



UNIVERSITÄT
LEIPZIG

Didaktik der Informatik

Lehramt Informatik (Gy, MS, BS, FS)
Modul Didaktik der Informatik - E-Learning und Tools

Werkzeuge für den Informatikunterricht

---- Java-Kara ----

Empfohlen für Klasse: Gy & GS 8 Klasse

Autor: Tom Bogatsch

Lizenz: Dieses Werk ist lizenziert unter [CC BY 4.0](https://creativecommons.org/licenses/by/4.0/)



1. Kurzvorstellung

Java-Kara beruht auf der Programmiersprache Java und bietet so für Schüler*Innen einen ersten Kontakt mit einer komplexeren Programmiersprache. Hauptfigur ist dabei Kara, der Marienkäfer, welcher in einer interaktiven Welt lebt. Durch manuelle Eingabe oder Programmierung mittels Java kann mit dieser Welt agieren werden. Die Programmierung erfolgt dabei mittels vorgefertigter Befehle und ist an die Lösung bestimmter Aufgaben angepasst. Besagte Aufgaben findet man direkt im Programm oder auf der Webseite der Hersteller.

Einordnung in den Lehrplan

Gymnasium:

Klasse	Lernbereich	Themen
8	1	<ul style="list-style-type: none">- Algorithmen kennenlernen- Eigenschaften von Algorithmen auf Lebenswelt übertragen- Beherrschen von Algorithmen in reduzierter Umgebung-> Sequenz, Verzweigung und Wiederholung- Bedeutung von Algorithmen in der Gesellschaft

Lernziele:

1. Die SuS wenden gelernte Eigenschaften von Algorithmen an
2. Die SuS übertragen Wissen zu Abläufen, die sie aus ihrer Erfahrungswelt kennen, auf algorithmische Eigenschaften

Oberschule:

Klasse	Lernbereich	Themen
8	1	<ul style="list-style-type: none">- Algorithmen kennenlernen- Eigenschaften von Algorithmen auf Lebenswelt übertragen- Beherrschen von Algorithmen in reduzierter Umgebung-> Sequenz, Verzweigung und Wiederholung- Umgang mit Fehlermeldungen von Informatiksystemen- Einblick in die Grenzen der Berechenbarkeit- Positionieren zur Bedeutung von Algorithmen in der Gesellschaft

Lernziele:

1. Die SuS wenden gelernte Eigenschaften von Algorithmen an
2. Die SuS übertragen Wissen zu Abläufen, die sie aus ihrer Erfahrungswelt kennen, auf algorithmische Eigenschaften
3. Die SuS erschließen sich Gründe für Fehlermeldungen und beheben diese

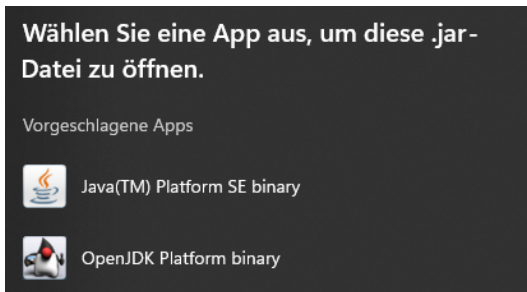
2. Öffnen von ... Speichern und Schließen

Schritt 1: Installation

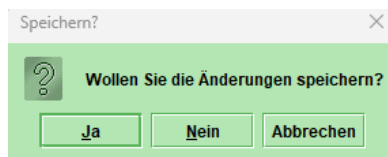
- Java
 - Java-Kara kommt ohne Java. Dies muss also zunächst installiert werden
 - Ist Java installiert? (cmd -> java --version)
 - Wenn Nein => Von Java Website installieren
 - Darauf achten, dass es sich um die nicht kommerzielle Version OpenJDK handelt
- Java-Kara
 - <https://www.swisseduc.ch/informatik/karatojava/javakara/>

Schritt 2: Grundfunktionen kennen

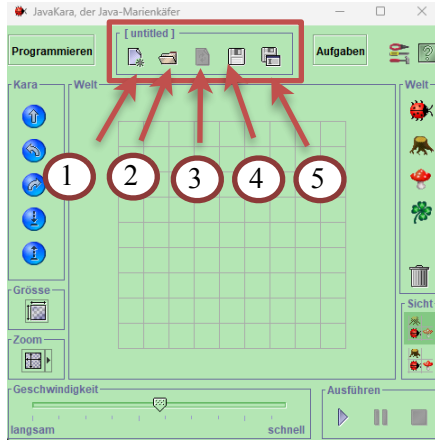
- Programm öffnen
 - Doppelklick auf das Programm und ggf. Java Version wählen



- Programm schließen
 - Sollte das Programm ohne das vorherige Speichern von Welt oder Programmiercode erfolgen, wird man daran erinnert und kann dies nachholen



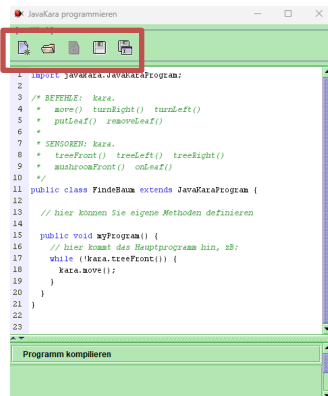
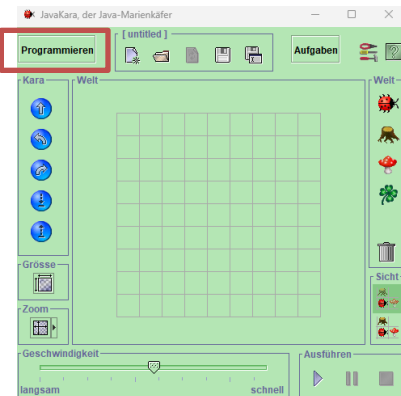
- Interaktive Spielwelt



1. Neue Welt
2. Öffnen einer gespeicherten Welt
3. Öffnet Welt erneut (Entfernt alle Änderungen)
4. Speichert eine Welt
5. Speichert eine Welt an einem bestimmten Ort

- Programmieren

- Speichern, Öffnen und neue Welt wie bei Spielwelt



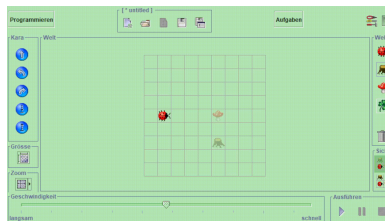
3. Weitere Bedienhandlungen

Objekte der Spielwelt

In der Welt von Kara existieren 4 verschiedene interaktive Objekte

	Symbol	Erklärung
Kara		Spielfigur. Kann sich zweidimensional bewegen
Baum		Festes Hindernis
Pilz		Kann verschoben werden.
Kleeblatt		Kann gesammelt und gesetzt werden

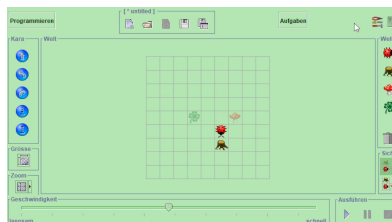
Objekte werden in die Welt gezogen und interagieren so mit Kara



Programmieren

Das Verhalten von Kara gegenüber Objekten kann nicht nur manuell stattfinden, sondern man kann auch programmieren, was Kara für Aufgaben erfüllen soll. Dazu öffnet man wie oben beschrieben das "Programmieren"-Fenster und kann dort mithilfe von vorgefertigten Methoden und eigenem Code Kara automatisch Aufgaben erfüllen lassen.

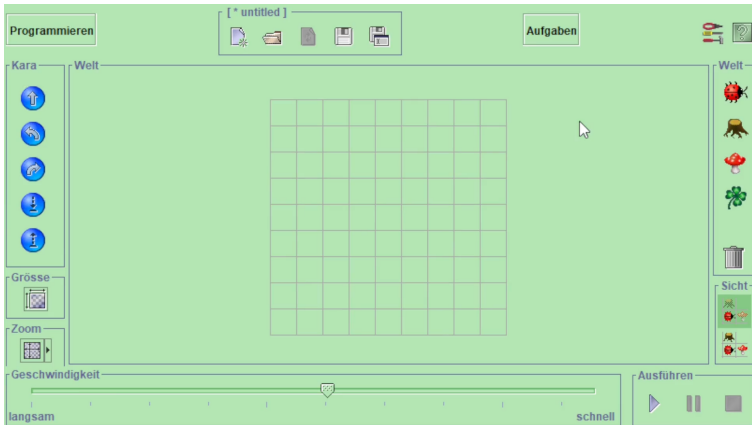
Die möglichen Methoden findet man über das Fragezeichen in der rechten oberen Ecke unter "JavaKaraProgramm"



Beispielvideo: (Mit Audio)

Aufgabe:

Kara befindet sich auf einem 1 mal x Pfad mit einem Baum am Ende. Lasse Kara bis zum Baum laufen und platziere auf dem Weg Kleeblätter da, wo keine Kleeblätter liegen und nimm Kleeblätter auf, die bereits liegen.



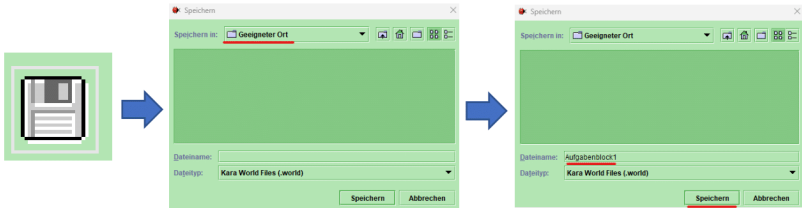
Aufgaben:

Unter dem Button "Aufgaben" ist bereits ein Kollektiv an Aufgaben vorhanden. Zu jeder Aufgabe gibt es die Spielwelt und eine Erklärung der Aufgabe. In der Version für Lehrkräfte gibt es ebenfalls die Programmierlösungen (Diese fehlt bei der Version für die Schüler*Innen). Die Beispielaufgabe von oben ist dort in abgewandelter Form zu finden.

Aufgabenblock 2: Speichern und Öffnen

Aufgabe 1

- Speichere die Spielwelt vom Aufgabenblock 1 ab einem geeigneten Ordner in deinem Netzwerkordner, sodass du sie jederzeit wiederfinden kannst. Gib ihr außerdem den Dateinamen **Aufgabenblock1**



Aufgabe 2

- Erstelle eine neue Spielwelt über folgendes Symbol:



Aufgabenblock 3: Erstes Programm

Vorabinformationen Programmoberfläche (Java)

- Es folgt eine kleine Erklärung der Inhalte des „Programmieren“-Feld in Java

Nicht relevant für uns

Auflistung von für uns vorerst relevanten Sensoren und Befehlen (Hierzu später mehr)

Hier hinein wird unser Programm geschrieben. Ersetze „FindeBaum“ mit dem Namen deines Programmes

Vorerst nicht relevant für uns

Entferne die while-Schleife und ihren Inhalt, denn hier kommt unser Programmcode hin

```
public void myProgram() {
    // hier kommt das Hauptprogramm hin, zB:
}
```

Kompiliert unser Programm und verbindet es mit unserer aktuellen Welt. Durch Betätigung der Fortfahren-, Pausieren- und Stopptaste im Spielwelt-Fenster wird es anschließend ausgeführt

Vorabinformationen Programmoberfläche (Python)

- Es folgt eine kleine Erklärung der Inhalte des „Programmieren“-Feld in Python

PythonKara programmieren

```

1 # BEFEHLE: kara.
2 #   move() turnRight() turnLeft()
3 #   putLeaf() removeLeaf()
4
5 # SENSOREN: kara.
6 #   treeFront() treeLeft() treeRight()
7 #   mushroomFront() onLeaf()
8 #
9
10 # hier können Sie eigene Methoden definieren
11
12 # hier kommt das Hauptprogramm hin, zB:
13 while not kara.treeFront():
14     kara.move()
15
16
17 # hier kommt das Hauptprogramm hin, zB:
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

```

Auflistung von für uns vorerst relevanten Sensoren und Befehle (Hierzu später mehr)

Vorerst nicht relevant für uns

Hier kommt der Programmcode hin. Entferne hierzu den bereits vorhandenen Code

Das Feld zum Kompilieren entfällt im Gegensatz zu Kara, da Code in Python live kompiliert wird

Die Ausführung des Programms wird durch das Betätigen der Fortfahren-, Pausieren- und Stopp-Taste gesteuert

Ausführen

Vorabinformationen Sensoren und Befehle

Befehle (Groß- und Kleinschreibung beachten und ; (Java) nicht vergessen!)

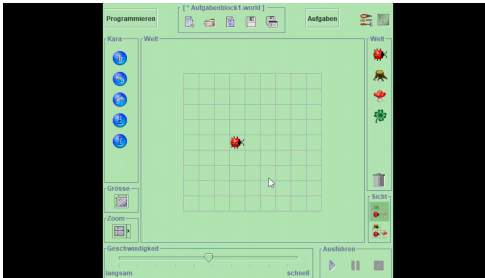
kara.move()	Bewegt Kara ein Kästchen nach vorne
kara.turnRight()	Dreht Kara um 90° nach rechts
kara.turnLeft()	Dreht Kara um 90° nach links
kara.putLeaf()	Platziert ein Kleeblatt unter Kara
kara.removeLeaf()	Hebt ein Kleeblatt unter Kara auf

Sensoren (Groß- und Kleinschreibung beachten und ; (Java) nicht vergessen!)

kara.treeFront()	Befindet sich ein Baum vor Kara?
kara.treeLeft()	Befindet sich ein Baum links von Kara?
kara.treeRight()	Befindet sich ein Baum rechts von Kara?
kara.mushroomFront()	Befindet sich ein Pilz vor Kara?
kara.onLeaf()	Befindet sich Kara auf einem Blatt?

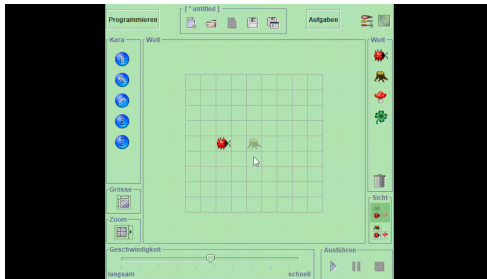
Aufgabe 1

- Platziere Kara in einer leeren Spielwelt
- Drücke die „Programmieren“-Taste, ändere den Namen des Programms zu **Block3** und speichere das Programm ebenfalls unter diesem Namen
- Erweitere dein leeres Programm, sodass sich Kara beim Betätigen der Fortführen-Taste ein Feld nach vorn bewegt



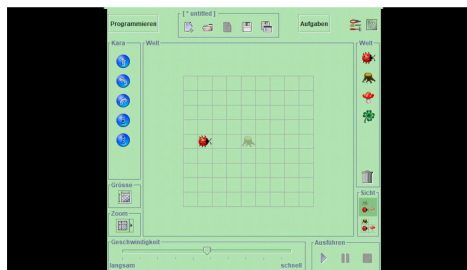
Aufgabe 2

- Stelle zwei Felder vor Kara einen unbeweglichen Baumstumpf und lasse Kara um diesen herumlaufen



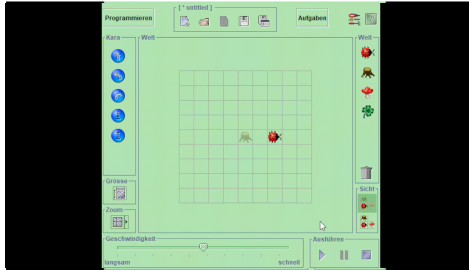
Aufgabe 3

- Erweitere das Programm mithilfe einer Verzweigung so, dass Kara nach vorn läuft, wenn kein unbeweglicher Baumstumpf vor ihr liegt und diesen umläuft, wenn ein unbeweglicher Baumstumpf vor ihr liegt



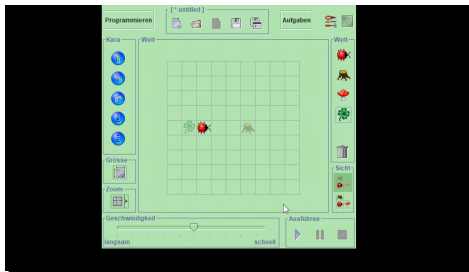
Aufgabe 4

- Erweitere das Programm mithilfe einer Schleife so, dass Kara mit einmaliger Betätigung der Fortfahren-Taste unendlich lange nach vorn läuft und einen unbeweglichen Baumstumpf umläuft, wenn sie auf einen trifft



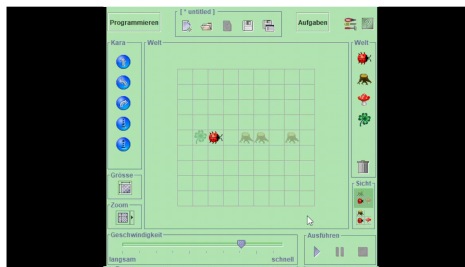
Aufgabe 5

- Platziere auf dem Weg, den Kara geht, ein Kleeblatt. Kara soll so lange laufen und unbewegliche Baumstümpfe umlaufen, bis sie auf dem Kleeblatt steht. Wenn sie auf dem Kleeblatt steht, soll dieses aufgehoben werden und Kara anschließend aufhören zu laufen



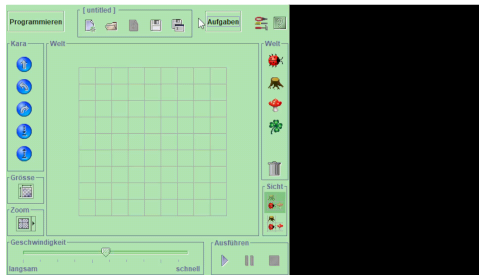
Aufgabe 6

- Platziere nun zwei unbewegliche Baumstümpfe nebeneinander und schau, ob dein Programm noch funktioniert. Wenn nicht, ändere es ab.
- Speichere das Programm anschließend ab und öffne ein neues „Programmieren“-Feld



Aufgabe 7 Abschlussaufgabe

- Diese Aufgabe baut auf einer in Kara bereits vorhandenen Spielwelt auf. Um diese Welt zu laden, gehe nach den folgenden Schritten vor:
 - o Drücke auf den „Aufgaben“-Knopf
 - o Drücke auf das „Top-Down“-Feld und wähle Kleeblattsuche im Wald III (einfach) aus
 - Hier kannst du dir zunächst die Aufgabenstellung durchlesen
 - o Anschließend kannst du die Spielwelt(en) über Welten und anschließendes Klicken auf „Wald mit Rundgang x“ laden
 - Fange mit Rundgang 1 an und schaue, ob dein Programm auch für die anderen Welten funktioniert



Aufgabe 8 Zusatzaufgabe

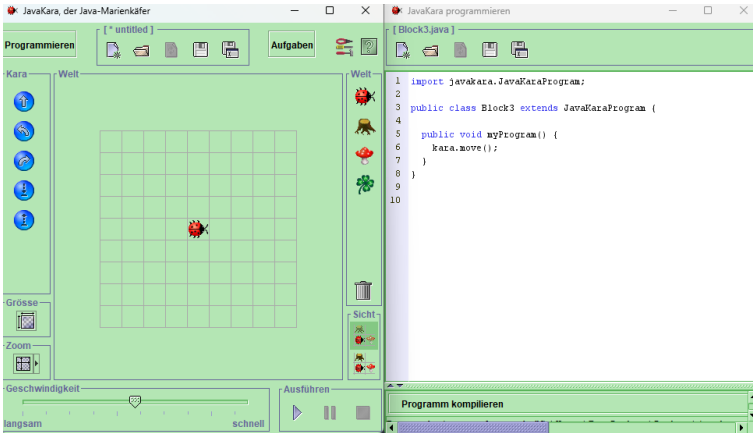
- Suche dir eine beliebige Aufgabe aus dem „Aufgaben“-Fenster aus und versuche sie zu lösen

Lösungen

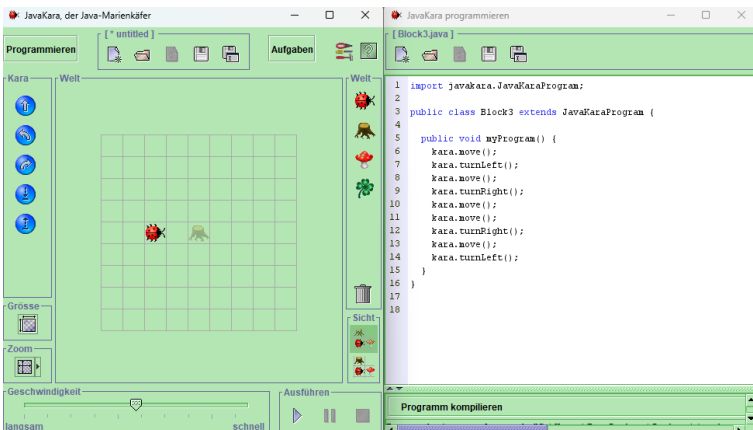
A1 und A2 Siehe Seiten mit Aufgaben

Aufgabe 3

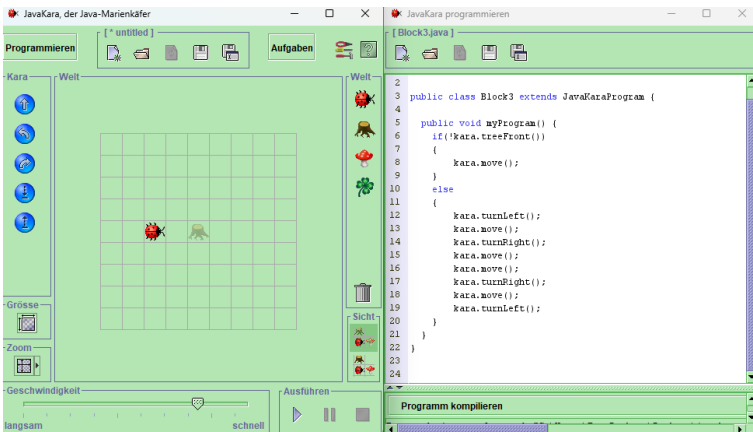
3.1



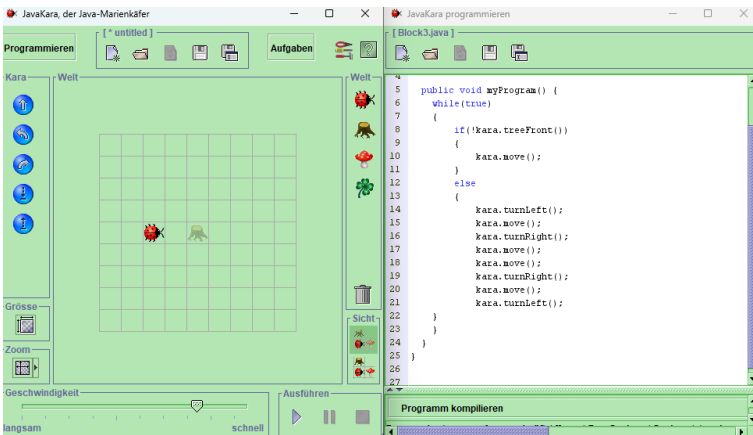
3.2



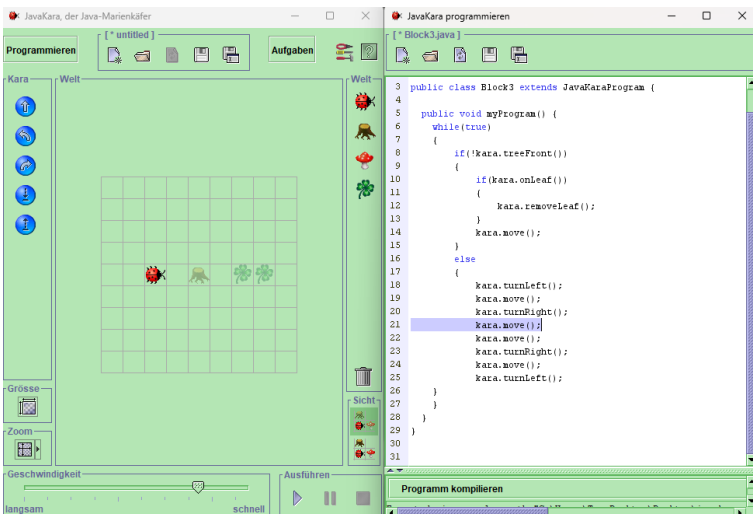
3.3



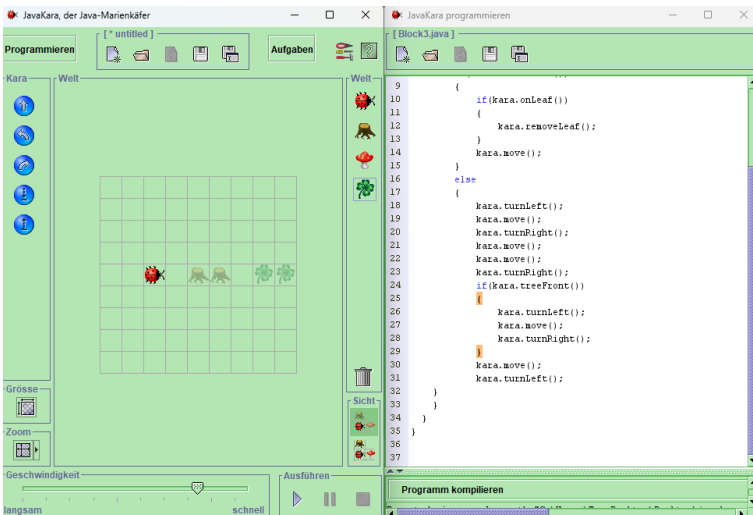
3.4



3.5



3.6



3.7

