

Kapitel 3.3 Programmlogik, Algorithmen und Darstellungsmittel für Programmabläufe

Prof. Dr.-Ing. Martin Strube / Prof. Dr.-Ing. Udo Triltsch

Inhalt

- Programmablauf
 - Sie kennen die vier grundlegenden Arten von Programmabläufen.
 - Sie kennen Bedingungen zur Steuerung eines Programmablaufs.
- Algorithmen
 - Sie verstehen, was man im Allgemeinen unter dem Begriff Algorithmus versteht.
 - Sie kennen eine in der Informatik gebräuchliche Definition des Begriffs Algorithmus.
 - Sie kennen die wesentlichen Eigenschaften von Algorithmen.
- Darstellungsmittel für Programmabläufe
 - Sie kennen die wesentlichen Darstellungsmittel für Programmabläufe und können diese benennen.
 - Sie können eine beschriebene Aufgabe mit Hilfe eines Programmablaufplans darstellen.

Abstraktion der Wirklichkeit

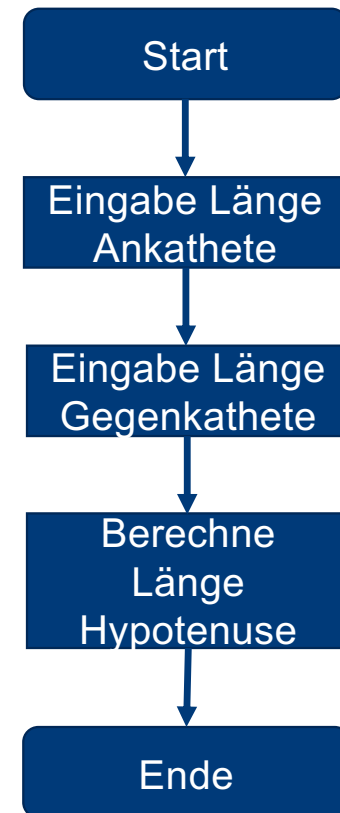
- Viele Aufgabenstellungen erscheinen in ihrer ganzheitlichen Betrachtung umfangreich und kompliziert.
- Die Vorbereitung der Entwicklung einer Software zur Lösung solcher Aufgabenstellungen erfordert daher Vorarbeit – „**stop coding, start thinking**“.
- Ein wichtiges Hilfsmittel ist dabei die Abstraktion der Wirklichkeit – d.h. das Problem auf das Wesentliche reduzieren.



Quelle: <https://www.kunstnet.de/werk/397498-out-of-order>

Programmablauf

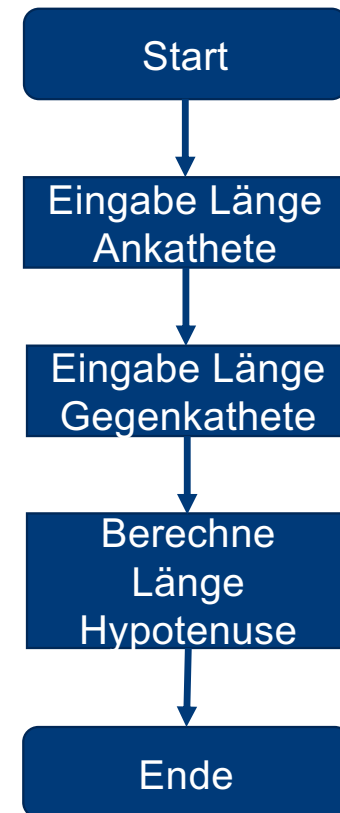
- Unter dem Begriff Programmablauf versteht man die programmiertechnische Umsetzung eines Algorithmus als Folge von Schritten in einem Programm.
- Diese Schritte können in verschiedenen Formen auftreten:
 - a) **sequenziell** ein Schritt nach dem anderen
 - b) **iterativ** mehrfache Ausführung best. Schritte
 - c) **auswählend** von mehreren Alternativen Schrittfolgen wird eine ausgeführt
 - d) **springend** die Ausführung der Schritte an einer bestimmten Stelle



Programmablauf

a) sequenziell:

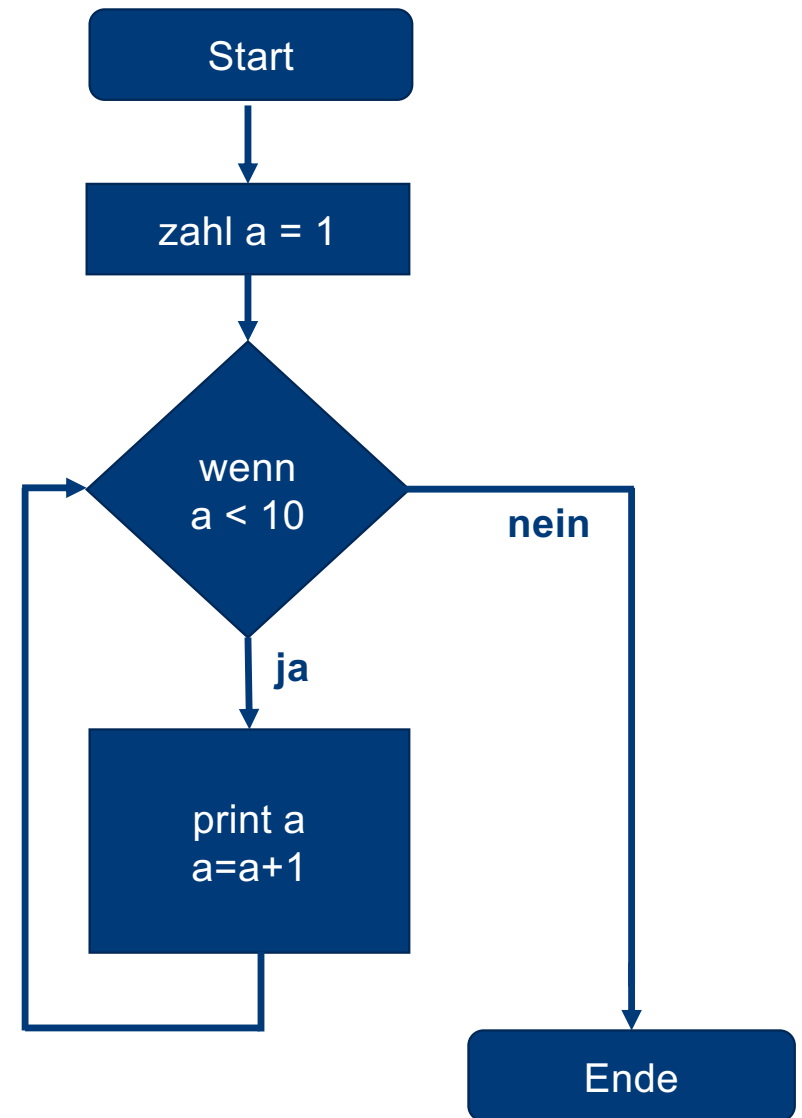
- Ein Schritt folgt dem nächsten, mit fest vorgegebener Folge.
- Sequenzen treten in fast jedem Programm auf.
- Im Beispiel rechts ist der Programmablauf zur Berechnung der Hypotenuse eines Dreiecks dargestellt, indem zuerst die Eingabe der Länge von An- und Gegenkathete erfolgen muss und anschließend die Länge der Hypotenuse berechnet wird.



Programmablauf

b) iterativ:

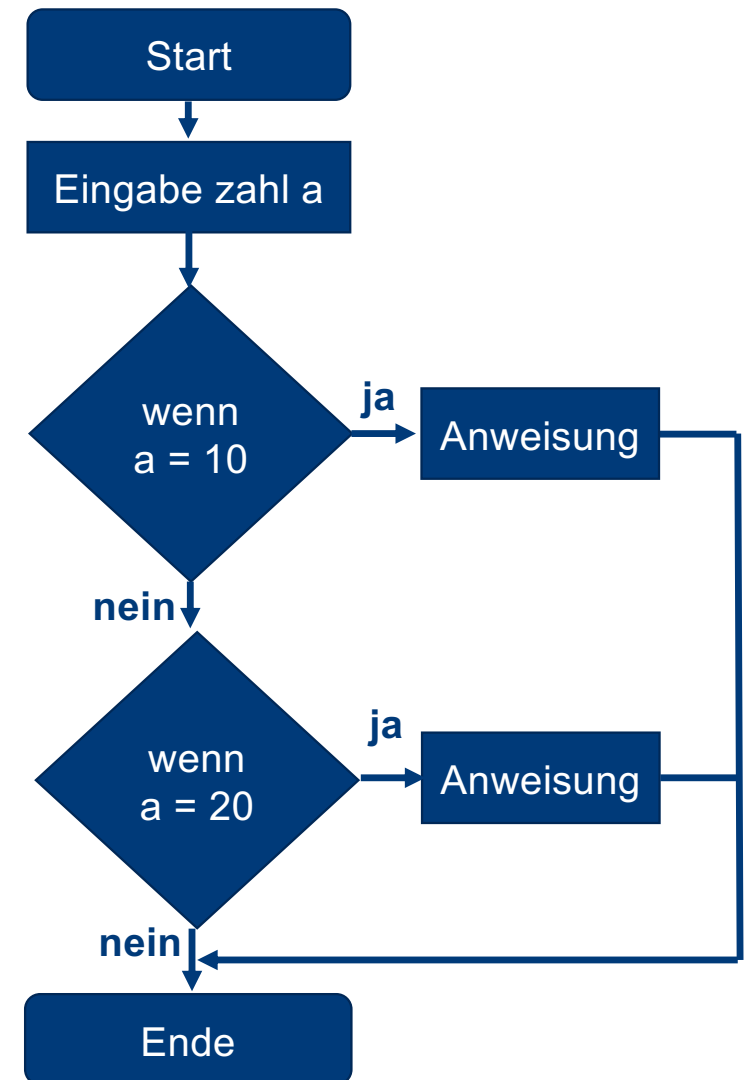
- Schritte werden in Form einer Schleife (wie beim Replay eine Songliste) wiederholt.
- Diese Wiederholung bezeichnet man als Iteration.
- Vor jeder Wiederholung wird anhand von Bedingungen (der Schleife) geprüft, ob eine Wiederholung notwendig ist.
- Im Beispiel rechts ist der Programmablauf zur Ausgabe (print) aller Zahlen von 1 bis 9 dargestellt.
- In der Bedingung wird jeweils geprüft, ob a kleiner als 10 ist. Wenn diese Bedingung erfüllt ist, wird der aktuelle Wert von a ausgegeben und a danach um 1 erhöht.



Programmablauf

c) auswählend:

- Der Programmfluss wird mithilfe einer Entscheidungsanweisung in Richtung einer von mehreren Alternativen geleitet.
- Auch hier wird vor Ausführung einer der möglichen Alternativen zunächst eine Bedingung geprüft.
- Im Beispiel rechts wird zuerst geprüft, ob eine eingegebene Zahl a den Wert 10 hat. Wenn ja, wird eine bestimmte Anweisung ausgeführt und das Programm endet danach.
- Hat die Zahl nicht den Wert 10, so wird im Anschluss geprüft, ob die Zahl a den Wert 20 hat. Wenn ja, wird eine bestimmte Anweisung ausgeführt und das Programm endet danach.
- Hat die Zahl a weder den Wert 10 noch den Wert 20, so endet das Programm ohne eine Anweisung auszuführen.



Programmablauf

d) springend:

- Der Programmfluss wird durch eine Sprunganweisung an einer bestimmten Stelle des Programms fortgesetzt



Quelle: <https://www.welt.de> / Sebastian Deeg

Bedingungen zur Steuerung des Programmablaufs

- Die Steuerung des Programmablaufs ist an Bedingungen gekoppelt, die sich mit logischen Tests vergleichen lassen (vergl. Mathe I Vorlesung, Kapitel Logik).
- Bei einem solchen Test wird jeweils geprüft, ob eine Bedingung erfüllt (wahr) ist oder nicht (falsch).
- Diese Bedingungen können auch kombiniert werden.
- Beispiel: **Wenn** die Zahl **a größer 10** ist
und
die Zahl **a ungleich 50** ist
und
die Zahl **a kleiner 100** ist, **dann ...**

Einführung Algorithmen

- Ganz allgemein kann man einen Algorithmus als eine eindeutige Handlungsvorschrift zur Lösung eines bestimmten Problems verstehen.
- Etwas detaillierter kann man dies auch wie folgt beschreiben:
 - „1. Eine präzise, d.h. in einer festgelegten Sprache abgefasste, endliche Beschreibung eines allgemeinen Verfahrens unter Verwendung elementarer Verarbeitungsschritte zur Lösung einer gegebenen Aufgabe.
 - 2. Lösungsverfahren in Form einer Verfahrensanweisung, die in einer wohldefinierten Abfolge von Schritten zur Problemlösung führt.“

<https://wirtschaftslexikon.gabler.de/definition/algorithmus-27106/version-250769>

- Mit Bezug zu unserer Vorlesung gilt:

Ein Algorithmus ist eine in der Beschreibung und Ausführung **endliche**, **deterministische** und **eindeutige** Vorschrift zur Lösung eines Problems, die **effizient** sein soll.

Wichtige Eigenschaften eines Algorithmus

- **Eindeutigkeit:**

- Jeder Schritt muss eindeutig sein und an jeder Stelle muss festgelegt sein, welcher Schritt der nächste ist.

- **Terminierung/Endlichkeit**

- Jeder Schritt besteht aus einer begrenzten Anzahl von Anweisungen mit begrenzter Länge. Die Anzahl der Schritte in einem Algorithmus ist ebenfalls begrenzt.

- **Determiniertheit**

- Ein Algorithmus liefert bei gleichen Eingabewerten und Startbedingungen auch stets das gleiche Ergebnis.

- **Effizienz**

- Ein Algorithmus soll sparsam bezüglich der Ressourcen (z.B. Rechenzeit und Speicherplatz), die er zur Lösung eines festgelegten Problems beansprucht, ausgelegt werden.

Kochrezepte sind im weitesten Sinne auch Algorithmen

- **Kochrezept für Rührei**

Zutaten: 3 Eier, Butter, Pfeffer, Salz

Zubereitung:

- 3 Eier aufschlagen und Eiklar und Eigelb von der Schale trennen.
 - Eiklar und Eigelb mit einer Messerspitze Pfeffer und Salz verrühren.
 - Pfanne auf mittlerer Stufe der Kochplatte erhitzen und 30 g Butter hinzugeben.
 - Wenn die Butter geschmolzen ist, dann Ei-Masse hinzugeben.
 - Unter ständigem Rühren solange braten, bis die Konsistenz von flüssig in fest übergegangen ist.
- Befolgt man diesen Algorithmus, so entsteht dabei aus rohen Eiern Rührei.



Ein **Algorithmus** ist eine eindeutige, ausführbare Folge von Anweisungen endlicher Länge zur Lösung eines Problems. Ein Algorithmus besteht aus einem Deklarationsteil (Was wird benötigt?) und einem Anweisungsteil (Wie wird das Problem gelöst?).

Übung Algorithmus

In folgendem Worträtsel sind in einem rechteckigen Feld Worte versteckt, die horizontal, vertikal oder diagonal (auch rückwärts) geschrieben sind. In einem kleinen Feld lassen sich gesuchte Worte noch schnell selbst finden. Schon bei 50 x 20 Zeichen ist es aber sehr mühsam.

Entwickeln Sie einen Algorithmus, der in einem beliebigen Feld von Buchstaben die Wörter:

MASCHINENBAU, INFORMATIK, HALLO, PYTHON, OSTFALIA

findet und die Position des Anfangsbuchstabens und die Richtung des Worts als Himmelsrichtung (N, O, S, W, NO, NW, SO, SW) ausgibt.

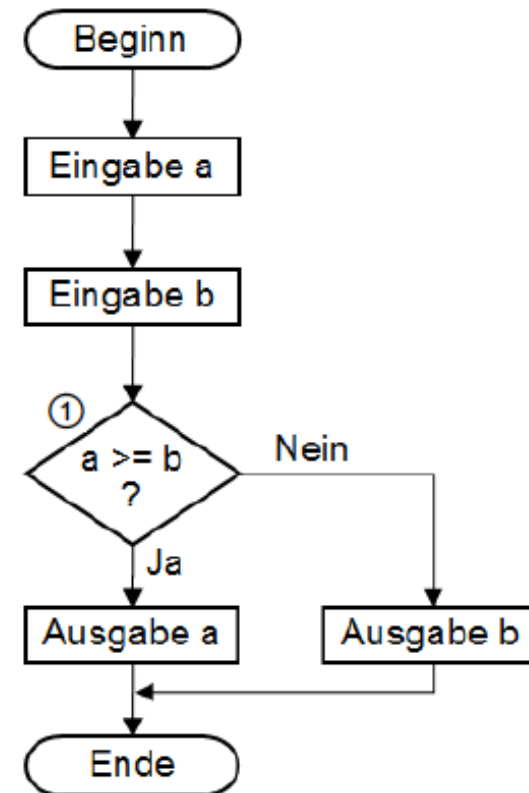
O	K	H	P	K	X	M	P	U	T	K	R	P	N	D	V	H	I	X	M
Z	O	I	J	V	X	G	Y	Q	Q	E	H	A	L	L	O	J	S	H	X
D	P	S	T	D	Z	V	T	Y	G	W	Y	E	M	G	D	F	N	Q	X
H	J	Y	T	A	P	F	H	F	O	I	K	B	H	X	T	R	L	T	Q
O	H	C	N	F	M	H	O	R	E	W	W	O	R	X	J	W	N	M	G
R	T	I	R	S	A	R	N	M	A	S	C	H	I	N	E	N	B	A	U
C	I	Z	G	X	Z	L	O	P	T	G	Z	E	N	S	C	X	Y	N	Y
U	E	G	K	Z	P	J	I	F	O	L	U	A	X	D	I	Q	E	W	D
R	J	D	H	Z	X	C	A	A	N	I	B	H	L	Y	E	H	M	M	H
D	J	I	L	O	J	D	F	T	M	I	H	X	R	D	I	S	Y	I	T

Entwurfstechniken für den Programmablauf

- Die Herausforderung des Programmierens liegt nicht im Schreiben des Programmcodes, sondern in der Entwicklung eines geeigneten Algorithmus zur Lösung der Aufgabenstellung.
- Oft sind zu diesem Zeitpunkt auch Kollegen an der Problemlösung beteiligt, die die jeweilige Programmiersprache nicht beherrschen, aber trotzdem an der Entwicklung eines Algorithmus mitwirken.
- Zur Vereinfachung und damit auch zum besseren Verständnis verwendet man daher zur Lösungsfindung Beschreibungsmittel, die ein einfaches Nachvollziehen des geplanten Lösungsweges ermöglichen.
- Man unterscheidet hierbei **grafische** (z.B. Programmablaufpläne oder Struktogramme) und **textuelle Beschreibungsmittel** (z.B. Pseudocode).

Programmablaufplan (PAP)

- Der Begriff PAP ist in der DIN 66001 genormt.
- In einem PAP werden grafische Symbole verwendet, um die Programmstruktur bildhaft darzustellen.
- PAP werden meist in mehreren Stufen entwickelt, wobei man zunächst die Grobstruktur des Programmablaufs definiert, die im Anschluss immer weiter verfeinert wird.



Programmablaufplan (PAP)

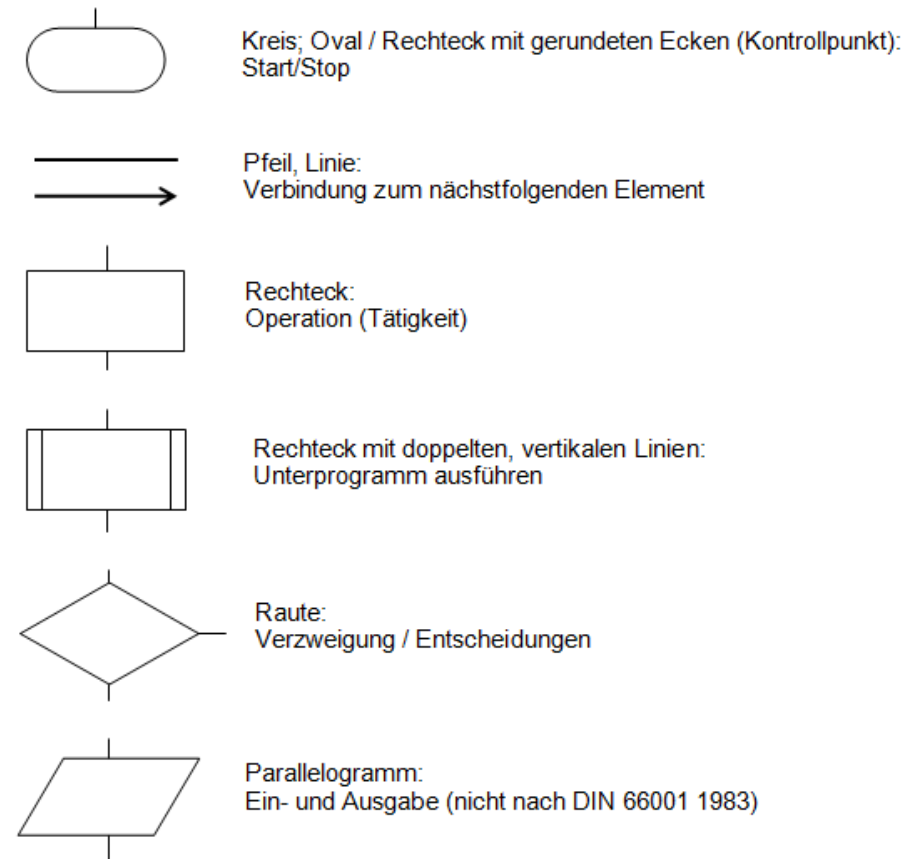
Quelle: Programmierung Grundlagen – mit Beispiel in Python, Ralph Steyer

Programmablaufplan (PAP)

- In der rechts dargestellten Abbildung sind einige der Symbole eines PAP dargestellt.
- Die Anwendung von PAP wird im Rahmen der Vorlesung nicht weiter vertieft.
- Vertiefende Informationen können Sie der DIN 66001 entnehmen.

Einen einfachen Editor finden Sie unter:

<https://www.diagrammeditor.de>



Quelle: <http://www.maschinenbau-wissen.de>

Pseudocode

- Unter Pseudocode versteht man eine halbformale, textuelle Beschreibung des Programmablaufs.
- Die textuelle Beschreibung erfolgt in Anlehnung an höhere Programmiersprachen.
- Für Kontrollstrukturen wird eine ähnliche Syntax wie in Programmiersprachen üblich verwendet, z.B. *if....then....else* oder *while*.
- Anweisungen können hierbei verbal oder in Anlehnung an Programmiersprachen formuliert werden, z.B. *erhöhe zahl i um 10*.

```
begin
  BetragPruefen
    Eingabe (a) ;
    Eingabe (b) ;
    if a >= b then
      Ausgabe (a) ;
    else
      Ausgabe (b) ;
    end if
  end BetragPruefen
```

Pseudocode

Quelle: Programmierung Grundlagen – mit Beispiel in Python, Ralph Steyer

Übung Programmablaufplan

Entwickeln Sie einen Programmablaufplan (PAP) für folgendes Problem:

Je nach Höhe des Umsatzes wird einem Kunden ein bestimmter Prozentsatz Rabatt gewährt. Ab 100 € Umsatz bekommt ein Kunde 5 % Rabatt. Ab 500 € bekommt der Kunde 10 % Rabatt. Das Programm soll den Rechnungsbetrag des Kunden abzüglich des Rabattes berechnen.