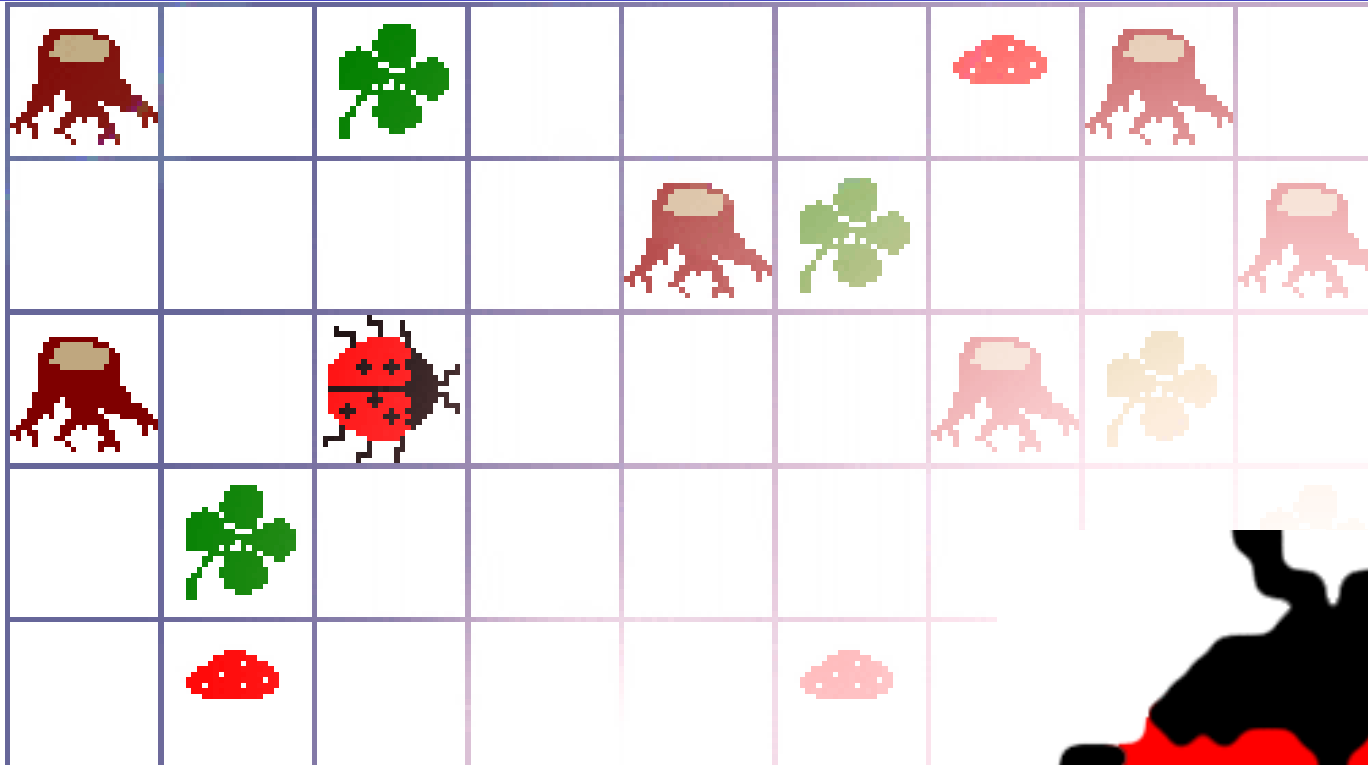


# *Kara, der programmierbare Marienkäfer !*

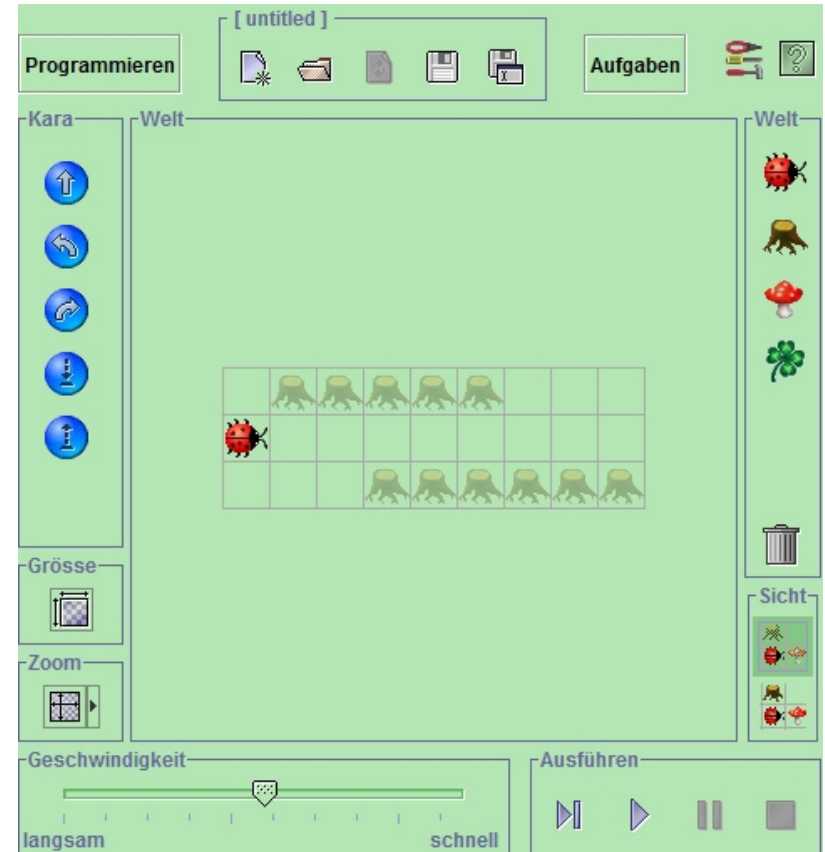


# *In der Welt des Marienkäfers...*

... gibt es:

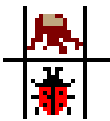
- unbewegliche **Baumstümpfe**,
- **Pilze**, die Kara verschieben und
- **Kleblätter**, die Kara legen und aufnehmen kann

... 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!

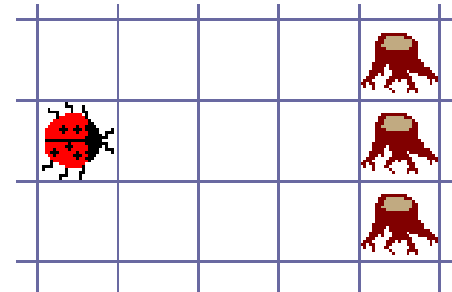
 lege ein Kleeblatt hin!

 nimm ein Kleeblatt auf!

# Kara sucht einen Baumstumpf !

## Die Aufgabe:

bis zum nächsten Baumstumpf  
laufen, dann 180° Drehung!



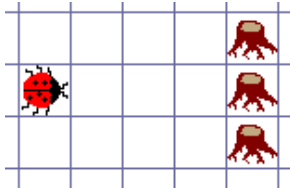
## Das Program:

ein Zustand „suche Baum“,  
ein Sensor „stehe vor  
Baum“



suche Baum		
	Kara executes ...	Go to state:
no		suche Ba...
yes		STOP

# Kara sucht einen Baumstumpf

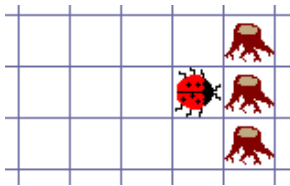


	Kara executes the...	Go to state:
no		suche Ba...
yes		STOP

	Kara executes the...	Go to state:
no		suche Ba...
yes		STOP



■ ■ ■



	Kara executes the...	Go to state:
no		suche Ba...
yes		STOP

	Kara executes the...	Go to state:
no		suche Ba...
yes		STOP

	Kara executes the...	Go to state:
no		suche Ba...
yes		STOP



1. Sensorwerte betrachten, Zeile auswählen
  2. Befehle ausführen
  3. In nächsten Zustand gehen (gelber Pfeil)
- 
1. Sensorwerte betrachten, Zeile auswählen
  2. Befehle ausführen
  3. In nächsten Zustand gehen (gelber Pfeil)

Zeit ↓

# Kara und die Blätter ... (einfach)

The image displays three windows from the 'Kara' programming environment:

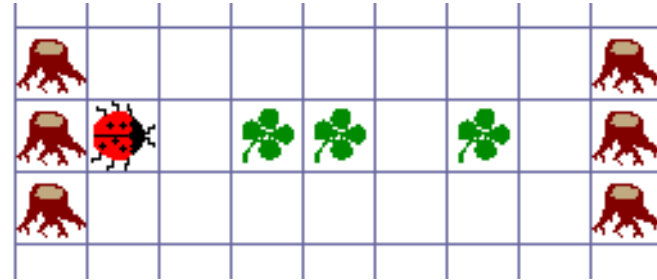
- Aufgaben (Tasks):** Shows the task 'Kara und die Blätter... (einfach)'. The instructions are: 'Schreiben Sie ein Programm, das Kara bis zum nächsten Baum führt. Liegt auf einem Feld ein Blatt, soll Kara es aufnehmen; liegt auf einem Feld kein Blatt, eines hinlegen. Bei dem Baum angekommen ist das Programm zu beenden.' Below the text is a world view with a red bug, green leaves, and a tree.
- Kara, der programmierbare Marienkäfer (Kara, the programmable ladybug):** Shows the 'Welt' (World) view with the same elements as the task window. It includes a toolbar with navigation and action icons, and a 'Sicht' (View) panel.
- Kara programmieren (Program Kara):** Shows the programming interface. It features a flowchart with a 'start' node pointing to a 'SucheBaum' node, which has a self-loop and an arrow pointing to a 'Stop' node. Below the flowchart is a table for defining the 'SucheBaum' state transitions.

	Kara macht:	Nächster Zustand:
	<input checked="" type="checkbox"/>  no  no	SucheBaum
	<input checked="" type="checkbox"/>  yes  no	SucheBaum
	<input checked="" type="checkbox"/>  no  yes	Stop
	<input checked="" type="checkbox"/>  yes  yes	Stop

# Kara, der Blättersammler !

## Die Aufgabe:








alle Blätter bis zum nächsten Baumstumpf aufnehmen!



## Das Programm:

ein Zustand „collect leaves“, benötigt zwei Sensoren

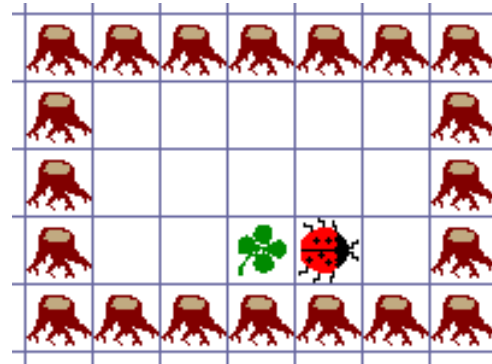


collect leaves		Kara executes ...	Go to state:
 			
no	no		collect le...
no	yes	 	collect le...
yes	no		STOP
yes	yes		STOP

# Kara, der Wächter (1)

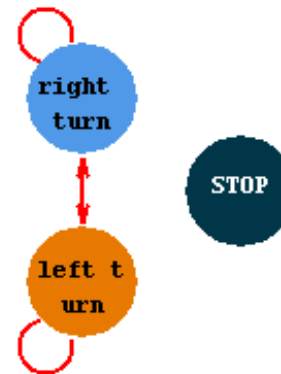
## Die Aufgabe:

Raum linksherum bis Kleeblatt,  
dann rechtsherum bis Kleeblatt,  
dann linksherum... laufen



## Das Programm:

zwei Zustände:  
einer für Linksdrehung,  
einer für Rechtsdrehung












right turn		Kara executes these ...	Go to state:
 			right turn
 	 		right turn
 	  		left turn

left turn		Kara executes these ...	Go to state:
 			left turn
 	 		left turn
 	  		right turn

# Kara, der Wächter (2)

**Das Programm:** ein Zustand !

guard			Kara executes t...	Go to state:
 no	 yes or no	 no		guard
 yes or no	 yes or no	 yes	  	guard
 yes	 no	 no	 	guard
 yes	 yes	 no	 	guard

**Welche Lösung ist „besser“ ?**

- Lesbarkeit des Programms ?
- Anzahl Befehle: Weniger = besser ?
- Anzahl Zustände: Weniger = besser?

⇒ **Eine Frage des persönlichen Programmierstils!**

