

Algorithmus¹

Definition

Ein **Algorithmus** ist eine **eindeutige, ausführbare** Folge von Anweisungen **endlicher** Länge zur **Lösung** eines **Problems**.

Ein Algorithmus besteht

- aus einem Deklarationsteil (Was wird benötigt?), z.B. bei einem Rezept die Zutaten
- und einem Anweisungsteil (Wie wird das Problem gelöst?), z.B. beim Rezept die Verarbeitungsanweisungen.

Beispiele für Algorithmen

- Vorschriften zum Addieren, Subtrahieren oder Multiplizieren von Zahlen
- euklidischer Algorithmus zur Berechnung des ggT zweier Zahlen
- (gute) Bastelanleitungen
- Spielregeln und alltägliche Vorschriften (sind aber leider nur selten exakt ausformuliert und enthalten Teile, die unterschiedlich interpretiert werden können)
- Spielen einer Melodie (z.B. nach Noten)
- Bedienung eines Handys zum Erreichen eines bestimmten Ziels (z.B. Ändern einer Einstellung des Systems)

Genauigkeit der Definition

Die obige Definition für den Begriff des Algorithmus ist zwar verständlich, aber mathematisch ungenau. In der Mathematik ab Beginn des 20. Jahrhunderts wurden eine Reihe von Ansätzen entwickelt, die zu einer genauen Definition führen sollten. Alan Turing zeigte, dass alle gefundenen Methoden letztlich alle gleich leistungsfähig (gleich mächtig). Das gelang ihm mit Hilfe eines mathematischen Modells: der Turing-Maschine. Mit Hilfe des Begriffs der **Turingmaschine** konnte folgende formale Definition des Begriffs formuliert werden:

Eine Berechnungsvorschrift zur Lösung eines Problems heißt genau dann Algorithmus, wenn eine zu dieser Berechnungsvorschrift äquivalente Turingmaschine existiert, die für jede Eingabe, die eine Lösung besitzt, stoppt.

Daraus ließen sich die folgenden Eigenschaften ableiten, die einen Algorithmus (für eine Maschine) genauer beschreiben.

Eigenschaften eines Algorithmus für eine Maschine

1. Endlichkeit (Finitheit)

Die Beschreibung eines Algorithmus besitzt eine **endliche** Länge, d. h. er besteht aus einer begrenzten Anzahl von Anweisungen mit begrenzter Länge. Zudem darf ein Algorithmus zu jedem Zeitpunkt für seine Daten nur endlich viel Platz belegen -> Das Verfahren muss in einem endlichen Text eindeutig beschreibbar sein.

2. Ausführbarkeit

Jede einzelne Anweisung eines Algorithmus muss vom Computer (vom Menschen) ausführbar sein -> Jeder Schritt des Verfahrens muss tatsächlich ausführbar sein.

3. Terminierung

Der Algorithmus ist nach endlich vielen Schritten **beendet**, d. h. er liefert ein Ergebnis oder hält an.

Für die Praxis werden Algorithmen häufig auf die folgenden Eigenschaften eingeschränkt:

Allgemeinheit

Ein Algorithmus ist allgemeingültig, d. h. er löst eine **Vielzahl von Problemen** (der gleichen Art). Die Auswahl eines einzelnen konkreten Problems erfolgt über **Eingabedaten** oder **Parameter**.

Beispiel: Um das Problem „Dreieck zeichnen“ in TigerJython mit der Turtle zu lösen, kann man eine Funktion `zeichneDreieck (hoehe)` schreiben, die für unterschiedliche Werte des Parameters `hoehe` verschieden große Dreiecke zeichnet.

Eindeutigkeit

An jeder Stelle des Algorithmus muss **eindeutig** festgelegt sein, was zu tun ist und welcher Schritt der Nächste ist. Dafür muss jede Anweisung unmissverständlich formuliert sein. Für die Umsetzung eines Algorithmus in eine Programmiersprache bedeutet das eine korrekte **syntaktische und semantische** Formulierung.

Determiniertheit

Wird ein Algorithmus mit den **gleichen Eingabewerten** und **Startbedingungen** wiederholt, so liefert er stets das **gleiche Ergebnis**.

¹ Mit Material von: http://www.info-wsf.de/index.php/Grundlagen_der_Programmierung