

# Werkzeuge für den Informatikunterricht

## Robot Karol

### 1 Kurzvorstellung

Zur Einführung in die Algorithmik und zum Erlernen des Programmierens für Schülerinnen und Schüler dient die Programmierumgebung **Robot Karol**. Die Programmiersprache von Robot Karol dient dem Konzept der "Mini-Languages". Diese Programmiersprachen haben die Eigenschaft, dass Sie über einen übersichtlichen beziehungsweise kleinen Sprachumfang verfügen. Dadurch wird der Einstieg in die Algorithmik erleichtert. Zusätzlich wird das Lernen dadurch unterstützt, da die Programmiersprache auf der deutschen Sprache basiert und daher Sachverhalte für Schülerinnen und Schüler besser dargestellt werden können. Richard E. Pattis veröffentlichte zum ersten mal das Prinzip von Karel the Robot, welches Robot Karol zugrunde liegt. Dieser Roboter lebt in einer sogenannten Bildschirmwelt. Zur Programmierung steht dafür ein Editor bereit. Die dadurch entstehenden Auswirkungen können die Schülerinnen und Schüler direkt in der Grafikanzeige sehen. [1]

### 2 Einordnung in die Lehrpläne

#### 2.1 Oberschule

In der Oberschule bin ich der Meinung, dass man Robot Karol an zwei Stellen im Unterricht verwenden kann.

- Klassenstufe 7 Lernbereich 2

| Lernbereich 2: Computer nutzen und anwenden: Objekte – Attribute – Methoden  | 12 Ustd.   |
|--|--|
| Kennen grundlegender Datenstrukturen in einer ausgewählten Anwendung<br>- Begriffe: Objekt, Attribut, Attributwert, Methode<br>- Zuordnung von konkreten Objekten zum Modell Objekt – Attribut – Attributwert<br><br>- eine einfache Darstellungsform<br>Beherrschen der Arbeit mit konkreten Objekten beim Lösen typischer Aufgaben mit der gewählten Anwendung<br><br>Übertragen der Kenntnisse zur Objektorientierung auf<br>- Lösen eines einfachen Problems<br>- Arbeiten mit einem anderen Programm der gleichen Anwendung | Unterschied zwischen Methode (Formatieren) und Aktion (Kopieren, Einfügen) beachten<br>Objekte der Tabellenkalkulation, grafische Objekte<br>→ MA HS, Kl. 7, LB 2<br>→ MA RS, Kl. 7, LB 1<br>→ KU, Kl. 6, LB 1<br>→ KU, Kl. 7, LB 1<br><br>UML-Notation (Unified Modeling Language)<br>Protokoll, Folie, Plakat, Zeichnung, Berechnung<br>Dokumentenrecherchen, Analysen von Statistiken<br><br>gleiche Aufgabenstellung mit anderer Software<br><br>Urheberrecht<br>⇒ Medienbildung |

In diesem Lernbereich werden die Begriffe von Objekt, Attribut, Attributwert und Methode eingeführt. Hierfür eignet sich Robot Karl samt mit seinen Eigenschaften, um diese Konzepte darzustellen. [2]

- Klassenstufe 8 Lernbereich 2

| Lernbereich 2: Informationen verarbeiten: Modell – Algorithmus – Lösung  | 17 Ustd.  |
|--|---|
| Kennen grundlegender Programmstrukturen<br>- Begriff: Algorithmus (Endlichkeit, Eindeutigkeit, Ausführbarkeit, Allgemeingültigkeit)<br><br>- Programmstrukturen: Folge, Wiederholung, Verzweigung<br>- eine Darstellungsform<br>Beherrschen der Umsetzung des Modells an einfachen Beispielen<br><br>Kennen des Problemlöseprozesses<br>- Problemanalyse, Lösungsentwurf, Umsetzung, Test, Dokumentation<br><br>- Lösen eines einfachen Problems unter Nutzung der Programmstrukturen<br><br>Übertragen der Kenntnisse zum Problemlöseprozess auf<br>- selbstständiges Lösen einfacher Probleme<br><br>- kritische Bewertung der Resultate | Auswahl schülergerechter Software<br>Endlichkeit der Beschreibung und der Ausführung beachten<br>Zusammenarbeit mit MA<br><br>Ausgangspunkt: umgangssprachliche Handlungsbeschreibung<br>Struktogramm, Ablaufplan, Zustandsdiagramm<br>Teilschritte aus Problemen herauslösen:<br>Zeichnen einer Figur, Vertauschen von zwei Elementen, Ansteuern eines Kanals<br>⇒ Methodenkompetenz<br>Entwurf: auch Modell/Algorithmus<br>Umsetzung: auch Implementierung<br>Test: auch Kritik und Korrektur<br><br>Sortieren, einfache Steuerung, einfache Makros, Erzeugen von Grafiken<br>Partnerarbeit<br><br>Ampelsteuerung, einfache Automaten, Aufgaben in einfachen grafischen Programmierumgebungen<br>⇒ Werteorientierung: Handlungs- und Urteilsfähigkeit |

Zum einen wird in diesem Lernbereich "Kennen grundlegender Programmstrukturen" behandelt. Um den Algorithmusbegriff und die Programmstrukturen einzuführen, würde sich Robot Karol eignen, da man alle Strukturen anschaulich darstellen kann. Zudem sollen die Schülerinnen und Schüler Problemlöseprozesse kennenlernen und diese anhand von Programmstrukturen lösen. Dies kann anschaulich mit Robot Karol vollzogen werden.[2]

## 2.2 Gymnasium

- Klassenstufe 9 Lernbereich 4

| Lernbereich 4: Algorithmen und Programme       |  | 17 Ustd. |
|--|--|----------|
| Kennen des Algorithmusbegriffes                | → KI, 8, LB 2<br>→ MA, KI, 8, LBW 1  |          |
| - Eigenschaften                                | Erkennen von algorithmischen Strukturen in Informatiksystemen  |          |
| - Darstellungsformen                           | verbale und grafische Beschreibung, Programm   |          |
| - Grenzen der Algorithmierbarkeit              | Beispiele aus der Erfahrungswelt der Schüler<br>Optimierung des Stundenplanes, Computer als Schachspieler                            |          |
| Kennen der Grundlagen der Programmierung       | → MA, KI, 9, LB 5  |          |
| - einfache Datentypen                          |  |          |
| - algorithmische Grundstrukturen               |  |          |
| - Sequenz                                      |  |          |
| - Selektion                                    |  |          |
| - Zyklus                                       |  |          |
| Einsicht gewinnen in die Modularisierung       | strukturiertes Denken<br>Bedeutung für die Arbeit im Team<br>⇒ Arbeitsorganisation   |          |
| Anwenden der Phasen des Problemlöseprozesses   | einfache Probleme<br>Auswählen adäquater digitaler Werkzeuge zur Lösung des Problems<br>⇒ Problemlösestrategien<br>→ P, KI, 9 und 10 |          |
| - Problemanalyse                               |  |          |
| - Modellierung der Problemlösung               |  |          |
| - Implementierung in einer Programmierumgebung |  |          |
| - Test und Fehlerbehebung                      |  |          |

Allgemein am Gymnasium gibt es weniger Möglichkeiten für den Einsatz von Robot Karol. Der Hauptgrund liegt darin, da, nicht wie in der Oberschule, sich nicht tiefgründig mit dem Konzept Objekt-Attribut-Attributwert beschäftigt wird. Jedoch gibt es, meiner Meinung nach, eine Stelle im Lehrplan, wo man es einsetzen könnte. So wäre es möglich, es als Einstieg in den Lernbereich 4 zu nehmen. Aber für mehr wäre es nicht zielführend, da Begriffe wie Datentyp usw. nicht mehr mit Robot Karol zu behandeln ist. Zudem ist Robot Karol für eine Klasse 9 ungeeignet ist, da es sich eher für kleinere Klasse eignet.[3]

## 2.3 Berufliches Gymnasium

- Klassenstufe 12/13 Lernbereich 3

| Lernbereich 3: Algorithmen und Programme   |   | 24 Ustd. |
|--|---|----------|
| Kennen von Programmiersprachen als Schnittstelle zwischen Mensch und Maschine      | historische Entwicklung   |          |
| - Grundbegriffe  | Syntax, Semantik<br>Compiler, Interpreter<br>Imperative und deklarative Sprachen<br>→ OS INF, KI, 9, LB 2 |          |
| - Klassifizierung von Programmiersprachen  | Programmablaufplan, Struktogramm<br>→ OS INF, KI, 10, LB1   |          |
| Beherrschen ausgewählter Problemlösestrategien in einer höheren Programmiersprache | Folge, Sequenz<br>Selektion, Entscheidung<br>Zyklus, Wiederholung   |          |
| - Visualisierung von Programmstrukturen  |   |          |
| - einfache und strukturierte Datentypen  |   |          |
| - Grundstrukturen  |   |          |
| - Modularisierung  |   |          |
| - ausgewählte Algorithmen und Verfahren  | Euklidischer Algorithmus, Such- und Sortieralgorithmen<br>Iteration, Rekursion<br>Formulare               |          |
| Anwenden von Komponenten der objektorientierten Programmierung                     |   |          |
| Sich positionieren zu Grenzen der Algorithmierbarkeit                              |   |          |
| - Endlichkeit  | numerische Grenze, zeitliche Grenze<br>Grenze der Algorithmierbarkeit<br>Grenze der Abbildungsmöglichkeit |          |
| - ethische Fragen der Nutzung von IT-Systemen                                      | Grenze der Zulässigkeit für die Anwendung von IT-Systemen<br>Grenze der Zuverlässigkeit von Ergebnisse    |          |

Rein theoretisch könnte man Robot Karol in diesem Lernbereich verwenden, da die Grundbegriffe/Grundstrukturen "von Programmiersprachen" behandelt werden soll. Aber für viel mehr würde es sich nicht eignen, da Robot Karol schnell an seine Grenzen kommen würde. Zudem sollte man bedenken, dass es mit seinen spielerischen Ansatz eher etwas für kleinere Klassen ist und nicht für ein berufliches Gymnasium.[4]

### 3 Lernziele

#### 3.1 Kognitive Lernziele

- Die SuS kennen die Begriffe Objekt, Attribut und Attributwert.
- Die SuS kennen die algorithmischen Grundstrukturen der Programmierung.
- Die SuS wenden die algorithmischen Grundstrukturen der Programmierung an.

#### 3.2 Psychomotorische Lernziele

- Die SuS manipulieren einen gegebenen Quelltext sodass er ein bestimmtes Problem löst.
- Die sind im Umgang mit der Tastatur, zur Eingabe von Quellcode, sicher.

#### 3.3 Affektive Lernziele

- Die SuS beachten die Regeln der Programmiersprache von Robot Karol.
- Die SuS werden auf die Vielfalt von Lösungen von Problemen in Robot Karol aufmerksam gemacht.

## 4 Kompetenzentwicklung

### 4.1 Fachkompetenz

- Die Schülerinnen und Schüler sind in der Lage selbstständig mit Robot Karol zu arbeiten.
- Die Schülerinnen und Schüler sind in Lage Objekt und Attribut zu Unterscheiden.
- Die Schülerinnen und Schüler können Programmcode zur Lösung beschriebener Probleme schreiben.
- Die Schülerinnen und Schüler können zu einem Objekt Attribute, Attributwerte und Methoden zuordnen.
- Die Schülerinnen und Schüler können die algorithmischen Grundstrukturen Folge, Wiederholung und Verzweigung in einem Robot Karol Programmcode identifizieren und eigenständig implementieren.
- Die Schülerinnen und Schüler können einen Programmcode in eine andere Darstellungsform(Struktogramm) überführen.

### 4.2 Lern/-Methodenkompetenz

- Die Schülerinnen und Schüler können eigenständig neue, das heißt, im Unterricht nicht vorher behandelte, Sachverhalte mithilfe von Robot Karol lösen.
- Die Schülerinnen und Schüler sind in der Lage, algorithmische Grundstrukturen begründet einzusetzen, um Probleme zu lösen.
- Die Schülerinnen und Schüler sind in der Lage sich in Gruppen zu organisieren, um Arbeitsaufträge zu lösen.

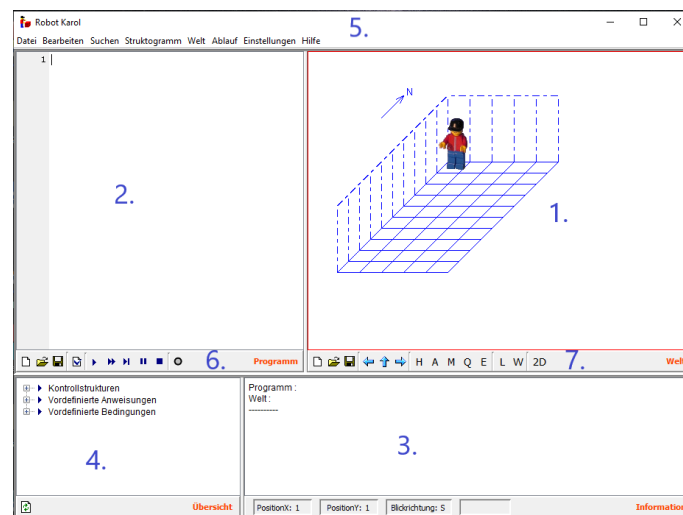
### 4.3 Sozialkompetenz

- Die Schülerinnen und Schüler können ihre Mitschülerinnen und Mitschülern bei der Verwendung des Programms Robot Karol gezielt unterstützen.
- Die Schülerinnen und Schüler können ihren Mitschülerinnen und Mitschülern konstruktives Feedback zur Aufgabenlösung kooperativer Aufgaben geben.
- Die Schülerinnen und Schüler sind in der Lage in Gruppen an einem Projekt zu arbeiten und Arbeitsteilung durchzuführen.

### 4.4 Selbstkompetenz

- Die Schülerinnen und Schüler sind in der Lage ihren Lernfortschritt im Vergleich zu den anderen Mitschülerinnen und Mitschülern selbständig einzuschätzen.
- Die Schülerinnen und Schüler sind in der Lage mit der Kritik von Lehrkräften und Mitschülerinnen und Mitschülern umzugehen und daraus ihren Lernprozess zu unterstützen.
- Die Schülerinnen und Schüler sind in der Lage eignen und fremde Programme zu beurteilen.

## 5 Prinzipieller Aufbau



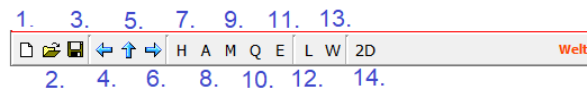
Prinzipiell kann das Programm Robot Karol in 4 Hauptfenster (1. - 4.) unterteilen werden.

Die Ansicht der Welt (1.) stellt die Auswirkungen während des Programmlaufes dar. Der Code zum Programm wird im Editor (2.) geschrieben. Eine Übersicht mit allen verfügbaren Befehlen, welche man im Editor verwenden kann, findet man im Feld mit der 4. Das vierte Hauptfeld (3.) stellt eine Informationsfläche dar, welche Meldungen zum Programm ausgibt (Fehlermeldungen ect.).

Zusätzlich zu den vier Hauptfenstern existiert noch eine Menüleiste (5.) darin sind am oberen Rand des Programmes alle Optionen des Programmes gelistet. Unter dem Editor befindet sich eine Programmleiste (6.), welche folgende Funktionen umfasst: Programm starten bzw. stoppen, Wiedergabegeschwindigkeit ändern, Syntax prüfen, Programme öffnen und speichern. Eine weitere Programmleiste befindet sich unter der Ansicht der Welt (7.). Dadurch lässt sich zum einen die Welt manuell ändern (Wechsel zwischen 2D/3D, Hinlegen, Bewegen ect.) und zum anderen lässt sich eine Welt speichern, Welten öffnen und neue Welten öffnen.

## 6 Handhabung

### 6.1 Ansicht der Welt



1. neue Welt erstellen (maximale Größe 100x100x10)
2. Welt öffnen/laden
3. Aktuelle Welt speichern
4. Karol dreht sich nach links
5. Karol geht ein Schritt nach vorn
6. Karol dreht sich nach rechts
7. Karol legt vor sich einen Ziegel
8. Karol hebt einen Ziegel vor sich auf
9. Karol löscht bzw. setzt eine Marke
10. Karol legt einen Quader vor sich hin
11. Karol entfernt einen Quader vor sich
12. Welt löschen
13. Welt wiederherstellen
14. zwischen 2D und 3D Ansicht wechseln

### 6.2 Erstellen und Ablauf des Programmcodes





- im Editor wird der Programmcode erstellt
- Groß- und Kleinschreibung ist irrelevant
- existiert farbliche Hervorhebung entsprechend der Syntax
- Zeilennummern können angezeigt werden
- Anweisungen können Zeilenweise oder durch Semikolon getrennt werden
- es besteht die Möglichkeit einen Objektbezug herzustellen

„Karol.Schritt()“

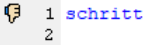
↔

„Schritt“

### 6.3 Programme starten

- Durch klicken des  startet das Programm
- Durch wählen von  kann das Programm im Schnelldurchlauf vollzogen werden.
- Weiterhin ist es möglich durch  das Programm in Einzelschritte zu durchlaufen.
- Außerdem kann man ein Programm mit diesen Buttons  pausieren und abbrechen

## 6.4 Meldungen auswerten

- Wenn  erscheint, hat Karol einen Befehl nicht verstanden. Im Informationsfeld erscheint dann eine Meldung mit Lösungsvorschlag:

```

Programm :
Welt:
-----
[Fehlernr 10][Zeile 2] Unbekannte Anweisung/Methode. Schreibweise kontrollieren!
  
```


- Wenn  erscheint, hat Karol einen Befehl verstanden kann diesen aber nicht ausführen. Im Informationsfeld erscheint dann eine Meldung mit Lösungsvorschlag:

```

Programm :
Welt:
-----

[Abbruch][Zeile 1]: Karol ist an der Wand angestoßen.
Programmabbruch
  
```

## 6.5 Programme speichern und laden

Jene Symbole  haben in dieser Reihenfolge folgende Aufgaben:

- neues Programm erstellen
- vorhandenes Programm öffnen/laden
- aktuelles Programm speichern

## 6.6 Sonstiges

- Endung beim speichern von Programmen: .kdp
- Endung beim speichern von Welten: .kdw
- über die *Menüleiste* → *Struktogramm* ist es möglich zum Programmcode ein Struktogramm zu erstellen
- über die *Menüleiste* → *Hilfe* finden ausführliche Tipps für Robot Karol

## 7 Screencast



<https://www.youtube.com/watch?v=cEX57ovslUU&feature=youtu.be>

## Literatur

- [1] Ulli Freiberger. Robot karol. *mebis Landesmedienzentrum Bayern*, 2018.
- [2] Lehrerinnen und Lehrern der Gymnasien Sächsischen Bildungsinstitut bzw. dem Landesamt für Schule und Bildung. *Lehrplan Sachsen Informatik Oberschule*. Sächsisches Staatsministerium für Kultus, 2019.
- [3] Lehrerinnen und Lehrern der Gymnasien Sächsischen Bildungsinstitut bzw. dem Landesamt für Schule und Bildung. *Lehrplan Sachsen Informatik Gymnasium*. Sächsisches Staatsministerium für Kultus, 2019.
- [4] Lehrerinnen und Lehrern der Gymnasien Sächsischen Bildungsinstitut bzw. dem Landesamt für Schule und Bildung. *Lehrplan Sachsen Informatik Berufsschule*. Sächsisches Staatsministerium für Kultus, 2018.

# Aufgaben zur Erprobung von Robot Karol

Modul: E-Learning und Tools

WiSe 19/20

## 1. Erste Schritte

Erstellen Sie eine neue Welt der Größe (10x10x10).

Nun lassen Sie Karol entlang der Wand zur nächsten Ecke laufen. Erweitern Sie ihr Programm, dass Karol eine Runde entlang der Wand läuft.

Zum Schluss soll Karol in jeder Ecke einen Ziegel ablegen.

## 2. Bibliothek

Karol soll an alle Seiten Ziegel legen, sodass ein rechteckiger Ring mit den Maßen 10x10 entsteht.

Robot Karol bietet die Möglichkeit in Programme bereits existierende Programme mit einzubinden, um somit die Anweisungen dieser Programme nutzen zu können. Dies ist mit dem Befehl *Einfügen* <Dateipfad>

\**Einfügen* möglich. Um daraus einen Vorteil zu ziehen, wäre es sinnvoll, Anweisungen, welche häufig genutzt werden, in einem Programm zu sammeln.

Schreiben dafür ein Programm *Bibliothek.kdp* in einen Ordner RobotKarol mit den Funktionen Umdrehen, SchrittZurueck und die Funktion RingBauen, welche Sie zuvor erstellt haben.

## 3. Treppe

Erstellen Sie eine neue Welt mit der Größe 8x8x10.

Schreiben Sie ein Programm, dass Karol eine Treppe der Höhe 4 bauen lässt. Da dies eine nützliche Funktion ist, schreiben Sie diese mit in ihre Bibliothek.kdp Datei.

## 4. Bau eines Schwimmbeckens

Schreiben Sie nun ein möglichst kurzes Programm, welches Robot ein quadratisches Schwimmbecken bauen lässt.

Die Seiten haben eine Breite von 6 Ziegeln und eine Höhe von 3.

Zudem steht Karol am Anfang und am Ende des Programms in seiner Ausgangsposition.

Tipp: Nutzen Sie für ihr Programm die Anweisung Wiederhole. Wenn Karol das Schwimmbecken baut, dann ändern Sie ihr Programm insofern ab, dass das Becken die Breite der Welt annimmt.

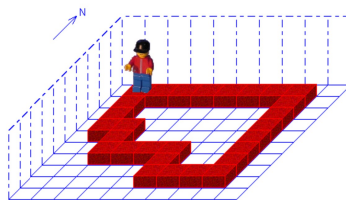
## 5. Goldener Weg

Gegeben ist ein Weg. Dieser soll noch mit Marken vergoldet werden.

Schreiben Sie ein Programm, welches den gegebenen Weg mit Marken verziert.

Die vorgegebene Welt befindet sich im Opal Kurs im Ordner Materialien.

Sie können diese Datei über die Menüleiste: *Welt* → *öffnen Welt...* laden.



(1)