

Kontrollstrukturen

3.1 Übersicht

Kontrollstrukturen sind grundlegende Konzepte in der Programmierung, die verwendet werden, um den Ablauf eines Programms zu steuern und Entscheidungen zu treffen. Es gibt folgende Kontrollstrukturen:

- Sequenz: Dies ist die einfachste Art von Kontrollstruktur. Die Anweisungen werden nacheinander, in der Reihenfolge, in der sie im Programm auftreten, ausgeführt.
- Selektion (Verzweigung): Hierbei wird eine Bedingung überprüft und basierend auf dem Ergebnis werden unterschiedliche Codeblöcke ausgeführt (if, else).
- Iteration (Schleifen): Mit Schleifen kannst du bestimmte Codeblöcke wiederholt ausführen, entweder eine festgelegte Anzahl von Malen oder bis eine Bedingung erfüllt ist (for, while).

3.2 Verzweigungen

3.2.1 if-elif-else

Das if-elif-else-Statement ist eine grundlegende Kontrollstruktur in der Programmierung, die verwendet wird, um Entscheidungen zu treffen und den Codepfad basierend auf verschiedenen Bedingungen zu steuern. Es ermöglicht Ihnen, unterschiedliche Codeblöcke auszuführen, abhängig davon, welche Bedingung wahr ist.

Allgemeine Syntax:

```
1     if Bedingung_1:
2         # Code, der ausgeführt wird, wenn Bedingung_1 wahr ist
3     elif Bedingung_2:
4         # Code, der ausgeführt wird, wenn Bedingung_2 wahr ist
5     elif Bedingung_3:
6         # Code, der ausgeführt wird, wenn Bedingung_3 wahr ist
7     # Weitere "elif" Blöcke können hinzugefügt werden
8     else:
9         # Code, der ausgeführt wird, wenn keine der Bedingungen wahr ist
```

Listing 3.1: if-elif-else (Syntax)

”elif” und ”else” sind optional.
Hier ein Beispiel:

```
1     score = 85
2
3     if score >= 90:
4         print("Sehr gut")
5     elif score >= 80:
6         print("Gut")
7     elif score >= 70:
8         print("Befriedigend")
9     else:
10        print("Nicht bestanden")
```

Listing 3.2: if-elif-else

Ergebnis

Gut

3.2.2 Short Hand If

```
1     a = 14
2     b = 11
3
4     if a > b: print("a ist grösser als b")
```

Listing 3.3: Short Hand If

Ergebnis

a ist grösser als b

- Man beachte den Doppelpunkt.
- Diese Form wird nicht als schöner Stil betrachtet.

3.2.3 Verschaltetes if

```
1 x = 10
2
3 if x > 5:
4     print("x ist grösser als 5")
5     if x > 8:
6         print("x ist auch grösser als 8")
7 else:
8     print("x ist nicht grösser als 5")
```

Listing 3.4: Verschaltetes if

Ergebnis

```
x ist grösser als 5
x ist auch grösser als 8
```

3.2.4 pass

Das folgende Listing generiert einen Fehler:

```
1 x = 10
2
3 if x > 5:
4
5
6 y = 20
```

Listing 3.5: Ohne pass

Wenn man während der Entwicklung noch keinen Code für die erfüllte Bedingung zur Verfügung hat, kann man "pass" verwenden um den Code lauffähig zu machen.

```
1 x = 10
2
3 if x > 5:
4     pass
5
6 y = 20
```

Listing 3.6: pass