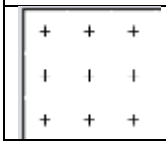
Fachbereich Informatik 11. Klasse/Opitz	**Grundlagen der Programmierung** Blatt 1	Datum:
	Klassen, Objekte und Methoden	Name:

In Karels Welt ist es ihm möglich, bestimmte Aufgabenstellungen mithilfe seiner **Methoden** zu erfüllen. In dieser Welt kann Karel auf zwei verschiedene **Objekte** (Beeper und Wände) treffen, wobei die Welt selbst auch ein **Objekt** ist. Karel besitzt eigene **Methoden**, z.B. kann er Beeper aufheben oder ablegen. Er kann zudem laufen und sich um 90 oder 180 Grad drehen; aber er kann nicht durch Wände laufen oder die Welt verlassen.

Aufgabe 1: Ergänzen Sie die Tabelle.

Objekt	Bezeichnung	Eigenschaften (= Attribute)
	Karel	Kennt seine Position in Form von x- und y- Koordinate Hat eine Farbe. Kennt seine Blickrichtung.
		
		

.

Gleichartige Objekte werden in **Klassen** zusammengefasst. Eine **Klasse** ist ein Bauplan mit abstrakten Eigenschaften (z.B. Haarfarbe, Größe, Geschlecht, Name). Ein Objekt ist eine Instanz der Klasse mit konkreten Eigenschaften (Haarfarbe = braun; Größe = 1,86 m; Geschlecht = männlich; Name = Michael Müller).

Beispiele:

Max, Hänschen und Lieschen sind Objekte der Klasse Mensch.

Ein Dackel, eine Bulldoge oder ein Pudel sind Objekte der Klasse Hund.

Aufgabe 2: Finden Sie 3 weitere Beispiele für Klassen und deren Objekte.

Aufgabe 3:

Wenn in Karels Welt mehrere Beeper vorkommen, dann sind dies alles Instanzen der Klasse Beeper. Wodurch unterscheiden sich diese Beeper?

Fachbereich Informatik 11. Klasse/Opitz	**Grundlagen der Programmierung** Blatt 2	Datum:
	Klassen, Objekte und Methoden	Name:

In einer **Klassendokumentation** werden eine Klasse und ihre Methoden so beschrieben, dass sie von Anwendern ohne Kenntnisse der eigentlich dahinterstehenden Programmierung verwendet werden kann.

Aufgabe 4: Ergänzen Sie die Klassendokumentation für die Klasse Roboter

Klassendokumentation Karel

moveForward()
Karel bewegt sind ein Feld in Blickrichtung weiter, sofern er nicht von einer Wand oder dem Ende der Welt daran gehindert wird. Dann erscheint eine Meldung.

turnLeft()
Karel dreht sich um 90 Grad nach links.

turnRight()

turnAround()

dropBeeper()

pickBeeper(9

Alle Aufträge an Karel sind sogenannte **Methoden**, die für diese Klasse implementiert sind. In einer **Methode** wird also das gewünschte Verhalten des Objektes programmiert.

Um Karel einen Auftrag erledigen zu lassen, müssen wir das gewünschte Verhalten in einer Methode festlegen.

In der Aufgabe „HomeRun“ hieß die Methode …

```
void karelsHomeRun()
{
moveForward();
pickbeeper();
..
}
```

Methodenkopf
Der Name kann frei gewählt werden, muss jedoch mit einem kleinen Buchstaben begonnen werden.
In die Klammer können Übergabewerte kommen, wie z.B. bei der Methode Repeat(4).

Jede Anweisung endet mit einem Semikolon

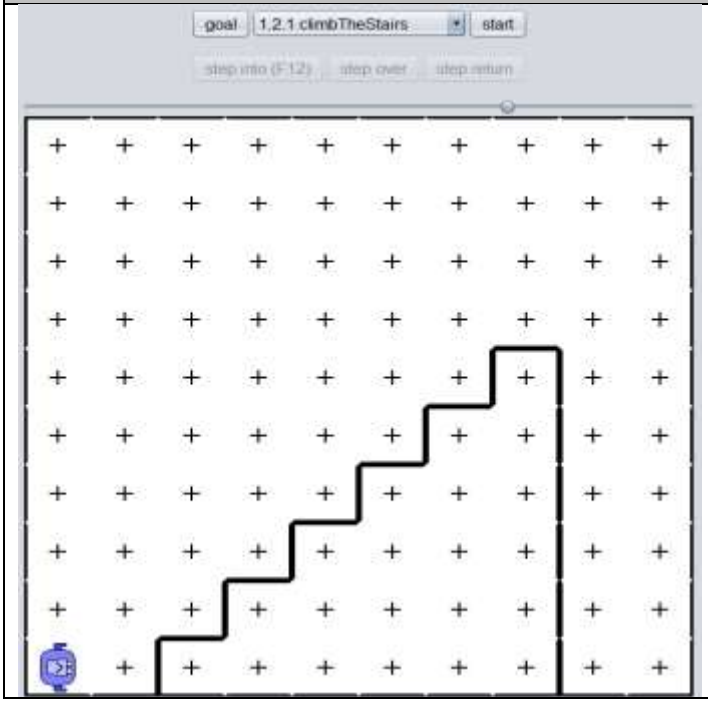
Methodenrumpf
Alle Anweisungen einer Methode sind von geschweiften Klammern umschlossen

osz lotis Fachbereich Informatik 11. Klasse/Opitz	**Grundlagen der Programmierung** Blatt 3	Datum:
	Klassen, Objekte und Methoden	Name:

Statt nur die vorgegebenen Methoden zu benutzen, können auch **eigene Methoden definiert** werden. Diese wiederrum können von anderen Methoden aufgerufen werden.


```
9
10 void defuseTwoBombs()
11 {
12     laufenUndUmdrehen();
13     pickBeeper();
14     laufenUndUmdrehen();
15     turnLeft();
16     laufenUndUmdrehen();
17     pickBeeper();
18     laufenUndUmdrehen();
19 }
20
21 void laufenUndUmdrehen()
22 {
23     repeat(9){
24         moveForward();}
25     turnAround();
26 }
27
28
29
30
```

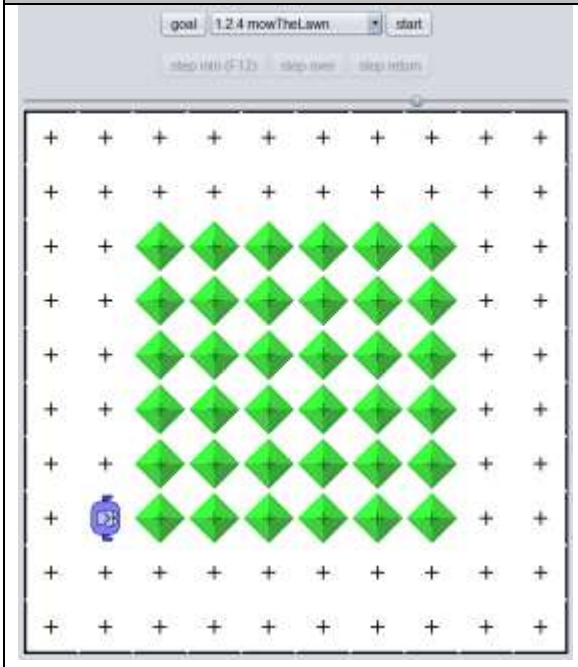
Aufgabe 5: Da der Fahrstuhl wiedermal kaputt ist, muss Karel die Treppen hochlaufen. Da er von seinem Home Run noch voller Energie ist, sollte es kein Problem für ihn sein. Lösen Sie unter Verwendung einer selbst definierten Methode.




```
void climbTheStairs()
{
```

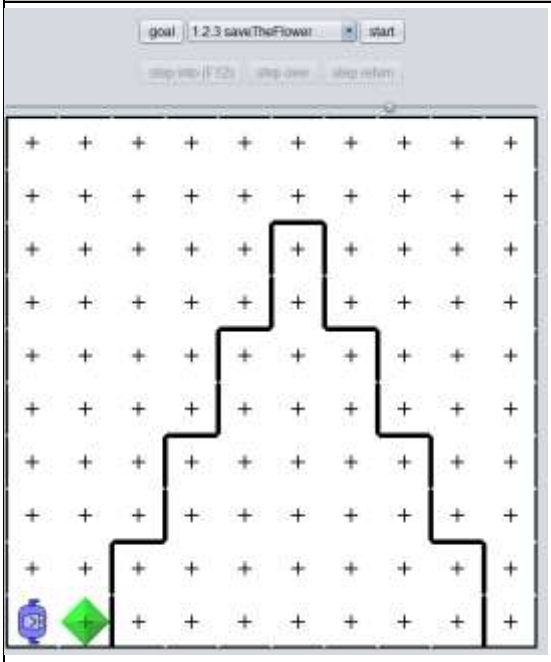
Fachbereich Informatik 11. Klasse/Opitz	**Grundlagen der Programmierung** Blatt 4	Datum:
	Klassen, Objekte und Methoden	Name:

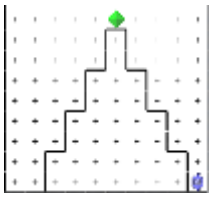
Aufgabe 6: Karel versprach seiner Tante im Garten zu helfen. Da Sie bereits das Unkraut entfernt hat, kann er sich voll auf das Rasenmähen konzentrieren. Helfen Sie Karel unter Verwendung selbstdefinierter Methoden.




```
void mowTheLawn()
{
```

Aufgabe 7: Karel will in den Alpen einen Berggipfel erklimmen und eine Blume näher zur Sonne bringen. Helfen Sie der Blume ins Sonnenlicht und Karel über den Gipfel unter Verwendung selbstdefinierter Methoden.





```
void saveTheFlower()
{
```

Jugendwettbewerb Informatik: Programmieren – leichter, als du denkst! (jwinf.de)