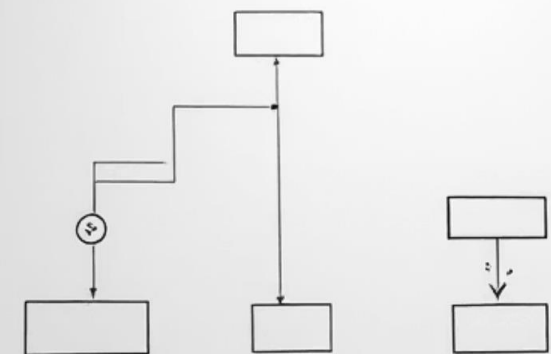


# Programmablaufplan (PAP)

Eine strukturierte visuelle Darstellung zur Beschreibung von Algorithmen und Prozessen in der Informatik

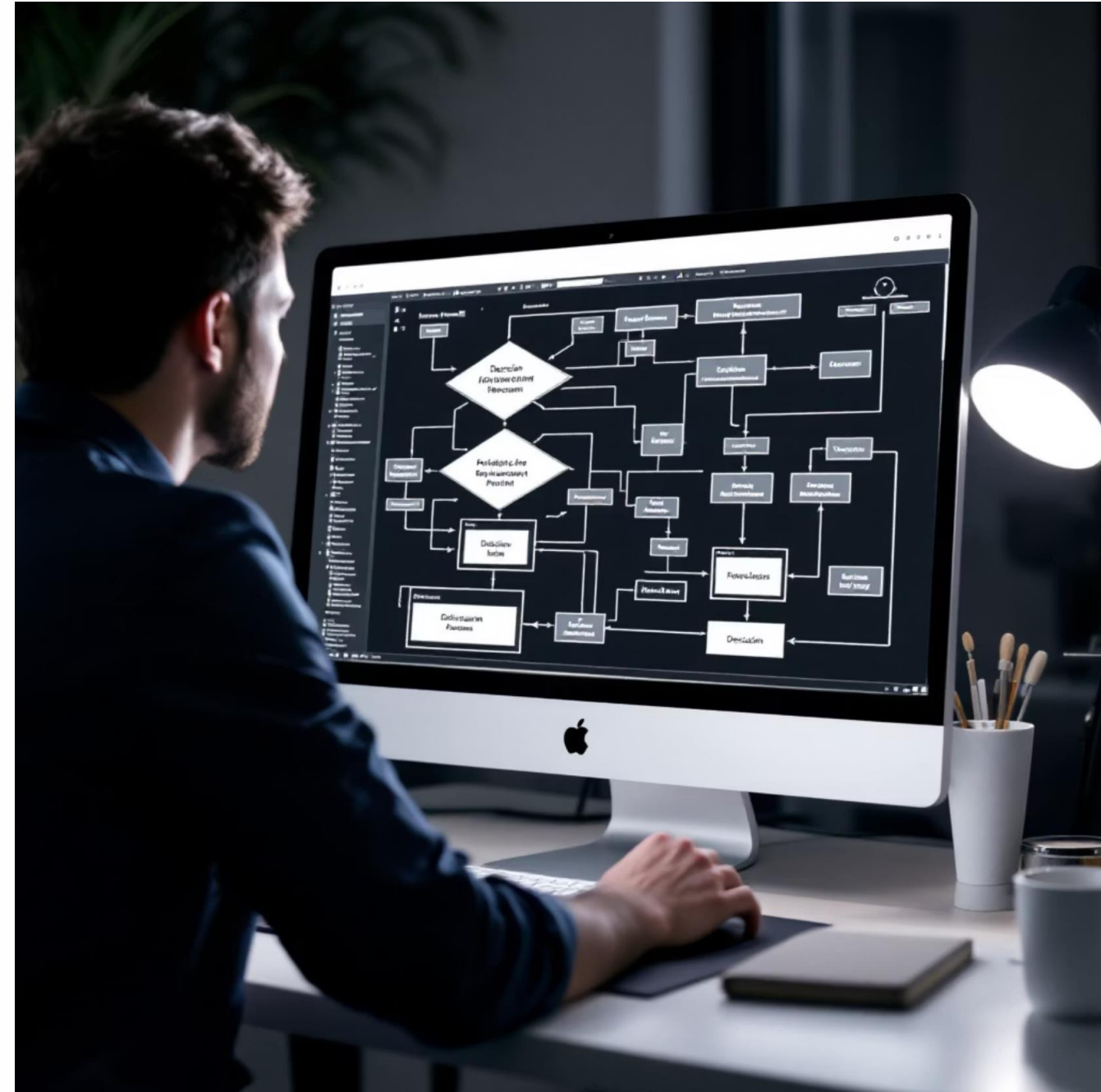


# Definition des Programmablaufplans

Ein Programmablaufplan (PAP) ist eine grafische Darstellung eines Algorithmus oder Prozesses mittels standardisierter Symbole, die durch Flusslinien verbunden sind.

Wesentliche Eigenschaften:

- Normierte Darstellung nach DIN 66001
- Eindeutige Ablaufrichtung (typischerweise von oben nach unten)
- Klare Visualisierung von Verzweigungen und Schleifen
- Elementare Operationen werden in Einzelschritten dargestellt



# Abgrenzung: PAP vs. Struktogramm

## Programmablaufplan (PAP)

- Freie zweidimensionale Anordnung
- Verwendung unterschiedlicher Symbole für verschiedene Operationen
- Verbindungslinien mit Pfeilen zeigen die Flussrichtung
- Normiert nach DIN 66001
- Mehr grafischer Aufwand in der Erstellung

## Struktogramm (Nassi-Shneiderman)

- Streng hierarchisch von oben nach unten
- Keine unterschiedlichen Symbole, nur rechteckige Blöcke
- Keine Verbindungslinien nötig
- Erzwingt strukturierte Programmierung
- Einfacher zu erstellen und zu modifizieren

# PAP-Symbole nach DIN 66001 (Teil 1)



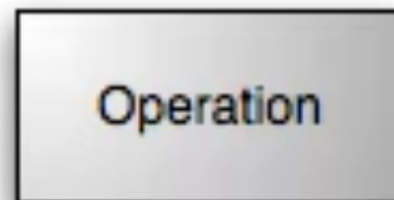
## Start/Stop

Dargestellt als abgerundetes Rechteck (Oval). Markiert den Beginn und das Ende eines Programmablaufs.



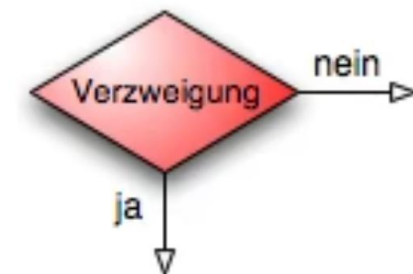
## Operation (Tätigkeit)

Dargestellt als Rechteck. Repräsentiert einen Verarbeitungsschritt, eine Berechnung oder eine Zuweisung.

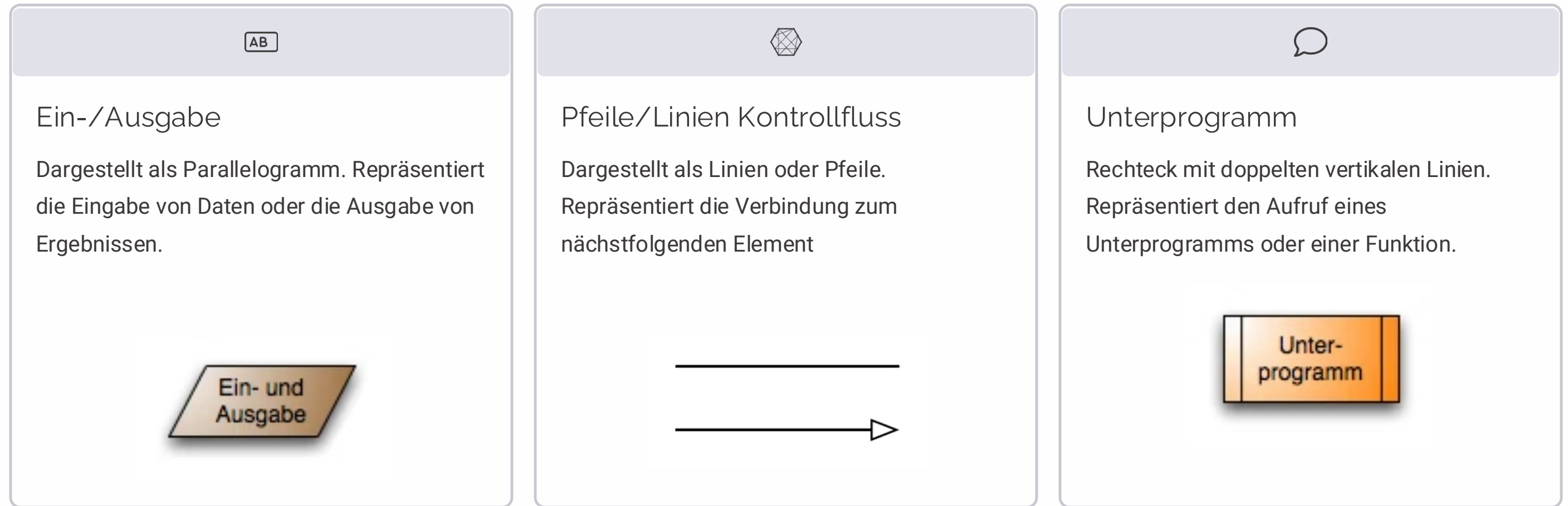


## Entscheidung

Dargestellt als Raute. Repräsentiert eine Verzweigung mit Ja/Nein-Ausgängen oder mehreren alternativen Pfaden.



# PAP-Symbole nach DIN 66001 (Teil 2)



Flusslinien

Eindeutige Richtungsangabe durch Pfeile

Horizontale und vertikale Linienführung  
bevorzugt

Kreuzungen vermeiden oder deutlich  
kennzeichnen

# Beispiel: Umsetzung PAP

## 1 Initialisierung

Die Zählvariable  $i$  wird mit dem Startwert 1 initialisiert.

## 2 Schleifenbeginn

In jedem Durchlauf wird zuerst der aktuelle Wert von  $i$  ausgegeben.

## 3 Bedingungsprüfung

Es folgt eine Verzweigung:

- Wenn  $i$  gleich 39 ist, wird  $i$  auf 61 gesetzt. Danach beginnt der nächste Schleifen- durchlauf.
- Wenn  $i$  nicht 39 ist, wird  $i$  um 1 erhöht.

## 4 Schleifenfortsetzung

Nach der Erhöhung von  $i$  wird geprüft, ob  $i$  kleiner oder gleich 100 ist.

- Wenn ja, beginnt ein neuer Schleifen- durchlauf.
- Wenn nein, wird die Schleife beendet.

## 5 Ausgabe

Das Programm gibt alle natürlichen Zahlen von 1 bis 39 und von 61 bis 100 aus, wobei 39 und 100 eingeschlossen sind.

