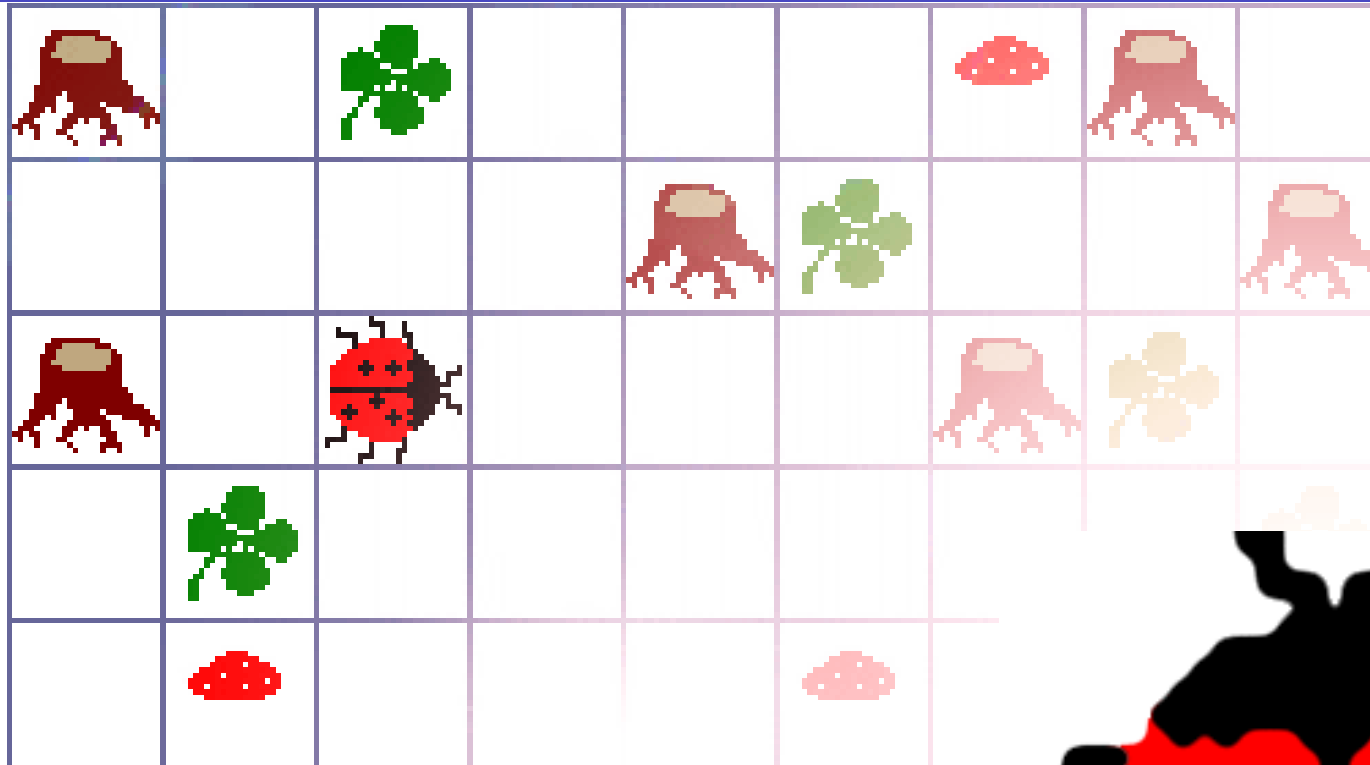


# *Kara, der programmierbare Marienkäfer !*

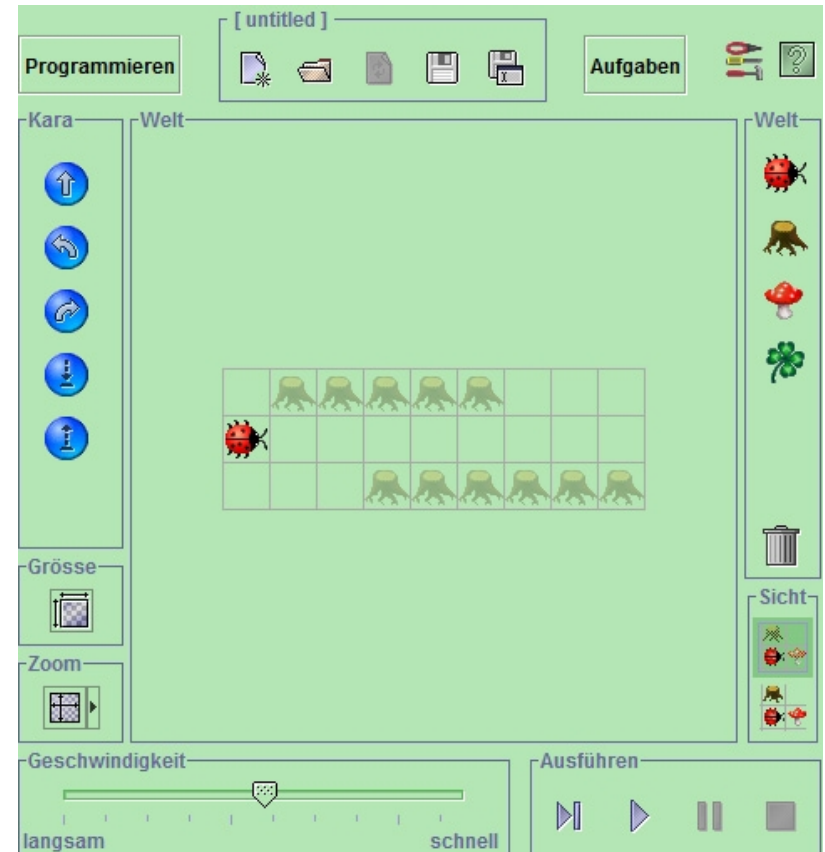


# *In der Welt des Marienkäfers...*

... gibt es:

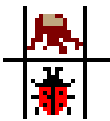
- unbewegliche **Baumstümpfe**,
- **Pilze**, die Kara verschieben und
- **Kleeblätter**, die Kara legen und aufnehmen kann

... und natürlich Kara selbst!



# *Kara, der Marienkäfer...*

... hat **Sensoren**, mit denen er seine Umwelt wahrnimmt:

 stehe ich vor einem Baumstumpf?


 ist links von mir ein Baumstumpf?


 ist rechts von mir ein Baumstumpf?


 stehe ich vor einem Pilz?

 stehe ich auf einem Kleeblatt?

... versteht einige **Befehle**, die er folgsam ausführt:

 mache einen Schritt vorwärts!

 drehe um 90° nach links!

 drehe um 90° nach rechts!

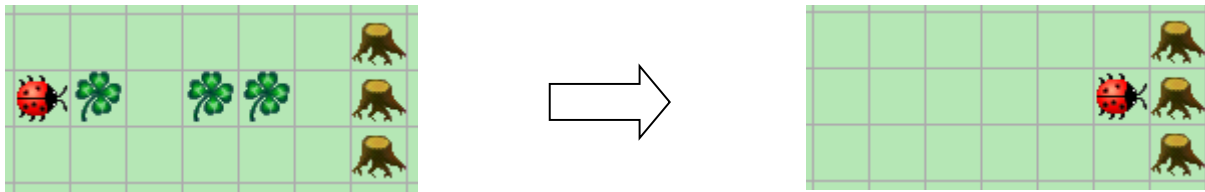
 lege ein Kleeblatt hin!

 nimm ein Kleeblatt auf!

# Kara, die Blättersammlerin

## Aufgabe:

Schreibe ein Programm, das Kara geradeaus bis zum nächsten Baumstrunk führt. Kara weiss, dass geradeaus vor ihm ein Baumstrunk steht. Liegt auf einem Feld ein Blatt, soll Kara es aufnehmen. Beim Baumstrunk angekommen ist das Programm zu beenden. (Achtung: Nur wenn ein Baumstrunk vor Kara liegt, läuft das Programm korrekt.)



Das Bild links zeigt die Ausgangslage und auf dem Bild rechts siehst du wie es aussehen muss, nachdem Kara ihren Auftrag erledigt hat.

# Kara, die Blättersammlerin

## Lösung (1 von 3):

### 1. Welche Zustände kommen vor?

Wir brauchen **zwei** Zustände: **einen** Zustand zum **Sammeln der Kleeblätter** und den **Stoppzustand**. Der Stoppzustand soll erreicht werden, sobald Kara vor dem Baum steht. Unseren bisherigen Automaten haben die Arbeit nie beendet, sondern einfach auf neue Benutzende gewartet. Bei Kara ist es jedoch anders, wir wollen, dass Kara sagt, wenn er mit der Arbeit fertig ist. Das erreichen wir dadurch, dass er nach getaner Arbeit in den Stoppzustand wechselt.



Die beiden Zustände des Automaten für die Blättersammlerin. So lange Kara mit der Arbeit nicht fertig ist, soll er im Zustand „Blätter sammeln“ bleiben. Sobald die Arbeit verrichtet ist, soll er in den Stoppzustand wechseln.

# Kara, die Blättersammlerin



## Lösung (2 von 3):

### Welche Übergänge werden benötigt?

Als nächstes werden für **alle möglichen Kombinationen der gewählten Sensoren** einen **Übergang** hinzugefügt.

(Beispiel) Ein möglicher Input durch die zwei Sensoren ist:

- vor Kara steht kein Baum
- und Kara steht nicht auf einem Kleeblatt.

	
×	×
×	✓
✓	×
✓	✓

In dieser Tabelle sind alle möglichen Kombinationen der Inputs aufgelistet, die Kara über die gewählten Sensoren erhalten kann. Ein Kreuz bedeutet, dass der Sensor „nein“ zurück meldet. Ein Häkchen zeigt an, dass der Sensor „ja“ meldet.

# Kara, die Blättersammlerin

## Lösung (2 von 3):

Nun müssen wir uns noch überlegen, wie Kara in jedem der vier Fälle reagieren soll und in welchem Zustand er anschliessend sein muss. Solange die Aufgabe nicht gelöst ist, darf noch nicht in den Stoppzustand gewechselt werden. Sobald aber der Auftrag erledigt ist, muss in den Stoppzustand übergegangen werden.

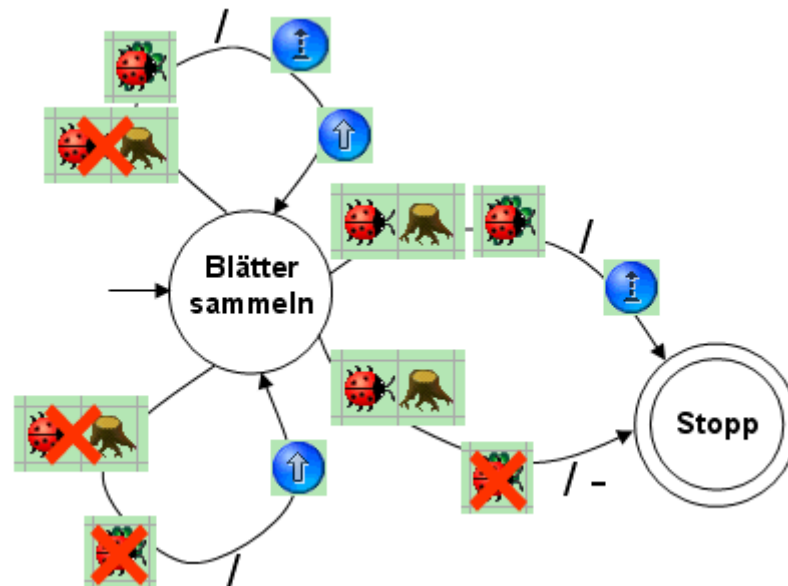
		Karas Aktionen	nächster Zustand
×	×		Blätter sammeln
×	✓	 	Blätter sammeln
✓	×	-	Stopp
✓	✓		Stopp

Diese Tabelle zeigt, welche Aktionen Kara aufgrund der Inputs der Sensoren ausführen soll. Die erste Zeile der Tabelle zeigt: Wenn Kara weder vor einem Baum, noch auf einem Kleeblatt steht, dann soll Kara einen Schritt nach vorne machen.

# Kara, die Blättersammlerin

## Lösung (3 von 3):

Die Zustände aus der vorherigen Abbildung und die Übergänge aus der Tabelle ergeben den folgenden Automaten:



Automat der Kara anleitet, geradeaus bis zum nächsten Baum zu gehen und unterwegs alle Blätter aufzunehmen

# *Kara, die Blättersammlerin*

## Erklärungen

- **Zustand:**

Ein Automat befindet sich immer in einem Zustand und wartet auf neuen Input. Die Zustände stellen das *Gedächtnis des Automaten* dar. Er kann sich zum Beispiel merken, wie viel Geld ein Käufer bereits in den Getränkeautomaten eingeworfen hat.

- **Input:**

Ein Automat reagiert nur auf neuen Input von aussen. Oft erfolgt dieser Input über Sensoren.

- **Aktionen:**

Die Aktionen sind die der Automat gibt, wie auf gewissen Input zu reagieren ist. Zum Beispiel soll der Getränkeautomat die Aktion „Ausgabe einer Cola“ ausführen, wenn der Käufer auf den Knopf Cola drückt. Natürlich dürfen nur jene Aktionen ausgeführt werden, die auf Übergängen aus dem aktuellen Zustand stehen. So muss der Getränkeautomat im Zustand „3.- Kredit“ sein, damit er eine Cola ausspucken darf.

- **Übergang:**

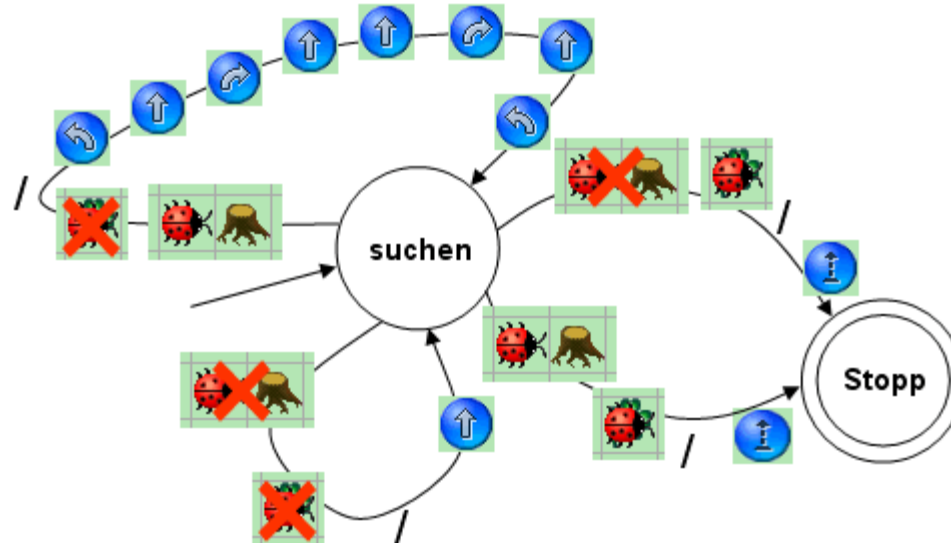
Übergänge sind die Verbindungen zwischen den Zuständen. Auf den Übergängen sind Input und Aktionen vermerkt. Der Input ist die Bedienung, die erfüllt sein muss, damit ein bestimmter Übergang genommen werden darf. Der Automat befindet sich anschliessend im Zustand am Ende des Übergangs.

- **Startzustand:**

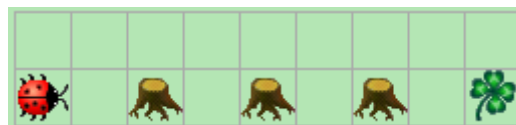
Nach dem Start befindet sich ein Automat in diesem Zustand.

# Kara – Kleeblattsuche im Wald

## Aufgabe:

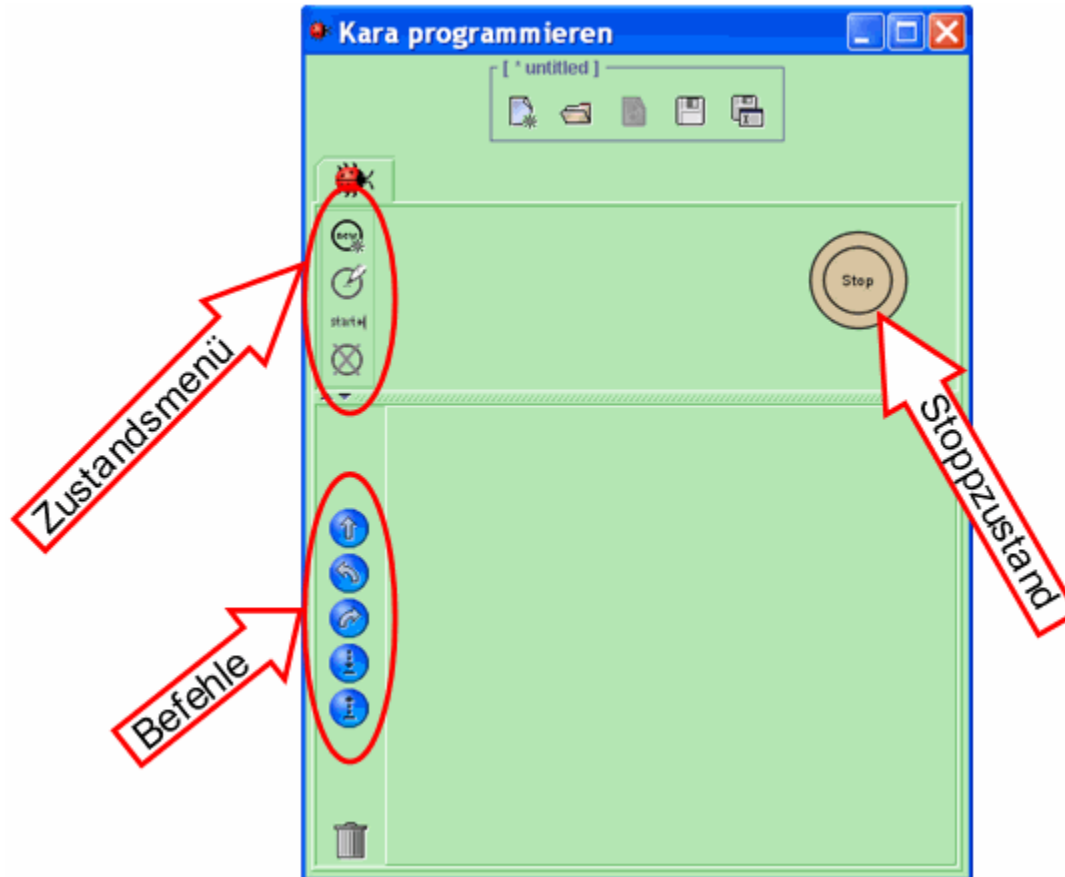


Gegeben ist die unten stehende Wiese. Zeichne den Weg ein, der Kara geht, wenn er durch den oben abgebildeten Automaten gesteuert wird.



# Kara am Computer

Auf den folgenden Seiten wirst du Schritt für Schritt angeleitet, den Automaten aus der vorherigen Aufgabe zu programmieren.








Der Stoppzustand ist standardmässig schon gezeichnet. Im Feld, wo der Stoppzustand ist, kann der Automat gezeichnet werden. Links davon sind alle Operationen, die benötigt werden, um einen Automaten zu zeichnen.

# Kara am Computer

## Zustände einfügen

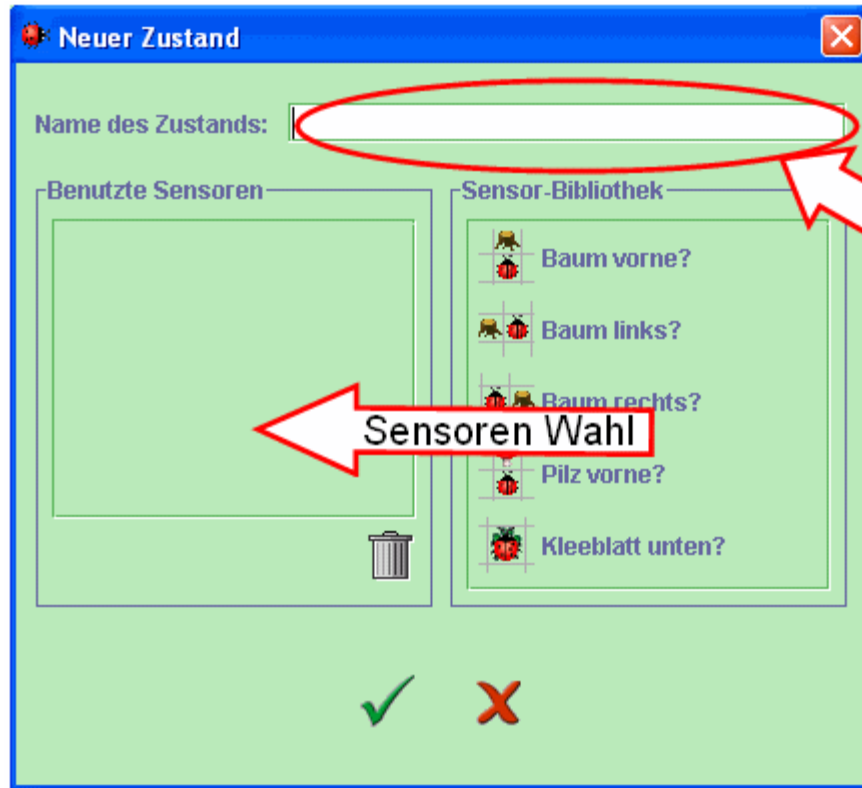
Die folgenden Operationen stehen dir zur Verfügung, um einen Automaten zu zeichnen

	Um einen neuen Zustand zu erhalten, klickst du mit der Maus auf dieses Symbol. Es geht ein Zustandseditor auf, in dem du den Zustand editieren kannst.
	Einen Zustand kann man verändern, in dem man ihn zuerst mit der Maus anklickt und dann mit einem Mausklick auf dieses Symbol den Zustandseditor öffnet.
	Jeder Automat muss einen Startzustand haben. Markiere einen Zustand und klicke mit der Maus auf dieses Symbol, um den Zustand als Startzustand auszuzeichnen.
	Ein Mausklick auf dieses Symbol löscht den zuvor mit der Maus ausgewählten Zustand.
	Einen Übergang zwischen zwei Zuständen erhältst du, wenn du im Ausgangszustand in den äusseren Ring klickst, die Maustaste gedrückt hältst, den Mauszeiger in den Folgezustand ziehst und die Maustaste wieder los lässt.

In dieser Tabelle sind alle möglichen Kombinationen der Inputs aufgelistet, die Kara über die gewählten Sensoren erhalten kann. Ein Kreuz bedeutet, dass der Sensor „nein“ zurück meldet. Ein Häkchen zeigt an, dass der Sensor „ja“ meldet.

# Kara am Computer

Zeichne einen neuen Zustand mit einem linken Mausklick auf das dieses Symbol .



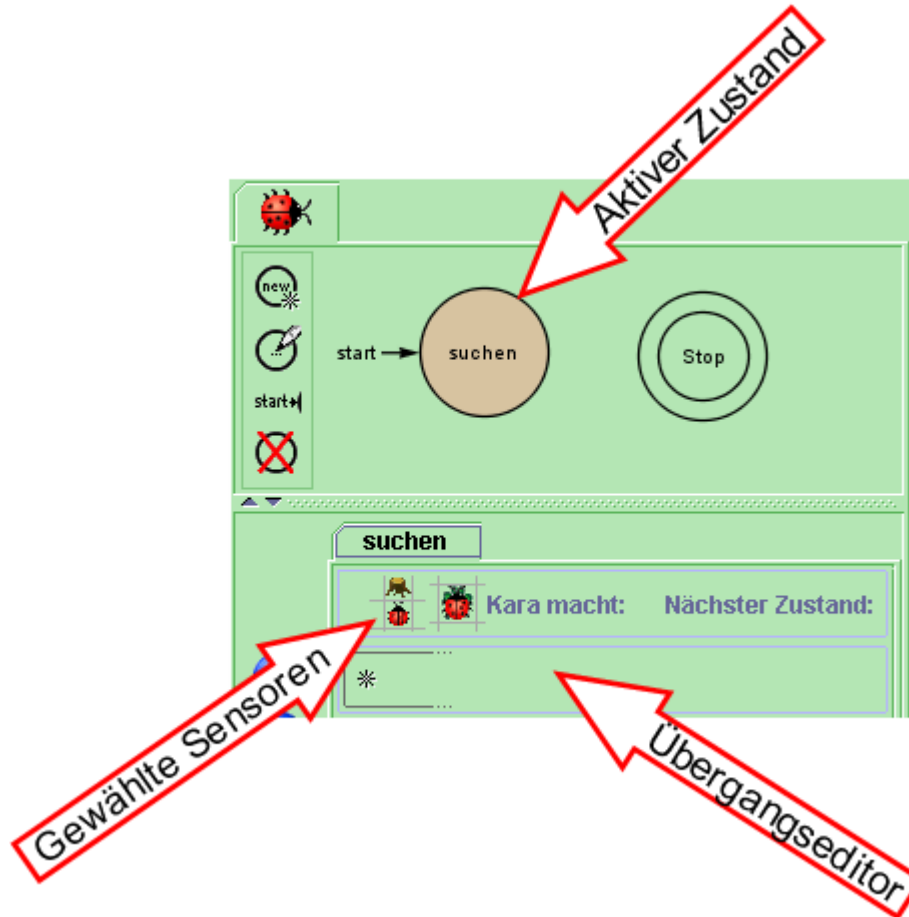
- Gib dem neuen Zustand im Namensfeld einen sprechenden Namen, zum Beispiel „suchen“

- Wähle dann die benötigten Sensoren und ziehe sie mit Drag-and-Drop in den Rahmen unter „Benutzte Sensoren“. Für das Blättersammeln müssen wir wissen, ob Kara vor einem Baum steht, oder ob er auf einem Kleeblatt steht.

Der Zustandseditor. Die gewählten Sensoren erscheinen im Übergangseditor des Zustandes. Der Übergangseditor liegt im Programmeditor unterhalb des Feldes, in dem der Automat gezeichnet wird

# Kara am Computer

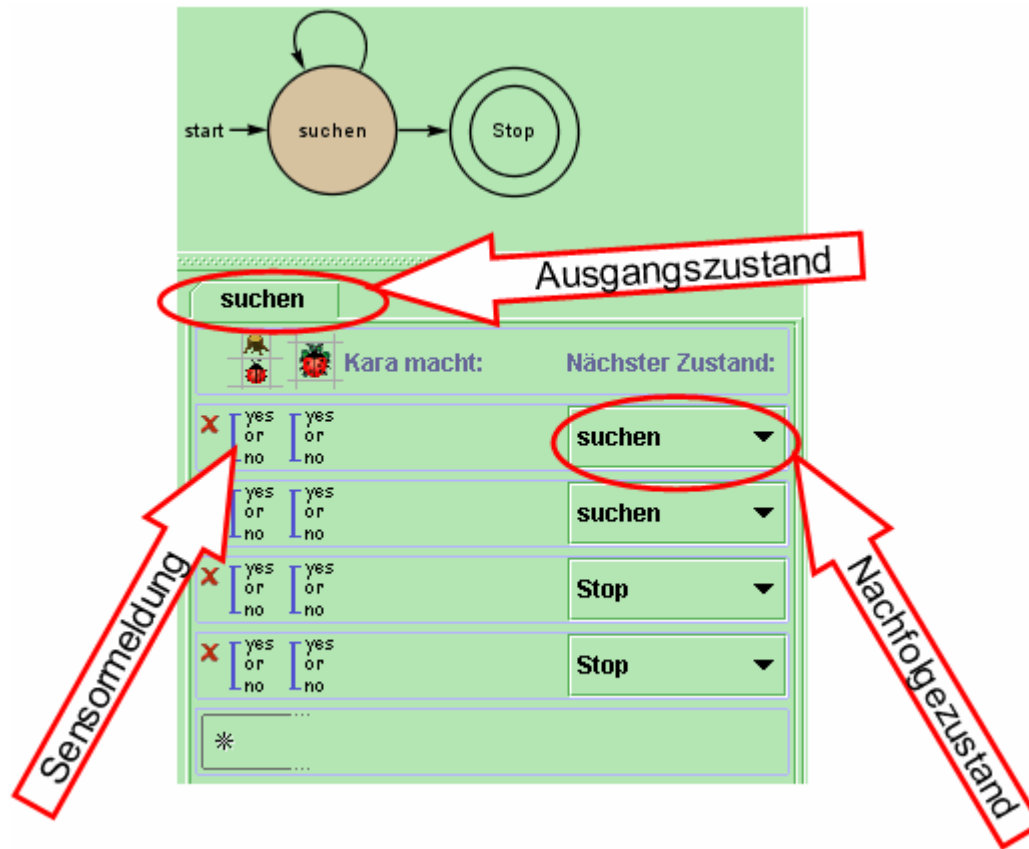
Im Programmmeditor ist nun ein zusätzlicher Zustand zu sehen. Es fehlen aber noch alle Übergänge. Diese werden wir im nächsten Schritt einfügen.



Diese Abbildung zeigt den Programmmeditor mit dem Übergangseditor für den aktiven Zustand. Im Übergangseditor werden die gewählten Sensoren angezeigt.

# Kara am Computer

**Übergänge:** Die Übergänge werden wie folgt gezeichnet: Klicke in den äusseren Rand des Startzustandes und ziehe den Mauszeiger auf den Folgezustand, ohne die Maustaste loszulassen.

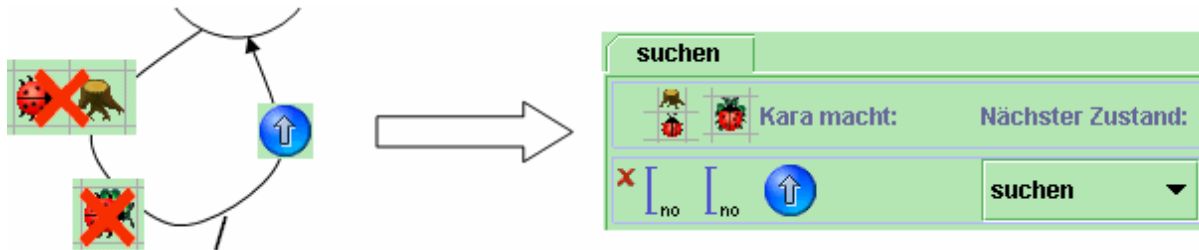


Hier können wir die Inputs (Bedingungen, Meldungen der Sensoren) und Aktionen (Anweisungen an Kara) der einzelnen Übergänge festlegen. Eine Zeile entspricht einem Übergang. Der Übergang geht vom Ausgangszustand in den Nachfolge-zustand. Nachfolgezustände müssen nicht für alle Übergänge dieselben sein.

# Kara am Computer

## Input und Aktionen

Die Beschriftung der Übergänge mit Input und Aktionen machen wir im Übergangseditor. Soll Kara vor einem Baum stehen oder nicht, wenn er den Übergang in der ersten Zeile nimmt? Und soll er im Zustand „suchen“ bleiben? Mit einem Mausklick auf den blauen Balken kann die Belegung auf *yes* oder *no* geändert werden. Wenn es aber keine Rolle spielt, welche Meldung der Sensor gibt, so kann auch *yes or no* beibehalten werden. Anschliessend können die Befehle mit Drag-and-Drop in der gewünschten Reihenfolge rechts der Sensorbelegungen eingefügt werden.

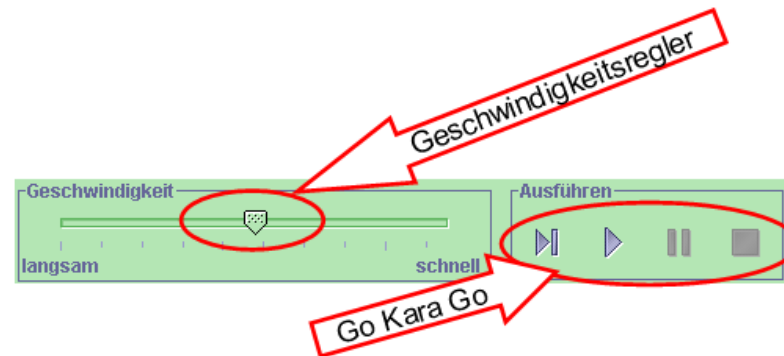


In dieser Abbildung ist das Übertragen eines Übergangs vom Papier ins Kara Programm zu sehen.

# Kara am Computer

## Ausführen des Programmes

Nun ist der Automat fertig und du kannst Kara mit deinem Automaten laufen lassen.



**Aufgabe:** Lasse Kara in den folgenden Welten laufen und teste, ob dein Programm in diesen Situationen richtig läuft.

**Welt 1**



**Welt 2**



**Welt 3**

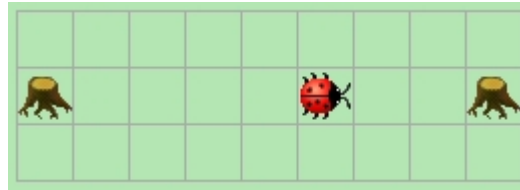




# „Hello World“ von Kara

## Aufgabe:

Kara steht in Blickrichtung zu einem Baum; K. startet; K. geht bis unmittelbar vor den Baum; K. dreht sich um; K. geht bis unmittelbar vor einen Baum; K. dreht sich um; K. geht bis unmittelbar vor einen Baum; K. dreht sich um; usw.

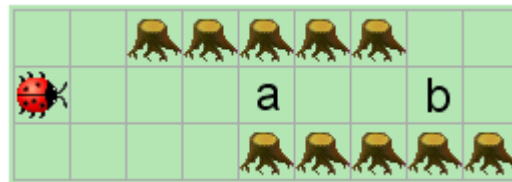


Programmiere Kara so, dass er endlos zwischen zwei Bäumen hin- und herläuft.

# *Kara, der Tunnelsucher*

## **Aufgabe:**

Teil a) Kara sucht den Eingang eines geraden Tunnels (Feld a). Schreibe ein Programm, das ihn auf dem ersten Feld im Tunnelinnern anhalten lässt. Aber Achtung: manche Tunnels haben zunächst eine einseitige Wand, manche links, manche rechts.



Teil b) Kara will den Ausgang des Tunnels finden (Feld b). Dazu muss er zunächst den Tunnel durchqueren. Schreibe ein Programm, das ihn auf dem ersten Feld nach dem Tunnel anhalten lässt - er soll nicht bis zum Ende der Galerie laufen!





# Kara, der Spaziergänger (Lösung)

**Frage:** Aufgrund welcher Eigenschaft können wir Kara durch den Rundgang steuern?

**Antwort:**

Ein Rundgang hat folgende Eigenschaft:

*Jedes Feld in dem Rundgang hat genau zwei freie, unmittelbar benachbarte Felder.*

Gehen wir davon aus, dass er zu Beginn hinter sich ein freies Feld hat, so ist genau eines der Felder vor, links oder rechts von ihm frei. Dieses Feld besucht er als nächstes. Jetzt gilt wieder, dass das Feld hinter ihm frei ist, und wieder kann er ganz einfach bestimmen, wohin er als nächstes läuft.

?

Kara    ?

?

Invariante beim Rundgang

Der Begriff der **Invariante**

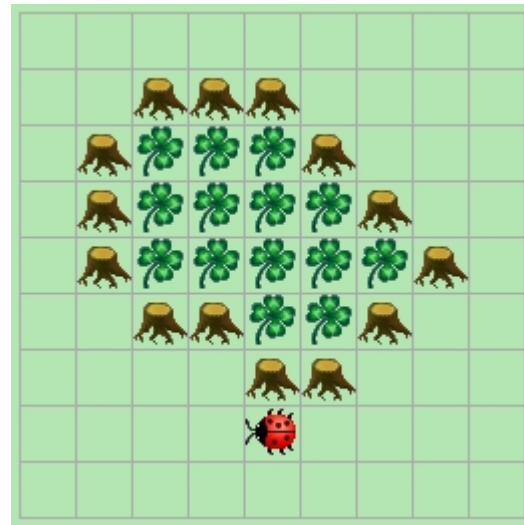
Die Invariante ist eine Aussage, die vor Eintritt in den Rechenschritt eines Programms erfüllt ist, und die nach dem Schritt erneut wahr ist.

Das Programm endet, wenn die **Endebedingung** erfüllt ist.

# Kara, der Kleeblattbewacher!

## Aufgabe:

Kara hat sich einen leckeren Vorrat von Kleeblättern angelegt. Dieser ist von Bäumen umringt. Damit sich niemand an seinen Kleeblättern vergreift, beschliesst Kara, diesen Wald zu patrouillieren, ihn immer wieder abzulaufen. So sieht der „Vorratswald“ aus:



Programmiere Kara so, dass er endlos um diesen „Wald“ läuft! Du kannst selbst entscheiden, ob du ihn links- oder rechtsherum laufen lässt.

Hinweis: Ein Zustand reicht für die Lösung der Aufgabe aus. Versuche, mit möglichst wenig Sensoren auszukommen, und möglichst wenig Übergänge in deinem Zustand zu haben!

# *Kara, der Kleeblattbewacher!*

## **Lösungsstrategie:**

Der Einfachheit halber nehmen wir an, dass Kara beim Start des Programms rechts oder rechts hinten die Wand berührt. Die Fragezeichen bedeuten, dass der „Inhalt“ des entsprechenden Feldes irrelevant ist.

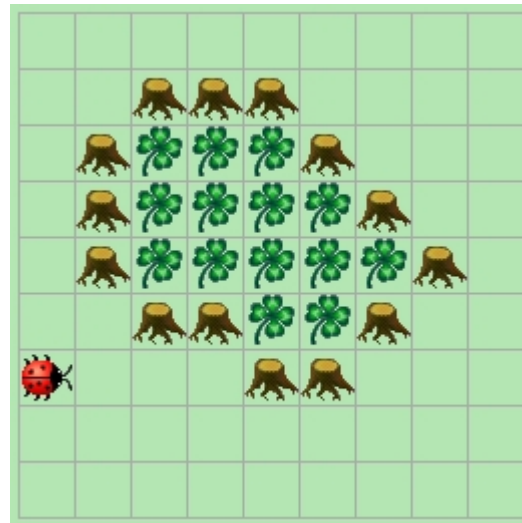
?	?	?		?	?	?
?	Kara	Baum		?	Kara	?
?	?	?		?	?	Baum

Kara muss die Vorbedingung bei jedem Übergang aufrecht erhalten, sonst verliert er den Kontakt zur Wand.

# Kara – Wand entlang

## Aufgabe:

Kara hat sich einen leckeren Vorrat von Kleeblättern angelegt. Dieser ist von Bäumen umringt. Damit sich niemand an seinen Kleeblättern vergreift, beschliesst Kara, diesen Wald zu patrouillieren, ihn immer wieder abzulaufen. So sieht der „Vorratswald“ aus:



**Erweitere Dein Programm so**, dass Kara zuerst geradeaus bis zum nächsten Baum läuft und dann erst mit dem Patrouillieren beginnt.

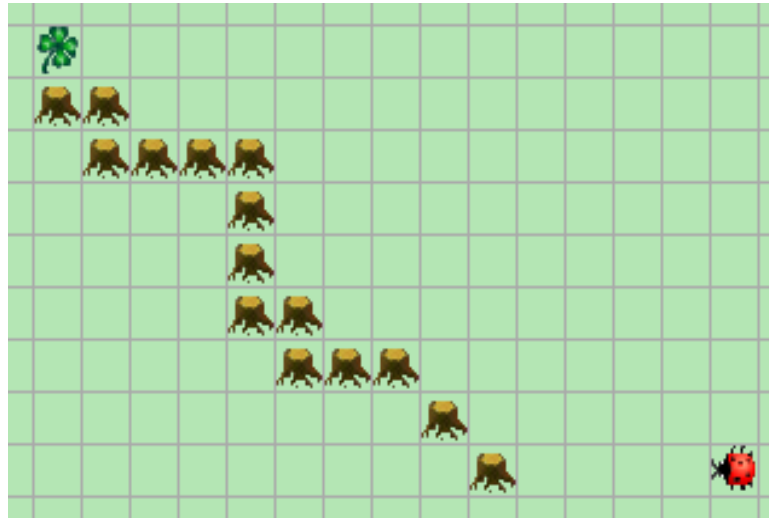
# *Kara und die Baumtreppe*

## **Aufgabe:**

Kara blickt auf den Fuß einer Baumtreppe, die beliebig viele, beliebig hohe und beliebig breite Stufen hat. Die einzelnen Stufen können unterschiedlich hoch und breit sein.

Karas Aufgabe besteht darin, das Kleeblatt aufzuheben.

Das Kleeblatt liegt am Ende auf der obersten Stufe.



Schreibe für diese Aufgabe eine allgemeine, gegliederte Lösung in Zustandskara.

# *Kara und die Baumtreppe*

**Lösung (Zustandsübergangstabelle):**

Zustand	Sensor 1	Sensor 2	Sensor 3	Aktionen	Nächster Zustand

Hinweis: Markiere auch den Startzustand.