

Programmieren 1

Python Grundlagen mit TurtleGraphics – Informatik 1M

Verzweigungen

A lush green forest with two paths leading into the distance. The forest is dense with tall, thin trees and a thick carpet of ferns on the ground. The paths are made of dirt and lead towards a bright light at the end of the forest. The overall atmosphere is serene and natural.

Schleifen

A lush, misty forest scene. A dirt path winds through the trees, forming a circular loop around a large, moss-covered tree stump in the center. The trees are tall and thin, with dense green foliage. The ground is covered in fallen leaves and moss. The overall atmosphere is serene and magical.

Verzweigungen

A lush green forest scene with two dirt paths leading into the distance. The forest is dense with tall, thin trees and a thick carpet of ferns on the ground. The lighting is soft and dappled, creating a serene atmosphere. The text 'Verzweigungen' is overlaid in white on the left side of the image.

Der Aufbau einer if-else-Verzweigung

```
if BEDINGUNG:  
    # code für den Fall, dass die Bedingung WAHR ist.  
else:  
    # code für den Fall, dass die Bedingung FALSCH ist.
```

In der **Bedingung** wird oft eine Variable mit einem Wert *vergleichen*.
Hier einige Beispiele:

```
if geheimzahl == 2: # wahr, wenn geheimzahl 2 ist.  
if resultat > 4: # wahr, wenn resultat grösser 4 ist.  
if geheimzahl != 2: # wahr, wenn geheimzahl nicht 2 ist.  
if antwort == "ja": # wahr, wenn antwort "ja" ist.
```

Vergleichsoperatoren

Bisher nutzen wir den Zuweisungsoperator und die klassischen Operatoren zum Rechnen – Beispiel:

```
zahl = 27 # Der Variable zahl wird der Wert 27 zugewiesen.  
zahl = zahl * 2 # Der Wert in zahl wird verdoppelt.  
zahl = zahl + 6 # Der Wert in zahl wird um 6 vergrössert.  
print(zahl) # Der Wert von zahl wird ausgegeben.
```

Für Vergleiche (in Bedingungen) nutzen wir Vergleichsoperatoren:

```
x == y: # wahr, wenn x gleich y.  
x > y: # wahr, wenn x grösser y.  
x >= y: # wahr, wenn x grösser oder gleich y.  
x != y: # wahr, wenn x ungleich y.
```

Beispiel: Auf Eingabe von User reagieren

```
antwort = input("Wie viele Seiten hat ein Dreieck")
if antwort == 3:
    print("Wie schlau von dir!")
else:
    print("Wie dumm von dir!")
```

- Die Funktion **input(text)** zeigt eine Meldung, die den **text** und ein Eingabefeld enthält.
- Die Funktion **input(text)** gibt zurück, was von der Benutzer:in eingegeben wurde.
- Was die Funktion zurückgibt, wird in der Variable **antwort** gespeichert.
- Die Variable **antwort** wird in der if-Bedingung **vergleichen**.
- Je nach Resultat wird eine andere Meldung mit **print(text)** ausgegeben.

A misty forest scene with a path that splits into two. The trees are tall and thin, with a dense canopy of green leaves. The ground is covered in fallen leaves and moss. The overall atmosphere is serene and mysterious. The text "Mehr als zwei Zweige" is overlaid in the center of the image.

Mehr als zwei Zweige

Mehr als zwei Zweige

```
antwort = input("Was ist die kleinste Primzahl?")
if antwort == 1:
    print("1 ist keine Primzahl")
elif antwort == 2:
    print("Richtig! Sehr gut.")
elif antwort == 3:
    print("Fast, es gibt noch eine kleinere Primzahl.")
else:
    print("Falsch!")
```

- Es können beliebig viele elif-Zweige eingefügt werden.
- Sobald eine Bedingung wahr ist, wird der entsprechende Code ausgeführt und der Rest der if-elif-else-Verzweigung übersprungen.

Programmieren mit Verzweigungen

- Öffne das [Wiki](#).
- Studiere die Beispiele im Kapitel Verzweigungen.
- Löse die Aufgaben E.

If-Zweig ohne else

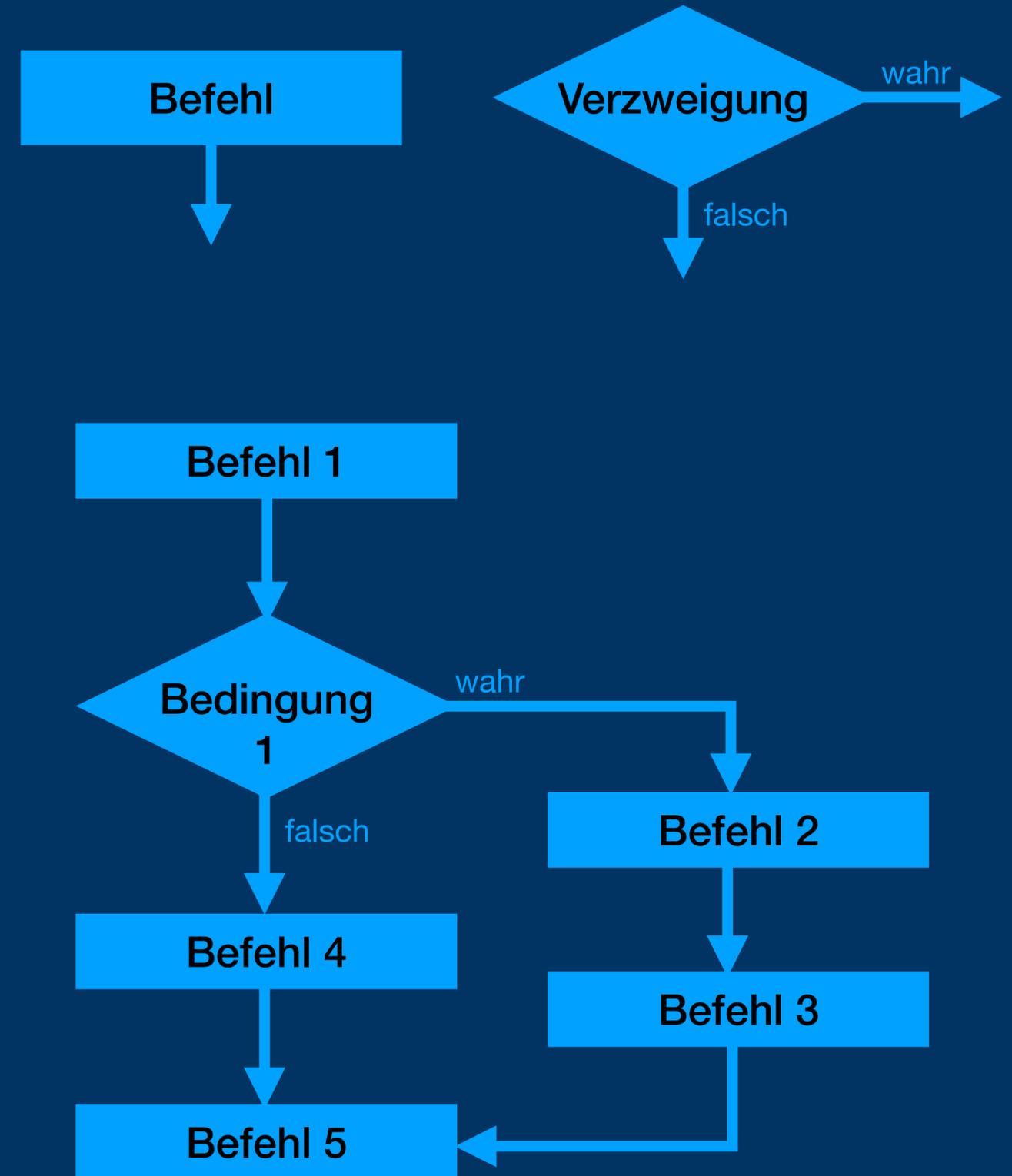
```
from gturtle import *
donatello = Turtle()
seite = 50
repeat 10:
    seite = seite + 10
    if seite > 100:
        donatello.setPenColor("blue")
    donatello.forward(seite)
    donatello.left(90)
```

- Die Schildkröte zeichnet eine quadratische Spirale. Ab einer bestimmten Seitenlänge wechselt die Farbe auf blau.

Codeflow – Beispiel 1

- Zeichne folgenden Code als Flussdiagramm: Befehle zeichnest du als Rechtecke, Verzweigungen als Rauten

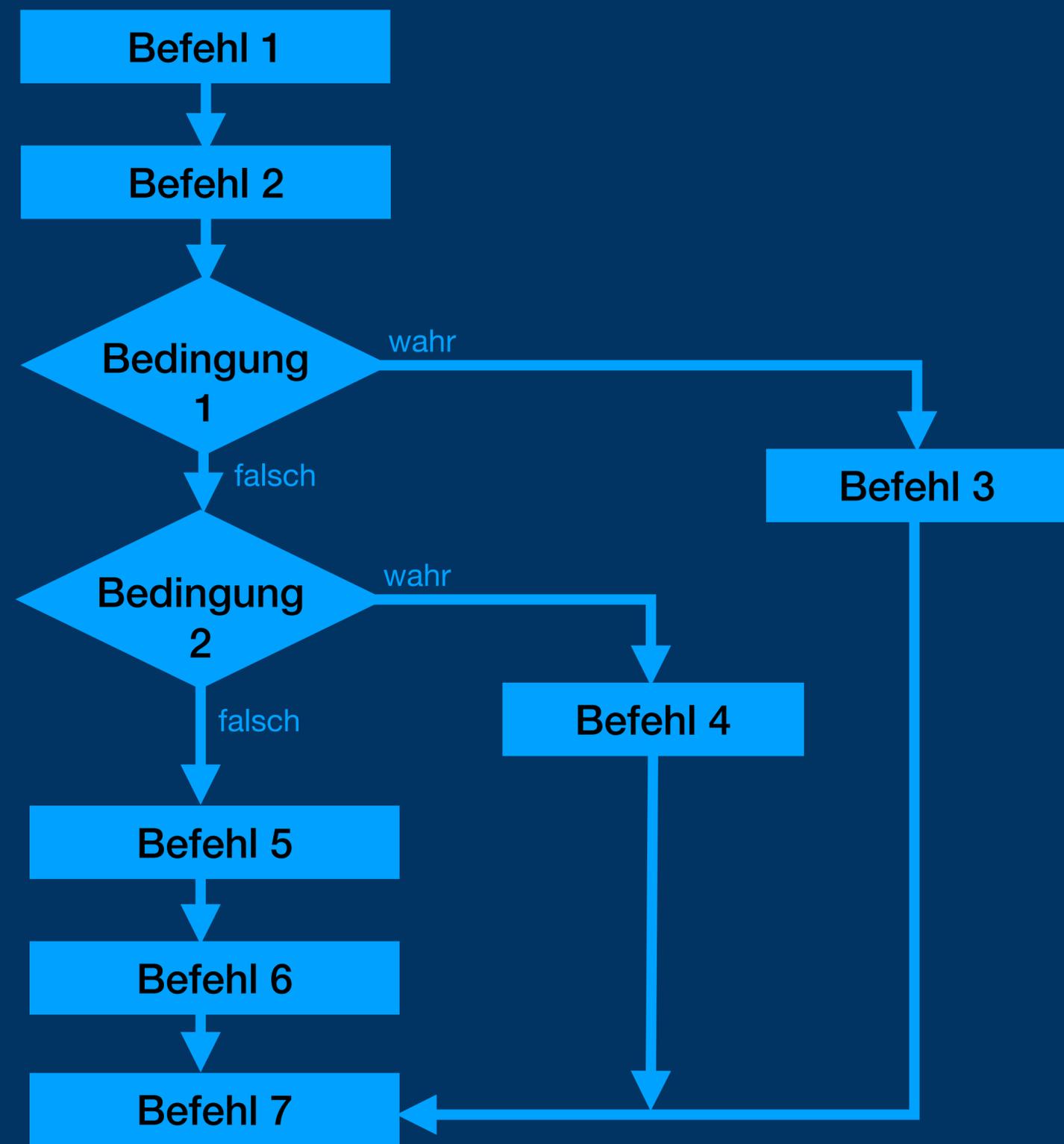
```
Befehl 1  
if Bedingung1:  
    Befehl 2  
    Befehl 3  
else:  
    Befehl 4  
Befehl5
```



Codeflow – Beispiel 2

- Zeichne folgenden Code als Flussdiagramm:

```
Befehl 1  
Befehl 2  
if Bedingung 1:  
    Befehl 3  
elif Bedingung2:  
    Befehl 4  
else:  
    Befehl 5  
    Befehl 6  
Befehl 7
```



- Wenn Bedingung 1 wahr ist, wird nicht mehr gefragt, ob auch Bedingung 2 wahr ist. Sobald eine Bedingung wahr ist, werden alle anderen Bedingungen übersprungen.