

# Übungsaufgaben: Komplexe Struktogramme

## Problemstellung 1: Login-Versuche

**Situation:** Ein Benutzer soll sich mit einem Passwort anmelden. Aus Sicherheitsgründen hat er dafür aber nur **drei Versuche**.

**Aufgabe:** Entwirf einen Algorithmus, der den Benutzer nach einem Passwort fragt. Wenn das Passwort falsch ist, soll die Anzahl der verbleibenden Versuche ausgegeben und die Abfrage wiederholt werden. Wenn das Passwort korrekt ist oder die drei Versuche aufgebraucht sind, soll das Programm mit einer entsprechenden Erfolgs- oder Fehlermeldung beendet werden. Das korrekte Passwort lautet "Admin123".

---

## Problemstellung 2: Notendurchschnitt mit Validierung

**Situation:** Ein Lehrer möchte den Notendurchschnitt seiner Klasse berechnen. Er möchte die Noten nacheinander eingeben. Das Programm soll dabei ungültige Noten (kleiner als 1 oder größer als 6) erkennen und ignorieren. Die Eingabe soll beendet werden, wenn der Lehrer die Zahl 0 eingibt.

**Aufgabe:** Erstelle ein Struktogramm für ein Programm, das Noten entgegennimmt, bis eine 0 eingegeben wird. Ungültige Noten sollen mit einer Fehlermeldung übersprungen werden. Am Ende soll der Durchschnitt der gültigen Noten berechnet und ausgegeben werden. Es muss auch der Fall bedacht werden, dass gar keine gültige Note eingegeben wurde.

---

## Problemstellung 3: Einfacher Warenkorb

**Situation:** Ein Kunde in einem kleinen Online-Shop legt Artikel in seinen Warenkorb, indem er nacheinander deren Preise eingibt. Wenn er fertig ist, gibt er eine 0 ein, um zum Bezahlvorgang zu gelangen. Der Shop bietet einen Rabatt von 10% an, wenn der Gesamtwert aller Artikel 100€ übersteigt.

**Aufgabe:** Modelliere den gesamten Prozess als Struktogramm. Der Algorithmus soll Preise sammeln, bis die Eingabe beendet wird. Anschließend soll er prüfen, ob ein Rabatt gewährt wird, den Endpreis berechnen und eine finale Abrechnung (Zwischensumme, Rabatt, Endpreis) ausgeben.