschaften

unterschiedlich auf dem Spielfeld angeordnet.

From:

<https://info-bw.de/> -

Permanent link:

<https://info-bw.de/faecher:informatik:mittelstufe:breakout:breakout06:start?rev=1697139192>

Last update: **12.10.2023 19:33**

