

NT7 2.3 Beschreibung von Abläufen durch Algorithmen Ostereiersuche

Jahrgangsstufen	7
Fach/Fächer	Natur und Technik - Schwerpunkt Informatik
Übergreifende Bildungs- und Erziehungsziele	
Zeitraumen	20 min
Benötigtes Material	Werkzeug: <i>Robot Karol</i> ¹ ; <i>Aufgabe_Ostereier</i> als Papierkopie, Beispielwelt für <i>Karol Ostereier1.kdw</i> als Datei

Kompetenzerwartungen

Die Schülerinnen und Schüler ...

- analysieren und strukturieren geeignete Problemstellungen u. a. aus ihrer Erfahrungswelt (z. B. Bedienung eines Geräts), entwickeln Algorithmen zu deren Lösung und beschreiben diese unter effizienter Verwendung von Kontrollstrukturen.
- setzen unter sinnvoller Nutzung algorithmischer Bausteine einfache Algorithmen mithilfe geeigneter Programmierwerkzeuge um.

Aufgabe

In Karols Welt sind einige Ostereier versteckt (dargestellt durch Markierungen). Karol soll systematisch die ganze Welt durchsuchen, jedes gefundene Ei aufheben (Markierung entfernen) und dort ein kleines Tänzchen vollführen (z. B. vier Linksdrehungen).

Karol steht zu Beginn in Ausgangsstellung (beispielsweise in der Welt *Ostereier1.kdw*).

- Schreibe ein Programm für Karols Ostereiersuche, das auf jede Welt anwendbar ist, die keine Ziegel aber einige Markierungen enthält.
- Dein Programm sollte in allen möglichen Welten funktionieren.
Erstelle weitere Welten mit unterschiedlichen Markierungen, z. B. mit nebeneinanderliegenden Markierungen oder markierten Ecken. Teste, ob dein Programm auch in diesen Welten fehlerfrei abläuft.
- Füge geeignete Kommentare in deinen Programmtext ein, so dass deine Mitschülerinnen und Mitschüler deine Lösungsstrategie nachvollziehen können.

¹ Bezugsquelle <https://www.mebis.bayern.de/infportal/faecher/mint/inf/robot-karol/> (Stand 18.7.15)

Hinweise zum Unterricht

Voraussetzungen:

- Grundlegender Umgang mit der Programmierumgebung *Robot Karol*
- Wiederholung mit Bedingung; ein- und zweiseitige bedingte Anweisung
- Verknüpfung der algorithmischen Grundbausteine

Einsatz im Unterricht:

Die Aufgabe ist als Übungs- oder Prüfungsaufgabe verwendbar. Die Schülerinnen und Schüler sollten sich bereits mit einigen Problemstellungen, die mehrere algorithmische Grundbausteine miteinander verknüpfen, auseinandergesetzt haben. Teilaufgabe c erfordert ggf. eine vorherige Einführung durch die Lehrkraft und ist im Rahmen einer Metareflexion vorwiegend für leistungsstärkere Schülerinnen und Schüler gedacht. Auch die folgenden Varianten können der Differenzierung für unterschiedlich leistungsstarke Schülerinnen und Schüler dienen.

Vereinfachung und Hinführung:

- Karol beschränkt die Suche auf alle vor ihm liegenden Felder, bis er auf die Wand stößt.
- Karol läuft (ohne Suche) durch alle Felder der Welt.
- Die Abmessungen der Welt sind geradzahlig.

Erweiterung:

- In der Welt sind einzelne Ziegel verstreut, die nicht am Rand liegen und sich auch nicht gegenseitig berühren. Karol darf bei der Suche die Ziegel nicht betreten, sondern muss sie jeweils umlaufen.
- Karol steht zu Beginn an einer beliebigen Stelle der Welt.
- Karol muss jedes Osterei durch einen Ziegel ersetzen.

Lösungshinweise

Beispiellösung (*Ostereier1.kdp*) zu Teilaufgaben a und c:

```
// Bahnen abgehen bis der Rand erreicht ist
wiederhole solange NichtIstWand
// Vorwärtsgehen bis zur nächsten Wand
wiederhole solange NichtIstWand
wenn IstMarke dann // Osterei gefunden
// Tanz aus vier Linksdrehungen
wiederhole 4 mal
  Linksdrehen
*wiederhole
// Osterei aufheben
MarkeLöschen
*wenn
  Schritt
*wiederhole
// Ende vorwärtsgehen
// Wenden 1. Teil
Linksdrehen
```

```
// Zurückgehen auf der nächsten Bahn
// nur falls der Rand der Welt noch nicht erreicht
wenn NichtIstWand dann
  // Wenden 2. Teil
  Schritt
  LinksDrehen
// Zurückgehen bis zur nächsten Wand
wiederhole solange NichtIstWand
  wenn IstMarke dann // Osterei gefunden
    // Tanz aus vier Linksdrehungen
    wiederhole 4 mal
      LinksDrehen
    *wiederhole
  // Osterei aufheben
  MarkeLöschen
  *wenn
  Schritt
  *wiederhole
// Ende zurückgehen
// Wenden 1. Teil
RechtsDrehen
// Wenden 2. Teil falls der Rand noch nicht erreicht ist
wenn NichtIstWand dann
  Schritt
  RechtsDrehen
  *wenn
  *wenn
  *wiederhole
```

Alternativlösungen könnten auch Bedingungen wie *IstNorden* verwenden.

Die vorgestellte Lösung versagt, falls auch das letzte erreichte Feld markiert ist wie beispielsweise in der Welt *Ostereier2.kdw*. Falls die Schülerinnen und Schüler vergleichbare Lösungen haben, sollte die Lehrkraft gezielt eine entsprechende Welt einfordern oder aber *Ostereier2.kdw* zur Verfügung stellen. Die Datei *Ostereier2.kdp* enthält ein korrigiertes Programm.

Anregung zum weiteren Lernen

In der ISB-Handreichung „Kompetenzorientierte Aufgaben für das Fach Informatik“² thematisieren die Aufgaben 2.2.18 bis 2.2.19 die Wiederholung mit Bedingung in Verknüpfung mit der bedingten Anweisung. Auch hier wird das Werkzeug *Robot Karol* verwendet.

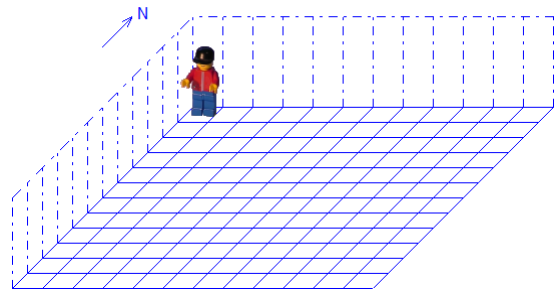
Weitere Standardaufgaben zur bedingten Wiederholung und der Verknüpfung der Grundbausteine mit dem Werkzeug *Robot Karol*, die teils als Vorstufen zur obigen Aufgabe zu verstehen sind und sowohl im Programmierwerkzeug als auch in eine Struktogrammdarstellung umgesetzt werden können, sind beispielsweise:

Schreibe ein Programm für Karol, in welchem er

- eine Reihe Ziegelsteine bis zur nächsten Wand legt.

² Staatsinstitut für Schulqualität und Bildungsforschung (Hsg.): Kompetenzorientierte Aufgaben für das Fach Informatik am Gymnasium; Brigg-Verlag, München 2012

- einen Ziegelrand rundum an den Rand der Welt legt.
- aus einer beliebigen Position bei beliebiger Blickrichtung in die Ausgangsstellung (siehe nebenstehendes Bild) geht. Gehe davon aus, dass keine unüberwindbaren Hindernisse im Weg stehen.
- von einem vor ihm liegenden Ziegelstapel alle Ziegel wegnimmt und auf das Feld rechts daneben hinlegt. Das Programm soll für unterschiedlich hohe Stapel geeignet sein.



- zu einem vor ihm liegenden Ziegelstapel rechts daneben einen doppelt so hohen baut. Der ursprüngliche Stapel darf dabei abgebaut werden. Das Programm soll für unterschiedlich hohe Stapel geeignet sein.
- bis zur nächsten Wand eine ziegelfreie Bahn schafft.
- die ganze Welt ziegelfrei macht. (Diese Aufgabe hat viel mit der Ostereieraufgabe gemeinsam.)
- 1000 Schritte vorwärts geht und sich dabei immer nach links dreht, wenn er vor einer Wand steht. Beschreibe den Weg, den Karol dabei abläuft.

Varianten:

- Falls Karol auf seinem Weg auf ein markiertes Feld tritt, soll er zusätzlich einen Tanz aufführen (z. B. sich einmal um seine Achse drehen).
- Falls Karol auf seinem Weg einen Ziegel vor sich sieht, dann soll er ihn wegnehmen und auf das Feld legen, auf dem er gerade steht.
- Falls am Rand der Welt einzelne Ziegel liegen, soll Karol ihnen ausweichen. Die Ausgangswelt hat dabei keine Ziegel in den Ecken.
- bis zur nächsten Wand geht und dabei jeden Ziegel, der vor ihm liegt, entfernt, im Gegenzug aber alle freien Stellen mit Ziegeln belegt.
- bis zur nächsten Wand geht und dabei links neben jede ziegelfreie Stelle einen Ziegel legt, die Plätze links neben den mit Ziegeln belegten Stellen aber freilässt bzw. ziegelfrei macht.

Weitere verwendbare Werkzeuge sind *Scratch* und seine Abkömmlinge. Mit ihnen werden die algorithmischen Kontrollstrukturen über eine grafische Oberfläche aufgebaut, ohne dass die Kenntnis von Schlüsselwörtern erforderlich ist und die Syntax explizit beachtet werden muss.

Mit *Scratch* können leistungsstarke Schülerinnen und Schüler auch Konzepte einsetzen, die über die Jahrgangsstufe 7 hinausgehen.

Der *Scratch*-Abkömmling *Snap* ist nur online verwendbar, eine Speicherung auf einem lokalen Datenträger ist nur schwer möglich. Bei Eröffnung von Online-Konten müssen die üblichen datenschutzrechtlichen Randbedingungen, wie z. B. die Zustimmung der Erziehungsberechtigten, beachtet werden.