

1. Flussdiagramme

- Schleifen in Flussdiagrammen erkennen und ihren Ablauf nachvollziehen.
- Die Ausführung einer Schleife Schritt für Schritt verfolgen und die resultierenden Ausgaben bestimmen.
- Erkennen, ob eine Schleife terminiert oder sich endlos wiederholt.

2. *for*-Schleifen mit Zählvariable

- Die Funktionsweise von `range()` verstehen und geeignete Parameter für vorgegebene Zahlenfolgen wählen.
- Schleifen mit auf- und absteigenden Zählvariablen nachvollziehen.
- Den Einfluss von Zuweisungen und arithmetischen Ausdrücken auf den Schleifenablauf erklären.
- Die Ausgabe eines Programms mit einer `for`-Schleife ermitteln.

3. Listengesteuerte *for*-Schleifen

- Die Elemente einer Liste mithilfe einer `for`-Schleife durchlaufen.
- Die Ausgabe von Programmen mit listenbasierten Schleifen bestimmen.
- Den Unterschied zwischen einer listengesteuerten Schleife und einer Schleife mit `range()` erläutern.

4. *while*-Schleifen

- Die Funktionsweise einer kopfgesteuerten `while`-Schleife erklären.
- Die Schleifenbedingung auswerten und feststellen, ob ein weiterer Schleifendurchlauf stattfindet.
- Die Veränderung von Variablen bei der Schleifenausführung nachvollziehen.
- Die Ausgaben bestimmen.
- Erkennen, ob eine Schleife regulär beendet wird oder sich endlos wiederholt.

5. *break* und *continue* in *for*- und *while*-Schleifen

- Die Wirkung von `break` und `continue` erklären.
- Ermitteln, welche Schleifendurchläufe vollständig, vorzeitig oder gar nicht ausgeführt werden.
- Bestimmen, welche Werte ausgegeben werden und welche Ausgaben aufgrund von `break` oder `continue` entfallen.
- Die Position von `break` und `continue` im Programmablauf berücksichtigen und ihre Wirkung in den beiden Schleifentypen nachvollziehen.
- Den Unterschied zwischen dem Abbruch eines Schleifendurchlaufs und dem Beenden der gesamten Schleife erklären.
- Erkennen, ob durch `break` oder `continue` eine Endlosschleife entstehen kann.

6. *Endlosschleifen*

- Erkennen, ob eine Schleife tatsächlich terminiert.
- Untersuchen, ob eine Schleifenbedingung jemals den Wert **False** annimmt.
- Erkennen, dass eine Schleife einen bestimmten Vergleichswert verfehlen kann und deshalb endlos weiterläuft.

7. *Theoriefragen zu Schleifen*

- Die Funktionsweise von **for**- und **while**-Schleifen beschreiben.
- Den Unterschied zwischen kopf- und fussgesteuerten Schleifen erklären.
- Erläutern, wie sich eine fussgesteuerte Schleife mithilfe anderer Sprachkonstrukte nachbilden lässt.
- Die Wirkung von **break** und **continue** erklären und voneinander unterscheiden.

8. *Schleifen selbst programmieren*

- Eine Problemstellung in einen geeigneten Schleifenalgorithmus umsetzen.
- Zwischen **for**- und **while**-Schleifen wählen und die Wahl begründen.
- Schleifen mit geeigneten Startwerten, Abbruchbedingungen und Schrittweiten formulieren.
- Zahlenfolgen erzeugen und Werte innerhalb einer Schleife verarbeiten.
- Berechnungen mit Schleifen durchführen und Ergebnisse ausgeben.
- Schleifen korrekt einrücken und syntaktisch korrekt formulieren.
- Den Programmablauf anhand der Aufgabenstellung überprüfen und typische Fehler vermeiden.