

Algorithmen

Algorithmen sind Handlungsvorschriften aus mehreren Einzelschritten. Sie bilden damit nicht nur die Grundlage der Programmierung, sondern sind – beispielsweise beim Überqueren der Straße – auch in unserem Alltag präsent, meist, ohne dass es uns bewusst ist.

Mit Hilfe von Struktogrammen lassen sich Algorithmen übersichtlich darstellen.

Das Material „Programmieren mit Klaus“ bietet die Möglichkeit, mit Hilfe von Befehlskarten aus Karton Algorithmen zu legen. Dabei können Schülerinnen und Schüler erste Programmierkenntnisse erwerben – und das ohne Computer.

Die Befehlskarten für das Material „Programmieren mit Klaus“ sind in der Materialbox für den Aufbaukurs Informatik enthalten.

Die visuelle Programmiersprache Scratch wird von einem Team der Lifelong Kindergarten Group am Media Lab der Universität Massachusetts Institute of Technology (MIT) entwickelt. Scratch ist besonders gut geeignet, bereits Kinder und Jugendliche mit den Grundlagen der Programmierung vertraut machen.

Die aktuelle Scratch-Version 3.0 ist als Online-Version im Webbrowser und als Offline-Version (ab Windows 10, ab macOS 10.13) verfügbar.

Themen

	Seite		
Was ist ein Algorithmus?	38	1	
„Programmieren mit Klausur“	Anweisungen und Wiederholungen	40	2
	Abfragen und Bedingungen	43	3
	Arbeiten mit Unterprogrammen	46	4
	Variablen	49	5
Struktogramme	52	6	
Scratch 3.0	Erste Schritte in Scratch 3.0	55	7
	Anweisungen und Wiederholungen	58	8
	Abfragen und Bedingungen	62	9
	Variablen und Zufallszahlen	66	10
	Nachrichten	69	11
	Klone und Kostüme	75	12

Was ist ein Algorithmus?

Stell dir vor, du möchtest Plätzchen backen. Dann brauchst du dafür ein Rezept, in dem Schritt für Schritt genau gesagt wird, was zu tun ist: zuerst Zutaten abmessen, Teig kneten, Teig ausrollen, Plätzchen ausstechen und schließlich backen.

Dieses Rezept ist ein Algorithmus – eine genaue Folge von Anweisungen.

In deinem Alltag begegnest du ständig Algorithmen, meistens ohne es zu bemerken.

Algorithmen, also feste Abfolgen von Handlungen, findest du in vielen Bereichen deines Alltags:

Zähneputzen

1. Zahnbürste nehmen
2. Zahnpasta drauf
3. oben putzen
4. unten putzen
5. ausspülen

Anziehen

1. Unterwäsche
2. Socken
3. Hose
4. T-Shirt
5. Pullover

Händewaschen

1. Wasser aufdrehen
2. Hände nass machen
3. Seife nehmen
4. gründlich einseifen
5. abspülen
6. abtrocknen

In der Informatik werden Algorithmen verwendet, um Computer bestimmte Aufgaben ausführen zu lassen. Ein Algorithmus ist wie eine Anleitung für den Computer, die ihm sagt, wie er eine Aufgabe lösen soll.



congerdesign (Pixabay)

Diese Anleitungen bestehen immer aus klaren, logischen Schritten und helfen, Entscheidungen zu treffen, Probleme zu lösen oder Dinge zu sortieren. Algorithmen dürfen keine Lücken enthalten, und es dürfen keine Missverständnisse auftreten.

Zum Beispiel könnte ein Algorithmus dem Computer sagen, wie er eine Liste von Namen alphabetisch sortieren soll, oder ganz konkret: „Wenn die Zahl größer als 5 ist, dann mache dies, sonst mache das.“

Algorithmen können einfach oder sehr komplex sein. Ein einfacher Algorithmus könnte sein, die Zahlen von 1 bis 10 zu zählen. Ein komplexerer Algorithmus könnte einem Roboter sagen, wie er einen Raum reinigen soll, indem er Hindernisse umgeht und sicherstellt, dass er keinen Bereich vergisst.

Es gibt viele verschiedene Arten von Algorithmen, und Informatiker entwickeln ständig neue, um komplexere Aufgaben zu lösen. Algorithmen sind also ein wichtiger Teil der Informatik und helfen uns, Computer zu programmieren, die uns bei vielen Aufgaben unterstützen können.

Was ist ein Algorithmus?

Aufgabe 1

Was ist ein Algorithmus?

Beispiellösung

Ein Algorithmus ist eine Liste, die dir Schritt für Schritt sagt, was du tun musst."

Aufgabe 2

Ben hat folgenden Algorithmus für das Überqueren einer Straße erstellt:

1. Anhalten
2. Nach links schauen
3. Nach rechts schauen
4. Über die Straße gehen

Ist dieser Algorithmus richtig?

Schreibe den Algorithmus ggf. noch einmal vollständig auf.

Beispiellösung

1. Anhalten
2. Nach links schauen
3. Nach rechts schauen
4. Noch einmal nach links schauen
5. Wenn Fahrzeuge kommen, warten, bis die Straße frei ist
6. Wenn die Straße frei ist, über die Straße gehen

Aufgabe 3

Bring die Anweisungen dieses Algorithmus für das Legen eines Puzzles in die richtige Reihenfolge:

- Stück für Stück zusammensetzen
- Ecken legen
- Farben oder Muster sortieren
- Randteile heraussuchen

Beispiellösung

1. Randteile heraussuchen
2. Ecken legen
3. Farben oder Muster sortieren
4. Stück für Stück zusammensetzen

Aufgabe 4

Schreibe einen Algorithmus für das Zubereiten deines Frühstücks.

Beispiellösungen

1. Kaffee, Tee oder Kakao zubereiten
2. Brot toasten
3. Marmelade oder Butter auf das Brot streichen
4. Frühstück auf einem Teller anrichten

Aufgabe 5

Schreibe einen Algorithmus für das Berechnen des Flächeninhalts eines Rechtecks.

Beispiellösung

1. Länge ermitteln
2. Breite ermitteln
3. Flächeninhalt berechnen
4. Flächeninhalt notieren

Aufgabe 6

Schreibe einen Algorithmus für den Weg von deinem Wohnhaus zur Schule.

Beispiellösung

1. Verlasse das Haus
2. Gehe geradeaus bis zur ersten Kreuzung.
3. Biege rechts ab.
4. Gehe bis zur Ampel.
5. Wenn die Ampel rot ist, warte, bis sie grün wird.
6. Wenn die Ampel grün ist, überquere die Straße.
7. Gehe geradeaus bis zur Bushaltestelle.
8. Warte auf den Bus Nummer 5.
9. Steige in den Bus ein.
10. Fahre 3 Stationen mit dem Bus.
11. Steige an der Haltestelle „Schulplatz“ aus.
12. Gehe 100 Meter geradeaus bis zur Schule.

Abfragen und Bedingungen

Wiederholungen und Programmende



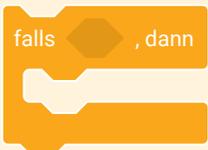
Mit diesem Block gibt man die Anweisung, dass ein Befehl oder eine Reihe von Befehlen wiederholt werden soll, und zwar so lange, bis das Skript durch einen Befehl gestoppt wird.

Für das Stoppen von Skripten kann dieser Block verwendet werden.



Das Ausklappenmenü bietet die Möglichkeit, nur das aktuelle Skript, nur die anderen Skripte der aktuellen Figur oder alles zu stoppen.

Abfragen und Bedingungen



Sollen Befehle nur ausgeführt werden, wenn eine Bedingung erfüllt ist, wird dieser Block eingesetzt.

Die Bedingungen werden mit Hilfe sechseckiger Blöcke wie diesem angegeben:



Er legt fest, dass die nachfolgenden Befehle nur ausgeführt werden, nachdem die Figur die Farbe Gelb berührt hat.

Der Bedingungs-Block wird in das sechseckige Feld im Abfrage-Block eingefügt.



Die in der Bedingung enthaltene Farbe kann frei gewählt werden. Mit einem Klick in das Farbfeld öffnet sich ein Menü zum Einstellen der Farbe. Die Pipette ermöglicht das Übernehmen einer beliebigen Farbe aus dem gewählten Bühnenbild.



Sollen Befehle nur ausgeführt werden, sobald eine Taste gedrückt wird, verwendet man diesen Block.



Durch einen Klick auf das Dreieck öffnet sich ein Ausklappenmenü, in dem weitere mögliche Tasten gewählt werden können.

Zeigen und Verstecken



Mit Hilfe dieser beiden Blöcke kann eine Figur während eines Skriptes ein- und ausgeblendet werden.

Meldungen

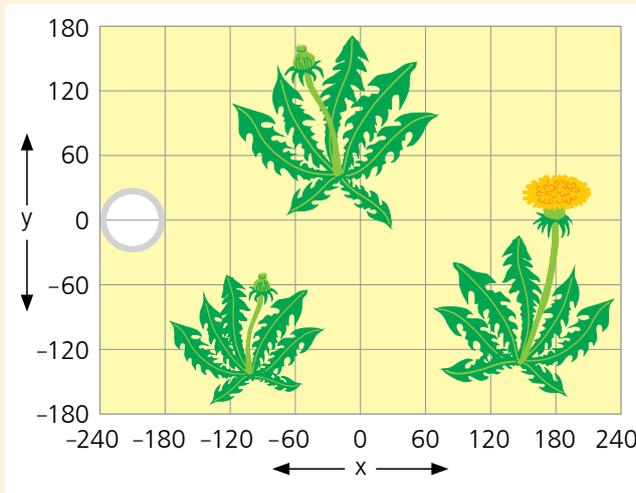


Mit diesem Block kann man eine Figur eine Meldung oder Mitteilung ausgeben lassen. Sie wird in das Parameterfeld eingegeben und erscheint während des Skriptes als Sprechblase.

Abfragen und Bedingungen

Bühnenbild

Loewenzahn.svg



Figur

Seifenblase.svg



Beachte, dass die Figur sich nicht mit ihrem linken oder rechten Rand, sondern mit ihrem Mittelpunkt auf den eingestellten Koordinaten platziert.

Aufgabe 1

Sobald auf die grüne Fahne  geklickt wird, soll die Seifenblase in der Mitte des grauen Ringes erscheinen.

```

Wenn  angeklickt wird
  gehe zu x: -210 y: 0
  zeige dich
    
```

Aufgabe 2

Sobald die Cursortasten gedrückt werden, soll sich die Seifenblase einen 10er-Schritt in die jeweilige Richtung bewegen.

Platziere dafür neben dem Skript aus Aufgabe 1 ein weiteres Skript, das ebenfalls mit einem Klick auf die grüne Fahne  gestartet wird. Die Abfragen, ob eine Taste gedrückt wird, sollen fortlaufend ausgeführt werden.

```

Wenn  angeklickt wird
  wiederhole fortlaufend
    falls Taste Pfeil nach oben gedrückt? , dann
      setze Richtung auf 0 Grad
      gehe 10 er Schritt
    falls Taste Pfeil nach unten gedrückt? , dann
      setze Richtung auf 180 Grad
      gehe 10 er Schritt
    falls Taste Pfeil nach rechts gedrückt? , dann
      setze Richtung auf 90 Grad
      gehe 10 er Schritt
    falls Taste Pfeil nach links gedrückt? , dann
      setze Richtung auf -90 Grad
      gehe 10 er Schritt
  
```

Abfragen und Bedingungen

Aufgabe 3

Sobald die Seifenblase ein grünes Löwenzahnblatt berührt, soll sie platzen (also verschwinden). Füge diese Abfrage zu dem Skript aus Aufgabe 1 hinzu. Die Abfrage soll fortlaufend ausgeführt werden.



Aufgabe 4

Füge eine weitere Anweisung in das Skript aus Aufgabe 3 ein: Sobald die Seifenblase ein grünes Löwenzahnblatt berührt hat und daraufhin geplatzt (also verschwunden) ist, soll das gesamte Skript gestoppt werden. (Sobald das Skript gestoppt ist, verschwindet der gelbe Saum um das Skript herum.)



Hinweis: Wenn der gelbe Saum um das Skript nicht verschwindet, ist das ein Hinweis darauf, dass das Skript noch einen Fehler enthält.

Abfragen und Bedingungen

Aufgabe 5

Sobald die Seifenblase die gelbe Löwenzahnblüte berührt, soll sie platzen (also verschwinden). Auch diese Abfrage soll fortlaufend ausgeführt werden. Füge diese Abfrage zu dem Skript aus Aufgabe 4 hinzu.



Aufgabe 6

Da es gar nicht so einfach ist, die Seifenblase bis zur Löwenzahnblüte zu manövrieren, soll zwei Sekunden lang die Meldung „Super!“ erscheinen, bevor die Seifenblase an der Löwenzahnblüte zerplatzt (also verschwindet). Füge diese Anweisung zu dem Skript aus Aufgabe 5 hinzu.

