

## 2.6.1 Modellieren und Codieren von Algorithmen

### Arbeitsblatt 06 Zufallszahlen

#### Zufallszahlen

In EOS steht die Methode `zufall(von, bis)` zur Verfügung; z. B. `zufall(-200, 370)` erzeugt eine zufällige Zahl von -200 bis 370, um die Position einer Wand festzulegen (siehe Grafiken unten).

- Wenn ein Wert an eine Methode übergeben wird, bezeichnen wir das als **Parameter**.  
Die beiden *Parameter* -200 und 370 geben das Intervall an, in dem die Zufallszahl liegen soll.

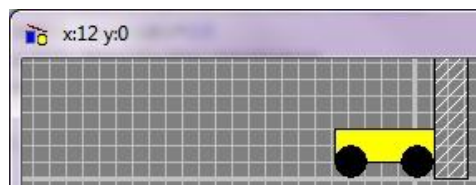
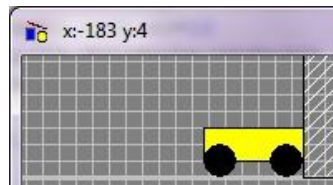
Die Methode `zufall()` liefert eine Zahl, die man als **Rückgabewert** bezeichnet. Diese Zahl kann in einer Variablen gespeichert werden, um sie verwenden zu können.

1. Die Wand aus Aufgabe 1 soll per Zufallsgenerator an einer beliebigen Stelle eingezeichnet werden.  
Die Seifenkiste fährt dann vom linken Rand aus bis zu dieser Wand.  
Ergänze die vorige Programmversion *seifenkiste6.eos* und speichere die neue Version als *seifenkiste7.eos*.  
(Vorlagedatei: v04-seifenkiste6.eos)  
Hinweis: Die Zufallszahl soll für den linken Rand der Wand gelten. Da die Wand 20 Pixel breit ist, kann der Wert für den rechten Rand mit `xwand+20` angegeben werden.

```
f: FENSTER
xwand: Integer
seifenkiste1: GRUPPE
kiste1: RECHTECK
rad01: KREIS
rad02: KREIS
wand1: RECHTECK
```

```
...
xwand:=zufall(-200, 370)
wand.linksObenSetzen(xwand, 200)
wand.rechtsUntenSetzen(xwand+20, 0)
wand.füllartSetzen(schraffiert)
f.zeichne(wand)

solange kiste.rechts < xwand tue
    seifenkiste.verschieben(1, 0)
*solange
```



Damit wird im Programm die ganzzahlige Variable `xwand` neu eingeführt.

#### SeifenkisteMitWand

```
xwand:Integer
seifenkiste1:Gruppe
kiste1:Rechteck
rad01:Kreis
rad02:Kreis
wand1:Rechteck
```

Auch in der Wiederholung mit Anfangsbedingung kann die Variable verwendet werden:

```
solange <Bedingung> tue
    <Anweisungen>
*solange
```

#### Die Klasse *Textfeld* in EOS

In EOS können Objekte der Klasse *Textfeld* verwendet werden. Die wichtigsten Attribute und Methoden sind in dem Klassendiagramm rechts aufgeführt. Die Breite und Höhe wird automatisch angepasst.

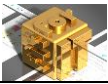
- Texte werden in Anführungszeichen ("Text") oder Hochkomma ('Text') geschrieben.
  - Beispiel: `crash.zeileHinzufügen ('crash!')`
- Mit Hilfe des Additionszeichens (+) können Texte verkettet („aneinandergesetzt“) werden.
  - Beispiel: `position.zeileHinzufügen ('Position der Wand: ' + xwand)`

2. Ergänze in deinem EOS-Programm die Textfelder `crash` und `position`, nachdem die Seifenkiste an der Wand angestoßen ist (*seifenkiste8.eos*).

#### Textfeld

```
links:Integer
oben:Integer
```

```
linksObenSetzen(links, oben)
zeileHinzufügen(text)
zeilenLöschen()
```



## 2.6.1 Modellieren und Codieren von Algorithmen

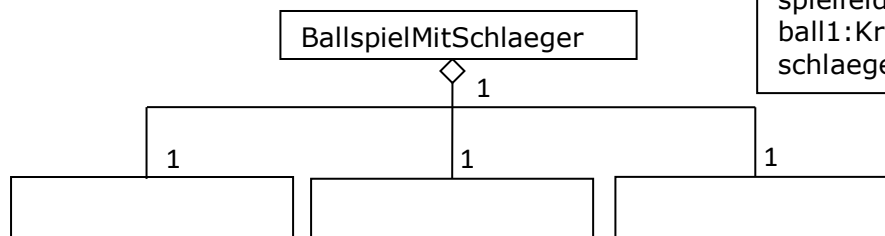
### Arbeitsblatt 06 Zufallszahlen

3. Erweitere dein EOS-Programm *ballspiel2.eos* dahingehend, dass der „Schläger“ zu Beginn an eine zufällige Position im Intervall von 100 bis 400 gesetzt wird (*ballspiel3.eos*).

(Vorlagedatei: v05-ballspiel2.eos)

Hinweis: Du benötigst eine weitere ganzzahlige Variable, um die Zufallszahl für die x-Position des Schlägers zu speichern.

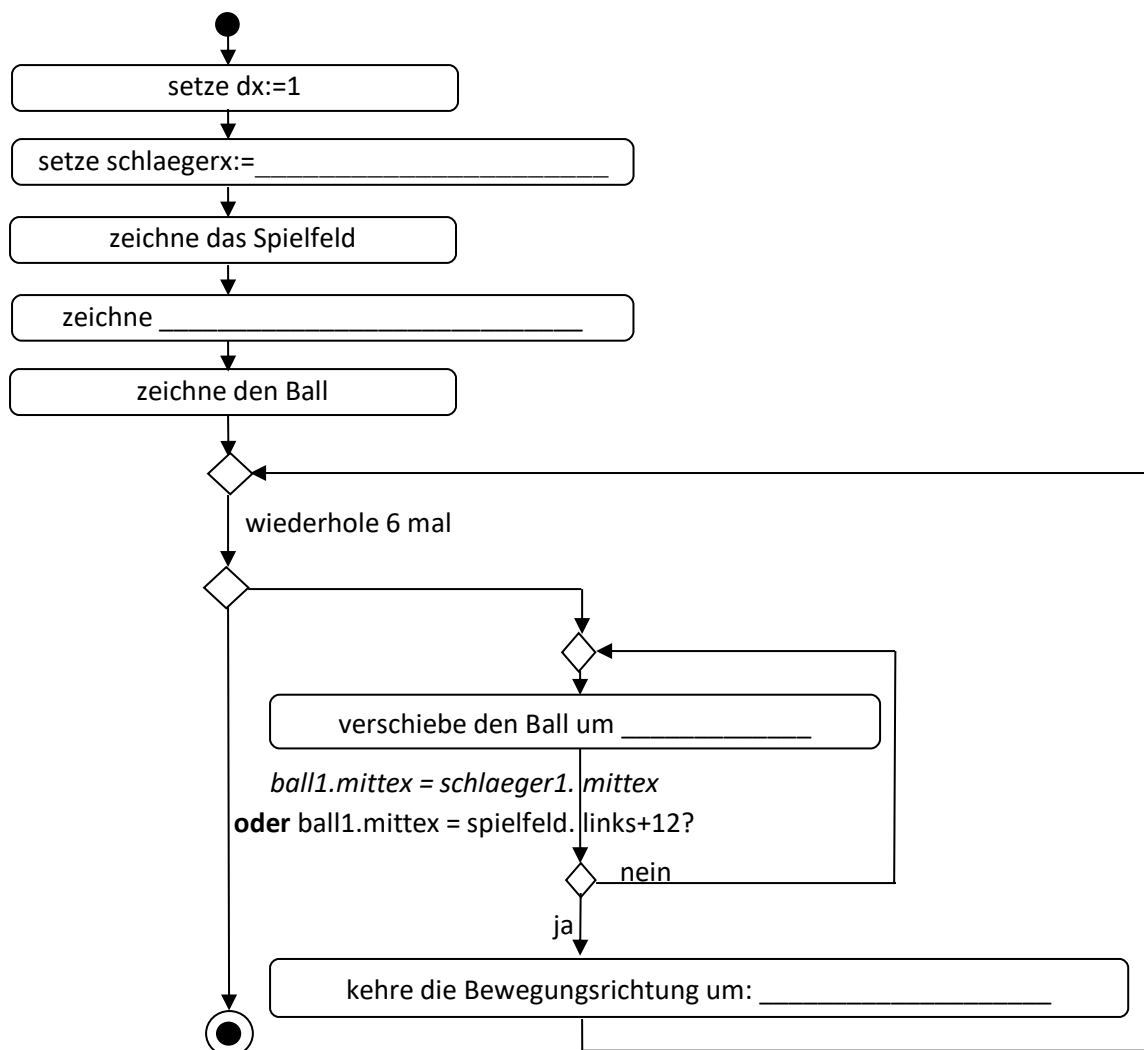
- Gegeben ist das Klassendiagramm rechts zu dieser Programmversion *BallspielMitSchlaeger*. Ergänze die Klassenstruktur in dem Klassendiagramm unten.



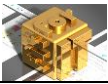
BallspielMitSchlaeger

**dx:Integer**  
**schlaegerx:Integer**  
spielfeld:Rechteck  
ball1:Kreis  
schlaeger1:Ellipse

- Ergänze das Aktivitätsdiagramm:



- Ergänze den Programmcode in EOS und teste das Programm, indem du es mehrfach startest und die Position des Schlägers sowie den „Flug“ des Balles kontrollierst.



## 2.6.1 Modellieren und Codieren von Algorithmen

### Arbeitsblatt 06 Zufallszahlen

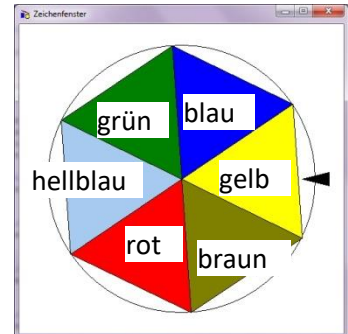
#### Zufallsversuche

Wenn bestimmte Ereignisse eintreten *können*, die Ergebnisse aber nicht vorherzusagen sind, spricht man von einem **Zufallsversuch**. Um Gesetzmäßigkeiten zu finden, wie *wahrscheinlich* bestimmte Ereignisse eintreten, kann der Versuch mit Hilfe eines Computers praktisch beliebig oft wiederholt werden.

Ein einfaches Beispiel für einen Zufallsversuch ist die Drehung eines Glücksrads.

Im Beispiel können sechs Ereignisse eintreten: Das Rad ist in sechs Farbfelder unterteilt; die Markierung zeigt nach der Drehung des Rads in eines der sechs Felder.

4. Mit einem EOS-Programm soll ein Glücksrad gezeichnet werden.  
Erstelle dazu im Programm die Methode `zeichneGlücksrad` und speichere deine Datei unter `gluecksrad1.eos`.



```
f:FENSTER
```

```
rad1:GRUPPE
```

```
umkreis:KREIS
```

```
feld:DREIECK
```

```
markierung:DREIECK
```

```
zeichneGlücksrad ()
```

```
methode zeichneGlücksrad
```

```
...
```

```
markierung.eckenSetzen(180,0,220,-10,220,10)
```

```
markierung.füllfarbeSetzen(schwarz)
```

```
umkreis.mittelpunktSetzen(0,0)
```

```
umkreis.radiusSetzen(200)
```

```
umkreis.füllartSetzen(durchsichtig)
```

```
rad1.schlucke(umkreis) ←
```

```
feld.eckenSetzen(0,0,173,-100,173,100)
```

```
feld.füllfarbeSetzen(gelb)
```

```
rad1.kopiere(feld)
```

```
feld.drehenUm(0,0,60)
```

```
feld.füllfarbeSetzen(blau)
```

```
rad1.kopiere(feld) ←
```

```
...
```

```
f.zeichne(rad1)
```

```
f.zeichne(markierung)
```

```
ende
```

- Ergänze in deinem Programm das Erstellen der restlichen Farbfelder.

```
f:FENSTER
```

```
LinksOben=200,100
```

```
Breite=500
```

```
Höhe=500
```

```
Hintergrundfarbe=weiß
```

```
Gitter=aus
```

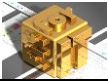
```
zeichneGlücksrad ()
```

Objektdiagramm

Die Methode `gruppe.schlucke (Objekt)` fügt das Objekt zu der Gruppe hinzu.

Die Methode `gruppe.kopiere (Objekt)` erzeugt aus dem ursprünglichen Objekt ein neues Objekt und fügt diese Kopie zu der Gruppe hinzu.

Man kann also immer neue Kopien des Objekts erstellen.



## 2.6.1 Modellieren und Codieren von Algorithmen

### Arbeitsblatt 06 Zufallszahlen

5. Das Glücksrad soll eine zufällige Anzahl im Bereich von zwei bis vier Umdrehungen ausführen. Erstelle dafür die Methode `dreheGlücksrad`.

Das Rad wird wiederholt um ein Grad gedreht. Insgesamt muss das Rad also um einen Winkel im Bereich von 720 bis 1440 Grad gedreht werden.

Auf die Variable `Drehung` wird eine Zufallszahl zugewiesen, die in diesem Bereich liegt.

In der Variablen `n` wird die Anzahl der durchgeführten Drehungen mitgezählt. Eine solche Variable bezeichnet man auch als *Zählvariable*.

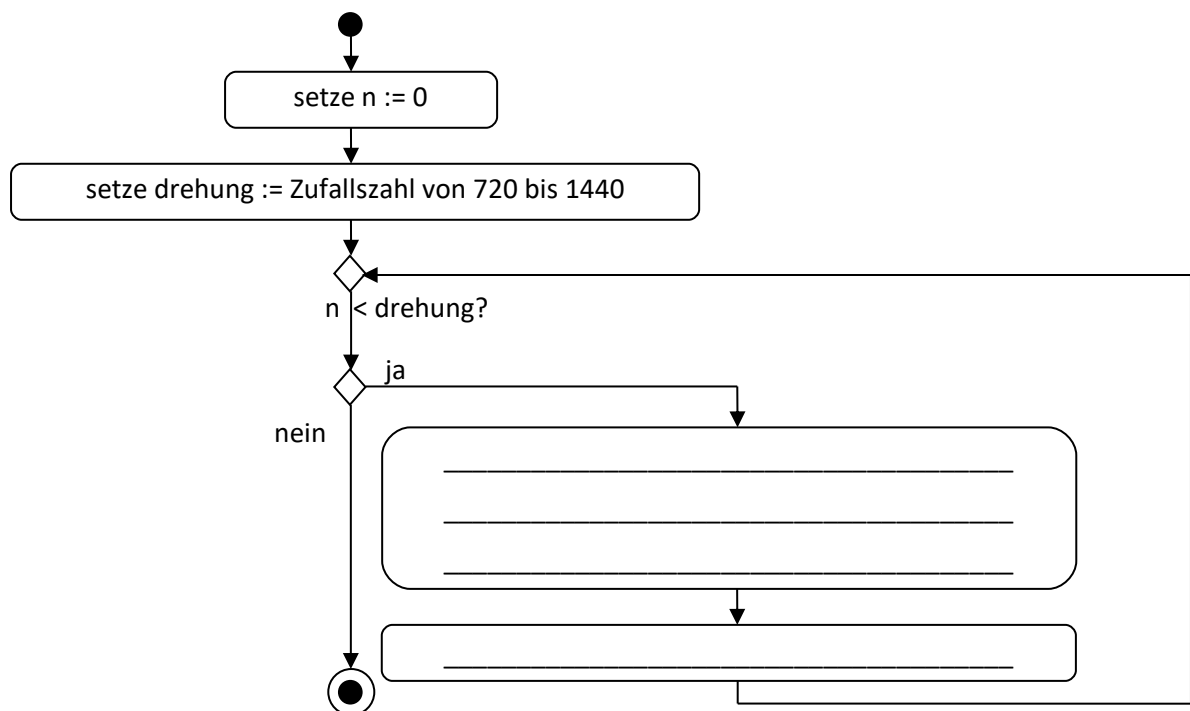
#### Glücksrad

`drehung:Integer`  
`n:Integer`  
`rad1:Gruppe`  
`umkreis:Kreis`  
`feld:Dreieck`  
`markierung::Dreieck`

`zeichneGlücksrad ()`  
`dreheGlücksrad ()`

*Klassendiagramm*

6. Ergänze das Aktivitätsdiagramm zur Methode `dreheGlücksrad`.



7. Was versteht man unter einem Zufallsversuch?

8. Welche Zufallsgeräte kennst du?