ALGORITHMEN - LERNÜBERSICHT

DEFINITION

Ein Algorithmus ist eine Verarbeitungsvorschrift, die aus einer **endlichen** Folge von **eindeutig ausführbaren** Anweisungen besteht.
Unter gleichen Voraussetzungen liefert die Ausführung eines Algorithmus stets gleiche Ergebnisse.

EIGENSCHAFTEN

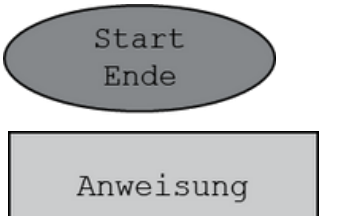
AUSFÜHRBARKEIT	EINDEUTIGKEIT	ENDLICH
Jede einzelne Anweisung muss für den Ausführenden Algorithmus verständlich und eindeutig ausführbar sein.	Mit jeder Anweisung ist auch die nächstfolgende festgelegt. Wird der Algorithmus mit den gleichen Voraussetzungen gestartet, so liefert die Ausführung stets gleiche Ergebnisse.	Ein Algorithmus besteht aus endlich vielen Anweisungen. In der Praxis soll ein Algorithmus nach endlich vielen Schritten ein Resultat liefern

DARSTELLUNG

DEFINITION: FLUSSDIAGRAMME

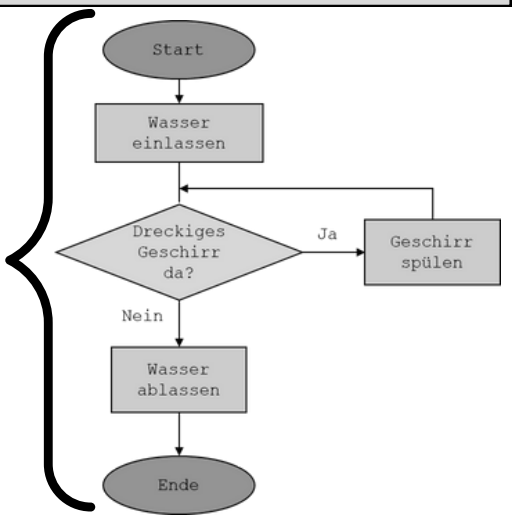
Ein Flussdiagramm ist eine grafische Darstellung eines Algorithmus, der als Programm umgesetzt werden soll. Sie helfen beim Schreiben und Verstehen von Programmen, da sie den Ablauf eines Algorithmus anschaulich beschreiben. Der Name kommt daher, dass sie den Ablauf oder "Fluss" eines Programms grafisch darstellen.

Ein anderer Name für ein Flussdiagramm für ein Algorithmus ist Programmablaufplan (PAP)



Pfeile verbinden Anweisungen

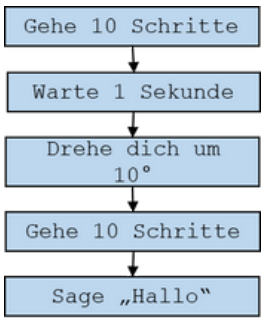
SEQUENZ
=Abfolge von Anweisungen

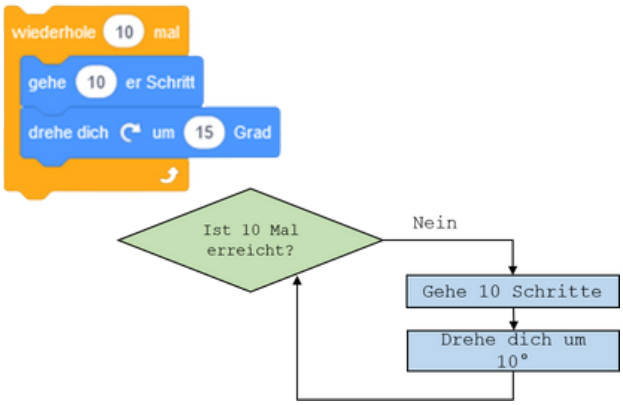


GRUNDSTRUKTUREN

SEQUENZ

Eine Folge von Methoden, die nacheinander ausgeführt werden, heißt **Sequenz**.
Rechts sehen wir einmal einen Code in Scratch und das dazugehörige Flussdiagramm.



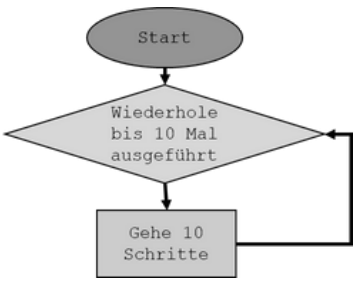


SCHLEIFEN / WIEDERHOLUNGEN

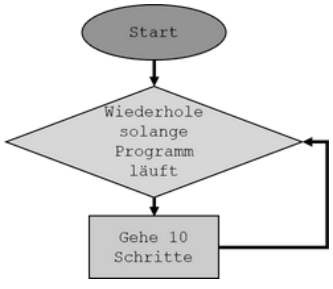
Schleifen sind eine Möglichkeit, bestimmte Aufgaben wiederholt auszuführen. Eine Schleife besteht aus einer Bedingung und einem Code-Block, der ausgeführt wird, solange die Bedingung wahr ist.
Links ist wieder ein Scratch Code und ein Flussdiagramm

UNTERSCHIEDLICHE ARTEN VON SCHLEIFEN IN SCRATCH UND ALS FLUSSDIAGRAMM

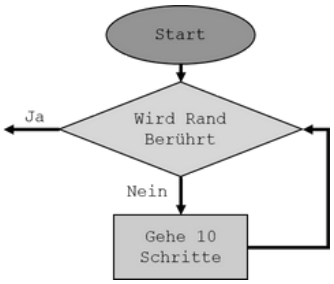
ZÄHLSCHLEIFE



DAUERSCHLEIFE

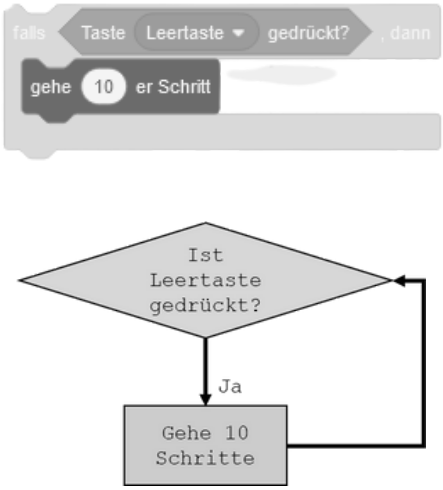


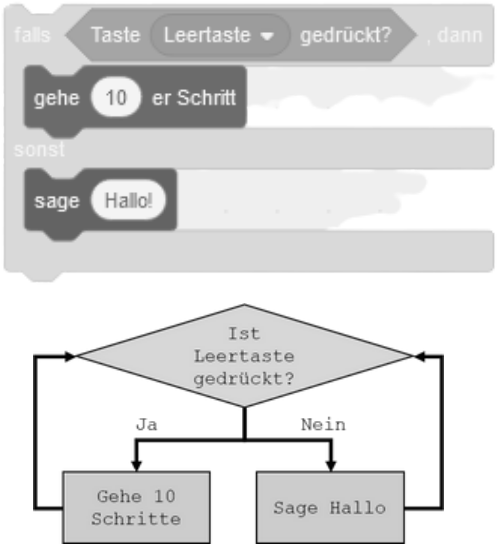
SCHLEIFE MIT BEDINGUNG



BEDINGUNGEN

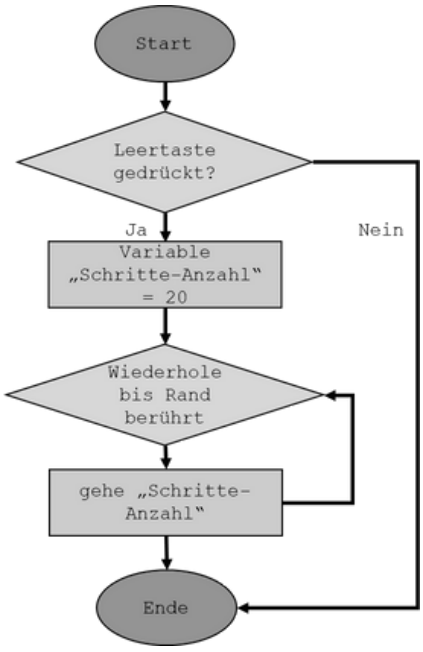
Bei Bedingungen wird überprüft, ob eine Voraussetzung erfüllt ist oder nicht. Je nachdem wird ein Code dann ausgeführt oder nicht.
Häufig werden auch Schleifen mit Bedingungen verknüpft.





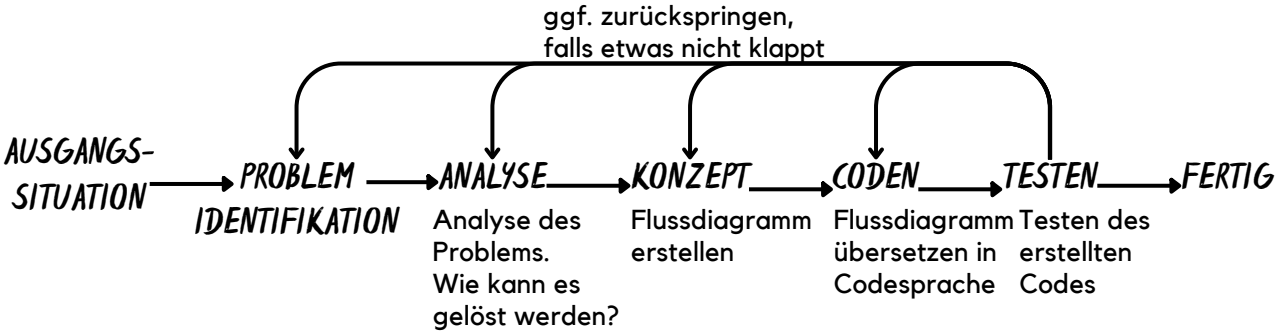
VARIABLEN

Variablen können genutzt werden, um Daten zu speichern. Rechts siehst du eine Sequenz, wo in der Variable Namens "Schritte-Anzahl", die Anzahl der Schritte gespeichert wird, die gegangen werden sollen.
Variablen in Scratch bestehen aus den folgenden Teilen:
Variablen Wert und Variablenname (hier: Schritte-Anzahl) und den Variablen-Wert (hier: 20)



ALGORITHMEN ENTWICKELN - ABLAUF

Das folgende habt ihr in den letzten Stunden eigentlich schon die ganze Zeit gemacht. Ihr seid ausgehend von einem Problem in die Entwicklung eines Problems gegangen. Hier ist es nur noch einmal schematisch dargestellt.
Was folgt ist ein Beispiel für den dargestellten ablauf.



AUSGANGSSITUATION

Wir wollen einen Algorithmus, wo eine Scratch-Figur mit einem Stift, der ein- und ausgeschaltet werden kann, ein Quadrat mit der Seitenlänge 30 zeichnet. Das Quadrat soll an einer Zufallsposition gezeichnet werden

PRROBLEM IDENTIFIKATION

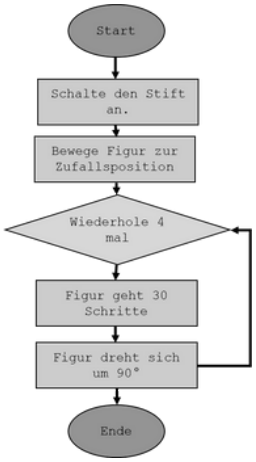
Wie muss der Algorithmus aussehen, damit die Scratch Figur ein Quadrat zeichnet?

ANALYSE

Wie kann ein Quadrat gezeichnet werden?

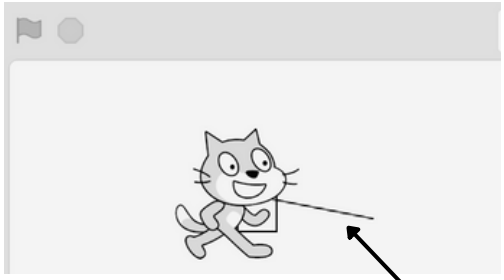
- indem die Figur mit einem angeschalteten Stift 30 Schritte geht und sich bei den Kanten um 90° dreht.
- Dies muss so lange wiederholt werden, bis das Quadrat fertig gezeichnet ist.

KONZEPT - FLUSSDIAGRAMM



CODEN

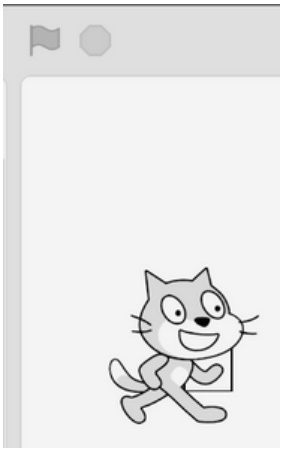
TEST-ERGEBNIS



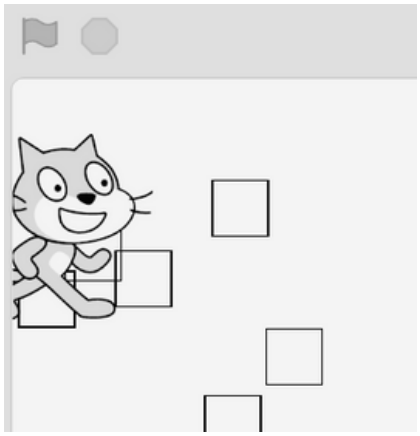
Der Stift wird zu früh eingeschaltet. Es entsteht ein ungewollter Strich. Die Figur muss erst zur Zufallsposition und in der Schleife den Stift anschalten. Am ende muss der Stift ausgeschalten werden.

CODEN 2. VERSUCH

TEST-ERGEBNIS 2. VERSUCH



TEST-ERGEBNIS NACH MEHREREN AUSFÜHRUNGEN



Ergebnis: Eine Figur zeichnet in Scratch bei jeder Ausführung ein Quadrat.

ZUSAMMENFASSUNG

Um von einem Problem zu einem Algorithmus zu kommen werden mehrere Schritte hintereinander ausgeführt. Manchmal ist es notwendig beim auftreten von Problemen oder Fehlern ein oder mehrere Schritte zu wiederholen und seine Zwischenergebnisse den neuen Erkenntnissen entsprechend anpassen. Das haben wir in dem Beispiel auch gemacht, als wir gemerkt haben, das der erste Algorithmus mehr als nur das Quadrat zeichnet. Also sind wir nochmal ins Coden gegangen und haben den Code so angepasst, dass nur das Quadrat gezeichnet wird und der Algorithmus mehrere Male hintereinander ausgeführt werden kann.

LERNINHALTE LEISTUNGSKONTROLLE:

Für die Leistungskontrolle solltet ihr euch mit dem folgenden vertraut gemacht haben:
Theorie

- Definition Algortihmus
- Eigenschaften von Algorithmen und deren Definititionen
- Grundstrukturen von Algorithmen, deren Definition und Gebrauch
  - Sequenz
  - Schleifen (Zählschleife, Dauerschleife, Schleife mit Bedingung)
  - Bedingungen
  - Variablen
- Aufbau eines Flussdiagramms

Praxis

- Verstehen was ein Algorithmus macht anhand eines Flussdiagrammes
- Angeben, was die Ausgabe eines Algorithmus ist, bei einer bestimmten Eingabe.
  - Vgl. Übungsblätter **5.2**
- Erstellen eines Flussdiagramms anhand eines vorgegebenen Problems
- Erkennen von algorithmischen Strukturen in deinem Alltag (Rezept, Weg zur Schule, ...)
- Erkennen von Fehlern in einem Algorithmus und den verbessern können
  - Vgl. **5.2**
- Den Ablauf eines Algorithmus beschreiben.
  - Bspw.: Was wird bei der Erfüllung unterschiedlicher Bedingungen gemacht (vgl. **5.1**)
  - Vgl. **5.2**