

Was ist das?

Eine Art Anleitung, um ein Problem/Aufgabe zu lösen.

Algorithmen können überall in unserem Alltag gefunden werden: zum Beispiel der Dijkstra-Algorithmus, der im Navi den kürzesten Weg zum Ziel findet und auch Ampeln werden mit Algorithmen gesteuert.

Es gibt aber auch sehr einfache Algorithmen wie zum Beispiel schauen, ob die Zahl A größer ist oder die Zahl B

Der Algorithmus würde so aussehen:

Wenn A größer ist als B:

Dann: Sagt man A

Sonst: Sagt man B

In Programmiersprache (hier Python) würde das so aussehen:

Code:

```
1 a = 1
2 b = 3
3 if a > b:
4     print("A ist größer als B")
5 else:
6     print("B ist größer als A")
```

Ausgabe:

```
B ist größer als A
```

Algorithmen werden in Funktionen geschrieben, damit sie immer wieder verwendet werden können. Funktionen beinhalten meist kleinere Algorithmen und wenn man diesen Algorithmus aufrufen möchte, muss man nur die Funktion aufrufen und nicht wieder den ganzen Algorithmus schreiben.

Autoren	Thema	Blattnummer:
Paul Raab, Clemens Fritschek, Tobias Larndorfer, Christoph Neulinger	Algorithmen	1

Das Beispiel von oben würde mit einer Funktion so aussehen:

Code:

```
a = 1
b = 3

def BiggerNumber(a,b):
    if(a>b):
        return "A ist größer als B"
    else:
        return "B ist größer als A"

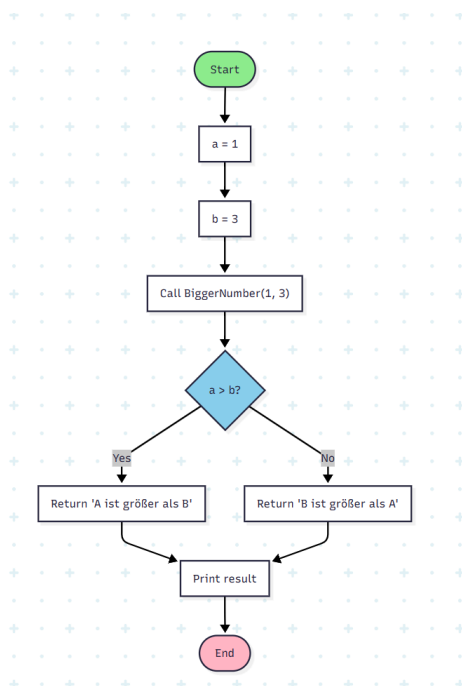
print(BiggerNumber(a,b))
```

Ausgabe:

```
B ist größer als A
```

Man kann diesen Algorithmus jetzt aber mehrmals verwenden und muss ihn nicht immer wieder schreiben, wenn man herausfinden will, welche Zahl größer ist.

Algorithmen kann man auch in Flowcharts darstellen um den Ablauf anschaulicher zu machen:



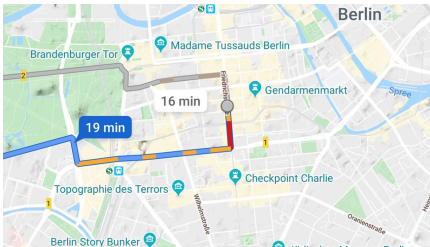
Autoren	Thema	Blattnummer:
Paul Raab, Clemens Fritschek, Tobias Larndorfer, Christoph Neulinger	Algorithmen	2

Wozu brauche ich das?

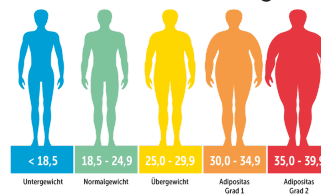
Algorithmen werden verwendet, um Probleme zu lösen, Daten zu verarbeiten und Entscheidungen zu treffen. Im modernen Alltag sind sie unverzichtbar, um komplexe Prozesse effizienter zu gestalten, große Datenmengen zu analysieren und automatisierte Ergebnisse zu liefern.

Konkrete Beispiele für Algorithmen

- **Navigationsysteme:** für jede gewünschte Strecke die richtige Route ermitteln



- **Sammeln von Nutzungsdaten (z.B. Klickverhalten):** Ein Algorithmus könnte durch die Frage „Erfolgt ein Klick eher auf das Werbebanner A oder das Werbebanner B oder auf keines von beiden?“ definiert sein. Auf Basis des tatsächlichen Klicks schneiden die Algorithmen zukünftige Werbebanner individuell auf die Nutzenden zu, basierend auf deren Interessen.
- **Grundlage für das Programmieren:** Wer Programmieren lernen will, stellt schnell fest, dass nahezu jedes Programm Algorithmen nutzt, um eine Aufgabe auszuführen. Leicht vereinfacht lässt sich sogar sagen, dass jedes Programm selbst ein Algorithmus ist.
- **Google-Algorithmus:** Wenn man einen Suchbegriff oder eine Suchanfrage in Google eingeben, dauert es nur einen Bruchteil einer Sekunde, bis man mehrere Seiten an Suchergebnissen erhält. Möglich macht das der Suchmaschinen-Algorithmus von Google. Dieser listet Ergebnisse nach Relevanz, Keywords, Lesbarkeit der Inhalte und Sicherheit auf.
- **Body-Mass-Index:** Der Körperfettanteil eines Menschen lässt sich durch den Body-Mass-Index berechnen. Hierbei wird das Körpergewicht in Relation zur Körpergröße gesetzt. Die Eingaben sind das aktuelle Gewicht und die Körpergröße, die Ausgabe wiederum ist der BMI. Der BMI gibt einen Anhaltspunkt dafür, ob Personen normalgewichtig, untergewichtig, übergewichtig oder fettleibig sind.



$$\text{BMI} = \frac{\text{Gewicht (kg)}}{\text{Grösse (m)}^2}$$

Autoren	Thema	Blattnummer:
Paul Raab, Clemens Fritschek, Tobias Larndorfer, Christoph Neulinger	Algorithmen	3

Was ist Programmieren?

Programmieren bedeutet, einem Computer genau zu sagen, was er tun soll. Dazu schreibt man Anweisungen, die der Computer Schritt für Schritt ausführen kann. Diese Anweisungen müssen sehr präzise sein. Wenn eine Anweisung unklar ist oder ein Fehler enthält, kann der Computer sie nicht richtig ausführen.

Die Anweisungen werden in sogenannten Programmiersprachen geschrieben. Das sind spezielle Sprachen, die so aufgebaut sind, dass Computer sie verstehen können, z.B. Python, C#, C++. Obwohl moderne Programme oft sehr groß und kompliziert wirken, bestehen sie im Grunde aus vielen einfachen Befehlen, die miteinander kombiniert werden.

Ein wichtiger Teil des Programmierens ist es, komplizierte Aufgaben in kleinere Teilaufgaben aufzuteilen. So kann man auch sehr komplexe Programme entwickeln.

Computer sind besonders gut darin, einfache Rechenoperationen extrem schnell auszuführen. Aufgaben wie Addieren, Vergleichen von Zahlen oder das Verschieben von Daten können sie Millionen oder Milliarden Mal pro Sekunde durchführen. Programme bestehen im Grunde aus vielen einfachen Operationen, die vom Computer sehr schnell abgearbeitet werden.

Grundsätzlich kann man einen Computer in zwei Bereiche einteilen: Hardware und Software.

Hardware bezeichnet alle physischen Teile eines Computers, also die Bauteile, die man anfassen kann. z.B:

- der Prozessor (CPU), der Berechnungen durchführt



- der Arbeitsspeicher (RAM), in dem Daten und Programme während der Ausführung gespeichert werden



- weitere Komponenten wie Festplatten, Grafikkarten oder Eingabegeräte



Software besteht aus Programmen und Abläufen, die auf der Hardware laufen. z.B:

- Betriebssystem (z.B. Windows oder Linux)
- Programme wie Browser, Spiele oder Word
- Apps auf dem Smartphone

Autoren	Thema	Blattnummer:
Paul Raab, Clemens Fritschek, Tobias Larndorfer, Christoph Neulinger	Algorithmen	4

Beim Programmieren arbeitet man an der Software. Man schreibt neue Programme oder verändert bestehende Programme, damit sie bestimmte Aufgaben erfüllen. Die Hardware bleibt dabei unverändert, sie führt nur die Befehle aus, die von der Software vorgegeben werden.

Wie kann ich das lernen

Programmieren ist eine Beschäftigung, die man hauptsächlich durch praktisches Üben und nicht nur durch Theorie lernt. Wie echte Sprachen, gibt es manche Programmiersprachen, die einfacher zu lernen und verstehen sind und andere wiederum, die komplizierter sind. Es gibt verschiedene Programmiersprachen, die für verschiedene Ziele verwendet werden.

Frontend (was man im Vordergrund sieht, z.B. Website):

- HTML
 - für Struktur und Inhalt
- CSS
 - Für Layout und Styling
- JavaScript
 - Animationen und Interaktivität

Backend (Der Code und die Algorithmen im Hintergrund):

- Python
 - Einfach zu lernen und verstehen
- C#
 - Komplex

Datenbank (Daten, wie Benutzername oder Geburtsdatum werden in einer Datenbank gespeichert):

- SQL
 - Datenbankverwaltung

Autoren	Thema	Blattnummer:
Paul Raab, Clemens Fritschek, Tobias Larndorfer, Christoph Neulinger	Algorithmen	5