

Werkzeuge für den Informatikunterricht

---- endlicher Automat Kara ----

1. Kurzvorstellung

Der endliche Automat Kara bringt den SuS die Grundlagen des Programmierens spielerisch bei. Hierfür wurde eine schülergerechte 2-Dimensionale Oberfläche geschaffen auf der sich der steuerbare Marienkäfer „Kara“ frei bewegen kann. Er kann mittels Programmierung mit dieser Oberfläche auf verschiedene Weisen interagieren und diese auch verändern.

2. Einordnung in die Lehrpläne

Lehrplan OS: In der Oberschule kann dieses Werkzeug in der 8. Klasse im Lernbereich 2 (Informationen verarbeiten: Modell –Algorithmus–Lösung), als Programmierumgebung verwendet werden.

Lehrplan Gy: Auf dem Gymnasium kann dieses Werkzeug in der 9./10. Klasse im Lernbereich 4 (Algorithmen und Programme), als Programmierumgebung verwendet werden.

Lehrplan BGy und Lehrplan BS: Hier bietet sich dieses Werkzeug nicht an, da es in seiner Komplexität und im Anwendungsumfang doch sehr begrenzt ist und somit für weiterführende Programmierung nicht geeignet ist.

3. Lernziele

Kognitive Lernziele:

Die Schüler-innen kennen die Teilschritte des Problemlöseprozesses.

Die Schüler-innen übertragen ihr Wissen über den endlichen Automaten auf das Automatenmodell in Kara.

Psychomotorische Lernziele:

Die Schüler-innen sind in der Lage ausgewählte Algorithmen in einer Programmumgebung zu implementieren.

Die Schüler-innen beherrschen das Prinzip des „Drag and Drop“

Affektive Lernziele:

Die Schüler-innen sind fähig ihre Resultate kritisch zu beurteilen.

4. Kompetenzentwicklung

Fachkompetenz:

Die Schüler-innen können ein Problem mithilfe eines algorithmischen Ansatzes lösen.

Lern-/Methodenkompetenz:

Die Schüler-innen sind in der Lage Probleme informatischer Natur zu analysieren.

Die Schüler-innen können Lösungsentwürfe für Probleme informatischer Natur erstellen.

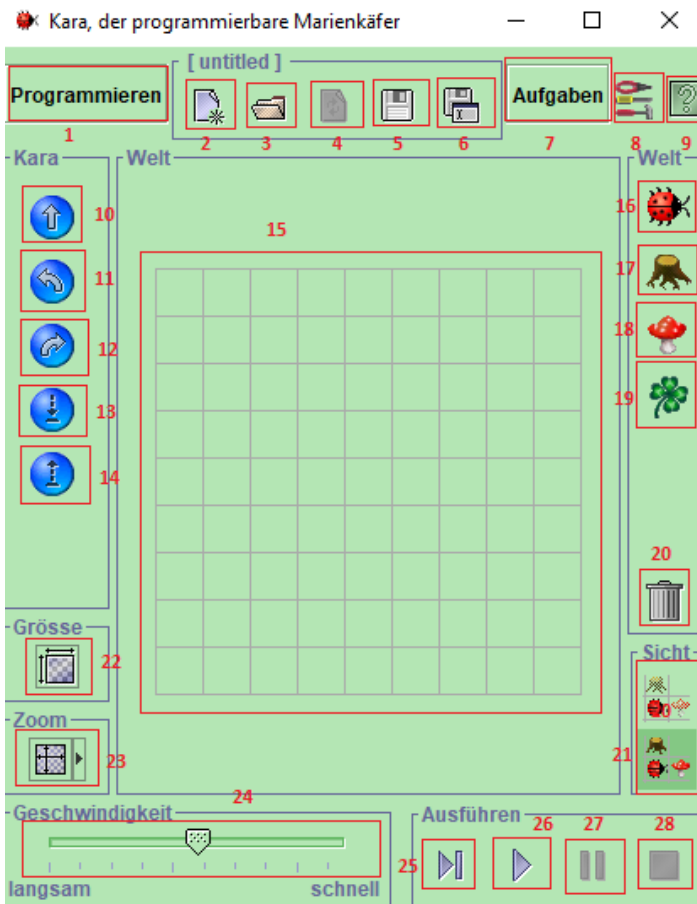
Sozialkompetenz:

Die Schüler-innen sind in der Lage ihren Lösungsansatz vorzustellen und Mitschülern zu erklären.

Selbstkompetenz:

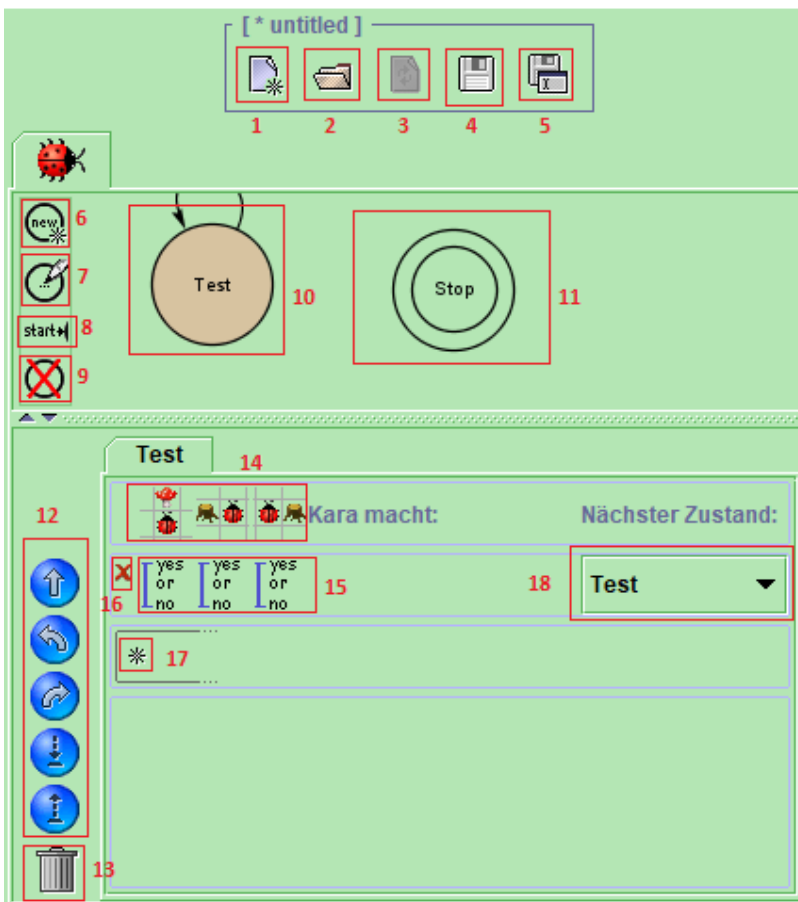
Die Schüler-innen sind in der Lage ihr Ergebnis selbständig auf Korrektheit zu überprüfen.

5. Prinzipieller Aufbau



Kara Weltenfenster (W)

- 1 Öffnen des Programmfensters
- 2 Neue Welt erstellen
- 3 Weltdatei öffnen
- 4 Welt zurücksetzen
- 5 Speichern
- 6 Speichern unter
- 7 Aufgabenpool öffnen
- 8 Einstellungen
- 9 Bedienhilfe
- 10 Kara läuft nach vorne
- 11 Kara dreht sich nach links
- 12 Kara dreht sich nach rechts
- 13 Kara legt ein Kleeblatt hin
- 14 Kara hebt ein Kleeblatt auf
- 15 Welt
- 16 Kara
- 17 Baumstumpf
- 18 Pilz
- 19 Kleeblatt
- 20 Mülltonne
- 21 Sichtweisen
- 22 Größe der Welt einstellen
- 23 Zoom
- 24 Geschwindigkeit des Programmes
- 25 Programm Schritt für Schritt ausführen
- 26 Programm ausführen
- 27 Programm pausieren
- 28 Programm stoppen



- 1 Neues Programm erstellen
- 2 Programmdatei öffnen
- 3 Programmdatei wiederherstellen
- 4 Speichern
- 5 Speichern unter
- 6 Neuen Zustand erstellen
- 7 Zustand bearbeiten
- 8 Startzustand festlegen
- 9 Zustand löschen
- 10 Zustand
- 11 Stopzustand
- 12 Aktionen
- 13 Mülleimer
- 14 gewählte Zustände
- 15 Bedingungen
- 16 Übergang löschen
- 17 Übergang erstellen
- 18 Nächster Zustand

6. Handhabung

Vorbereitungen:

- Programm starten und Programmfenster mit W1 öffnen
- Weltdatei und Programmdatei separat speichern

Welt erstellen:

- Größe der Welt einstellen über W22
- Alle benötigten Objekte(W16-W19) mittels Drag and Drop an ihre Position ziehen
- Wichtig dabei ist, dass Kara unbedingt in der Welt existieren muss
- Kara mittels W10-W14 an die richtige Position und Richtung bewegen

Programm erstellen:

- Alle benötigten Zustände mithilfe von P6 erstellen(Name wählen und benötigte Sensoren reinziehen)
- Bei jedem Zustand alle Übergänge erstellen(P17) und die Bedingungen für diese festlegen(P15)
- In die Übergänge die Aktionen(P12) reinziehen, welche benötigt werden
- Für jeden Übergang den nächsten Zustand wählen(P18)
- Startzustand festlegen(P8)

Lösung überprüfen:

- Das Programm mithilfe von W25-W28 starten, pausieren und stoppen
- Überprüfen, ob die Aufgabenstellung erfüllt wurde
- Falls nicht, einfach die Welt wiederherstellen(W4) und mit angepassten Programm neu versuchen

7. Screencast

Link und QR-Code zum Screencast



<https://youtu.be/H6IbD84MAmI>

8. Aufgaben

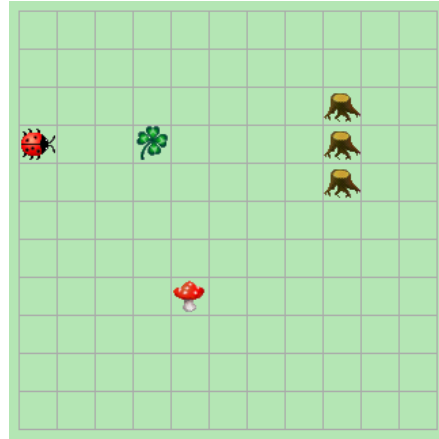
Aufgabe 1:

Mache dich mit den Möglichkeiten von Kara vertraut, indem du die Teilaufgaben löst.

- a) Erstelle hierfür eine Welt genau nach dem Bild. Dazu stellst du zuerst über



Größe die Länge und Breite jeweils auf 11. Nun ziehst du alle Objekte von der rechten Seite auf ihren Platz. Falls du etwas falsch platzierst, ziehe es einfach in den Mülleimer um es zu löschen.

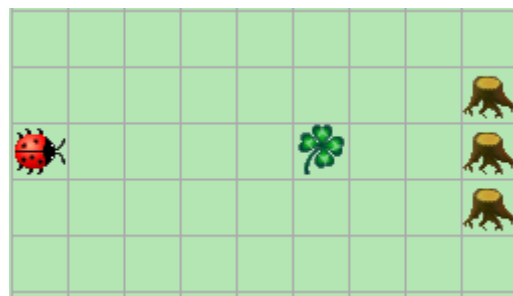


- b) Steuere Kara nun bis zum Kleeblatt und lass es ihn aufheben. Danach lässt du ihn bis zu den Baumstümpfen laufen. Versuche noch einen weiteren Schritt zu machen. Schreibe in deinen Hefter was passiert.
- c) Lege jetzt an einem beliebigen Feld ein Kleeblatt ab und steuere Kara so, dass er direkt vor dem Pilz steht und ihn anschaut. Mache nun noch einen Schritt nach vorn. Schreibe in deinen Hefter was passiert.

Aufgabe 2:

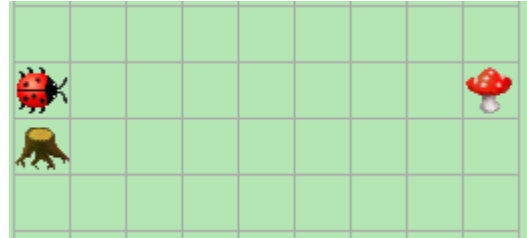
Kara steht auf einer Wiese und läuft gerade bis zum Wald. Wenn er unterwegs ein Kleeblatt findet, soll er es aufheben.

- a) Welche Sensoren müsst ihr für die Erfüllung der Aufgaben aktivieren?
- b) Nennt den Zustand „Sammle“ und füge die Sensoren hinzu.
- c) Überlege, welche Kombinationen der Sensoren es geben kann und erstelle für jede einen Übergang.
- d) Erstelle die Welt nach dem Bild und schreibe das Programm in Kara!



Aufgabe 3:

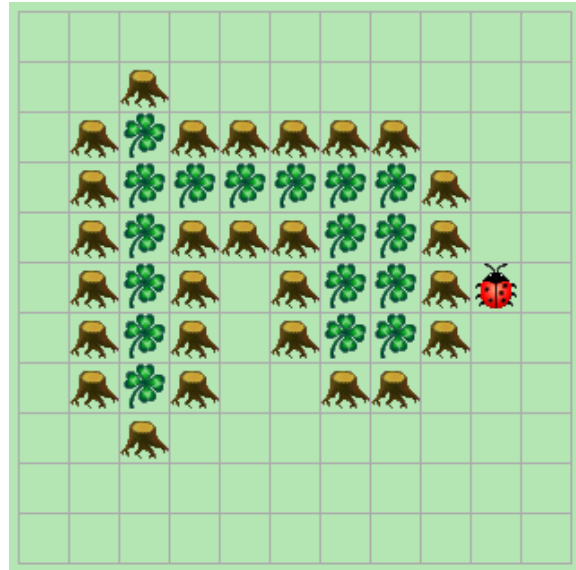
Kara soll bis zum Pilz laufen, dort umdrehen und auf der unteren Spur zum Baum zurück laufen. Löse alle Aufgaben mit einem Zustand



- a) Erstelle die Welt nach dem Bild und löse Kara sein Problem.
- b) Wandel dein Programm nun so um, dass Kara vor dem Pilz und vor dem Baumstumpf ein Kleeblatt ablegt.
- c) Der Orientierungssinn von Kara ist leider nicht der beste. Damit er wieder nach Hause findet, möchte er nach seinem Spaziergang an seinem Startplatz ankommen. Verändere dein Programm so, dass Kara am Ende genau wie vor dem starten des Programms steht.

Aufgabe 4

Kara wurde als Wächter für den Kleeblattvorrat eingeteilt. Hierfür soll er das Lager immer entgegen dem Uhrzeigersinn umrunden, ohne dass es einen Stopzustand gibt. Damit wirklich niemand etwas stehlen kann, muss er sich immer so nah wie möglich an den Baumstümpfen aufhalten

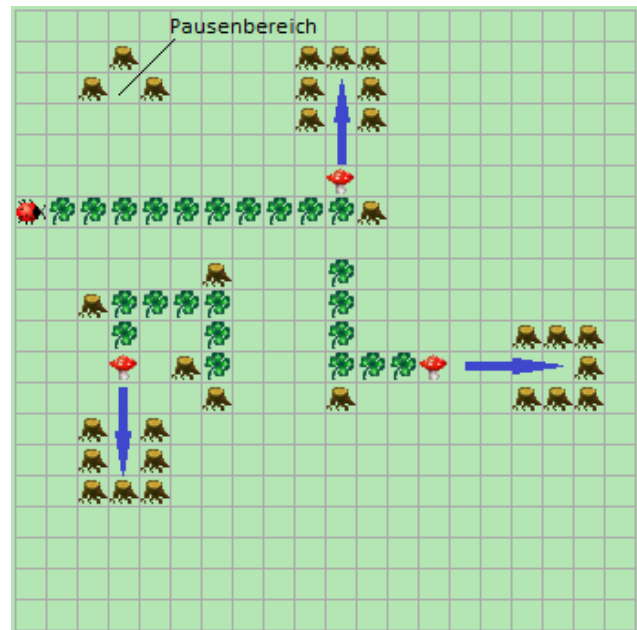


- a) Erstelle die Welt wie auf dem Bild.
- b) Ermittle welche Sensoren du benötigst und füge sie dem Zustand Wächter hinzu.
- c) Schreibe das Programm!

Aufgabe 5

Kara soll das Lager aufräumen. Dafür muss er die Pilze in die jeweils für sie vorgesehenen Bereiche schieben und alle Kleeblätter aufsammeln. Am Ende seiner Arbeit soll er sich oben im Pausenbereich ausruhen.

- Baue die Welt nach dem Bild nach.
- Überlege welche Sensoren du alles brauchst und ziehe sie in den neu erstellten Zustand „Lagerarbeiter“.
- Schreibe das Programm womit Kara seine Arbeit erledigen kann.



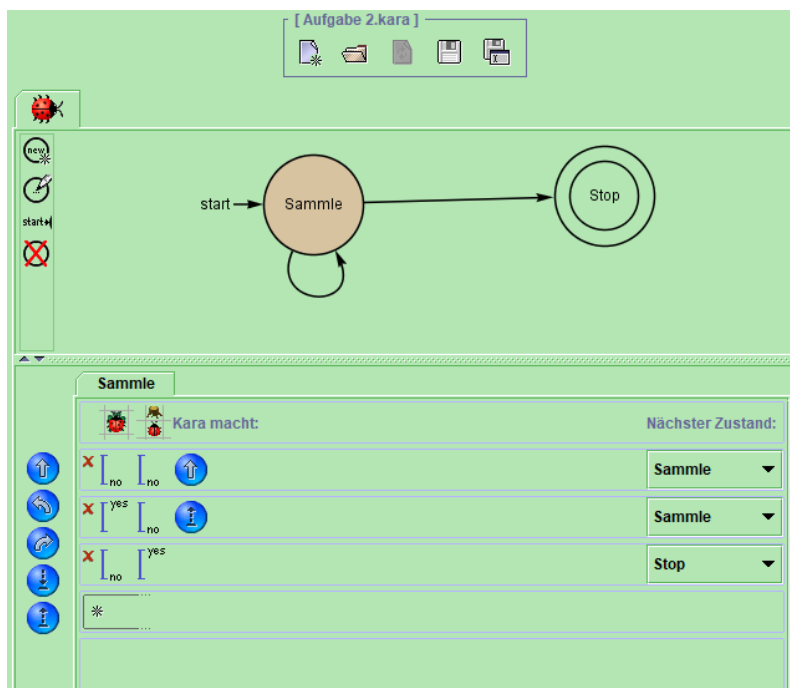
9. Lösungen für Aufgaben

Lösung Aufgabe 1:

- a) –
- b) Kara kann sich nicht durch einen Baumstumpf bewegen
- c) Kara kann einen Pilz verschieben

Lösung Aufgabe 2:

- a) Sensoren: Kleeblatt unter mir, Baumstumpf vor mir



- b) –
- c) –

Lösung Aufgabe 3:

Screenshot a) shows a state machine editor interface. The top part displays a state transition diagram with two states: 'Laufen' (start state, represented by a circle with a self-loop) and 'Stop' (final state, represented by a double circle). A transition arrow points from 'Laufen' to 'Stop'. Below the diagram is a table for the 'Laufen' state, titled 'Kara macht:'. The table has four rows, each representing a possible action (yes/no) and the resulting next state. The 'Nächster Zustand:' column shows the next state for each row.

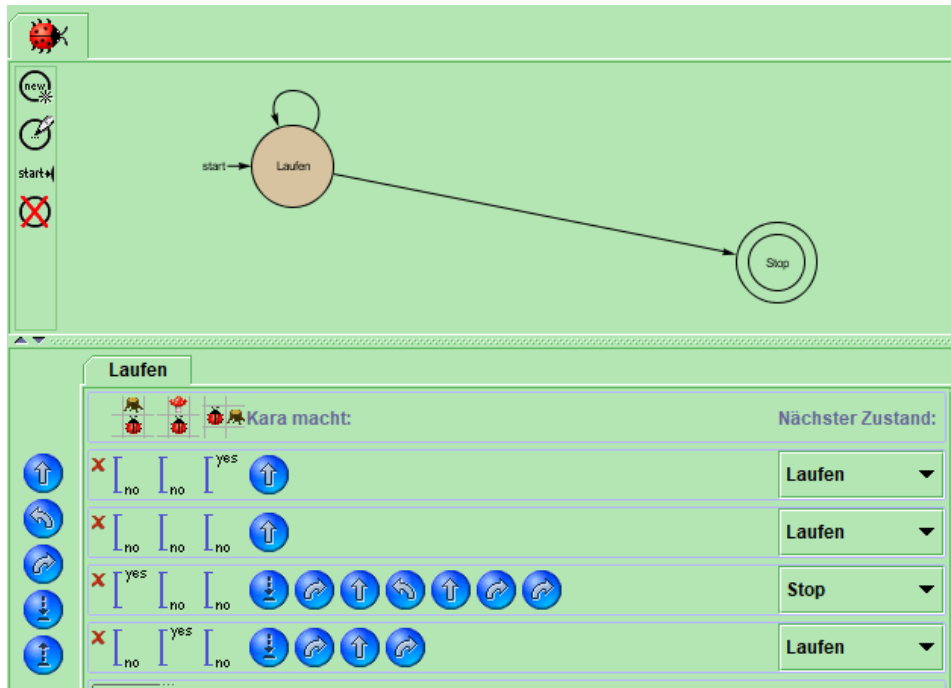
Kara macht:			Nächster Zustand:
no	no	yes	Laufen
no	no	no	Laufen
yes	no	no	Stop
no	yes	no	Laufen

a)

Screenshot b) shows the same state machine editor interface as in a). The state transition diagram is identical. The table for the 'Laufen' state is also identical, but the 'Nächster Zustand:' column for the third row (yes/no/no) is 'Stop', and for the fourth row (no/yes/no) is 'Laufen'.

Kara macht:			Nächster Zustand:
no	no	yes	Laufen
no	no	no	Laufen
yes	no	no	Stop
no	yes	no	Laufen

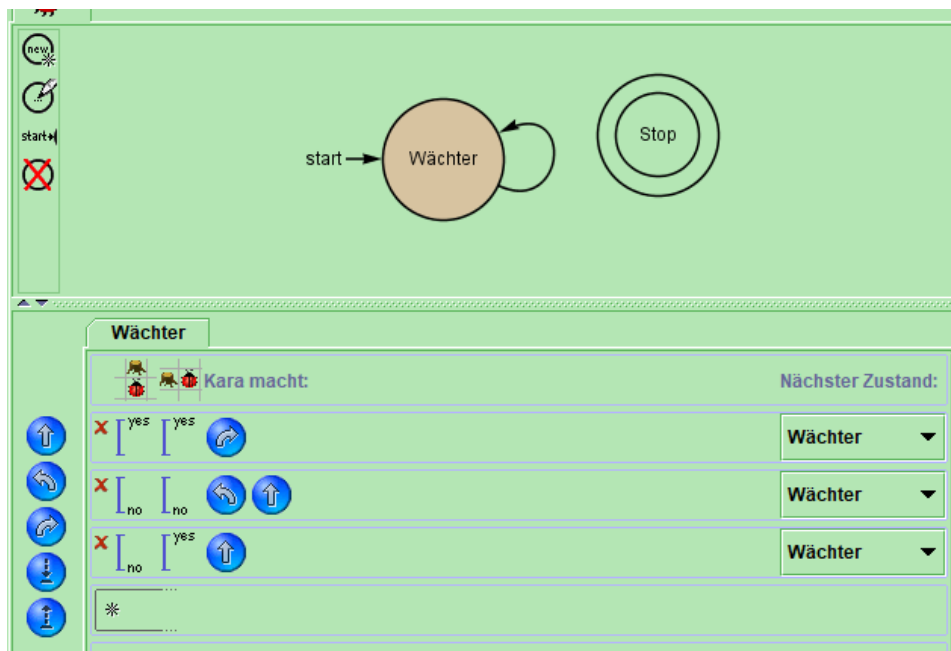
b)



c)

Lösung Aufgabe 4:

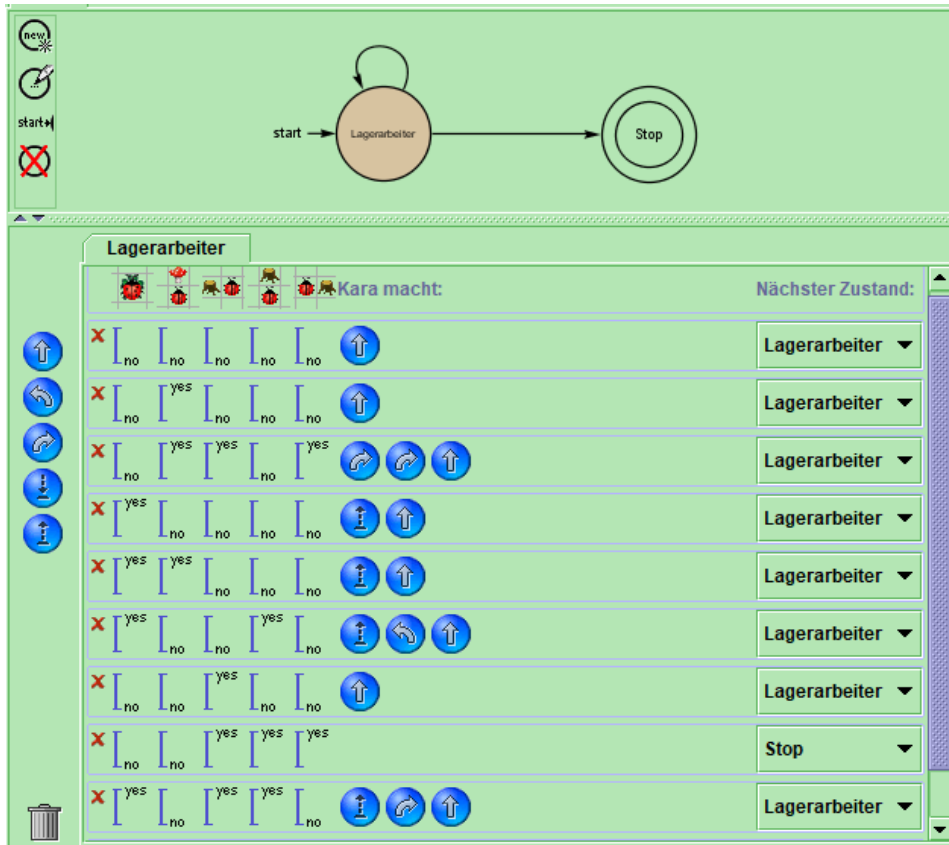
- a) -
- b) -



c)

Lösung Aufgabe 5:

- a) –
b) –



c)

10. Quellenangabe

Sächsisches Staatsministerium für Kultus (2019): Lehrplan Gymnasium. Informatik:

http://lpdb.schule-sachsen.de/lpdb/web/downloads/2345_lp_gy_informatik_2019.pdf?v2 (20.03.2021)

Sächsisches Staatsministerium für Kultus (2019): Lehrplan Oberschule. Informatik:

http://lpdb.schule-sachsen.de/lpdb/web/downloads/42_lp_os_informatik_2019 (20.03.2021)

Swiss Educ: Programm Kara

<https://www.swisseduc.ch/informatik/karatojava/kara/> (20.03.2021)