

Übungsaufgaben: Planung & Pseudocode

Deine Aufgabe ist es, für jedes Szenario eine **Top-Down-Zerlegung** des Problems zu erstellen und den daraus resultierenden Algorithmus als **Pseudocode** zu formulieren.

Szenario 1: Pfandberechnung

Situation: Ein Supermarkt bekommt einen neuen Leergutautomaten. Der Automat soll für zwei Arten von Flaschen den korrekten Pfandbetrag berechnen. PET-Flaschen (Einweg) sind 0,25 € wert, Glasflaschen (Mehrweg) sind 0,08 € wert. **Aufgabe:** Erstelle einen Plan (Zerlegung & Pseudocode), der den Benutzer nach der Anzahl der jeweiligen Flaschen fragt und den Gesamtpfandbetrag berechnet und ausgibt.

Szenario 2: Ticketpreis-Kalkulator

Situation: Ein Kino möchte die Preise für seine Tickets automatisch berechnen. Erwachsene zahlen 12 €. Kinder (unter 16) und Senioren (ab 65) zahlen einen ermäßigten Preis von 9 €. **Aufgabe:** Erstelle einen Plan (Zerlegung & Pseudocode), der das Alter des Kunden abfragt und den korrekten Ticketpreis ermittelt und ausgibt.

Szenario 3: Bonuspunkte-System

Situation: Ein Online-Shop hat ein Bonuspunkte-System. Für jeden Euro Einkaufswert gibt es einen Bonuspunkt. Bei Einkäufen über 100 € werden die Punkte für den gesamten Einkauf verdoppelt. Premium-Mitglieder erhalten zusätzlich pauschal 50 Punkte gutgeschrieben. **Aufgabe:** Erstelle einen Plan (Zerlegung & Pseudocode) für die Berechnung der finalen Bonuspunkte.

Szenario 4: Versandkosten-Rechner

Situation: Ein Paketdienst berechnet seine Kosten nach dem Gewicht der Sendung. Sendungen bis 2kg kosten 5€. Sendungen von über 2kg bis 5kg kosten 8€. Alles, was schwerer als 5kg ist, kostet 12€. **Aufgabe:** Erstelle einen Plan (Zerlegung & Pseudocode), der das Gewicht eines Pakets entgegennimmt und die anfallenden Versandkosten ausgibt.

Szenario 5: Passwort-Validierung

Situation: Bei der Registrierung auf einer Webseite muss das Passwort des Benutzers validiert werden. Die Regel lautet: Das Passwort muss mindestens 8 Zeichen lang sein UND mindestens eines der folgenden Sonderzeichen enthalten: **!**, **?**, **#**. **Aufgabe:** Erstelle einen Plan (Zerlegung & Pseudocode), der ein Passwort prüft und ausgibt, ob es gültig ist oder nicht.

Lösungen

Lösung zu Szenario 1: Pfandberechnung

Top-Down-Zerlegung:

- **Ebene 1:** Gesamtpfand für eine Leergut-Abgabe berechnen.
- **Ebene 2:**
 - Teilproblem 2.1: Anzahl der PET-Flaschen und Glasflaschen einlesen.
 - Teilproblem 2.2: Pfandwert für jede Flaschensorte berechnen.
 - Teilproblem 2.3: Einzelne Pfandwerte zum Gesamtpfand addieren.
 - Teilproblem 2.4: Gesamtpfand ausgeben.

Pseudocode:

```
START
  EINGABE von anzahl_pet
  EINGABE von anzahl_glas

  SETZE pfand_pet AUF anzahl_pet * 0.25
  SETZE pfand_glas AUF anzahl_glas * 0.08

  SETZE gesamtpfand AUF pfand_pet + pfand_glas

  AUSGABE von "Ihr Gesamtpfand beträgt: " + gesamtpfand + " €"
ENDE
```

Code-Umsetzung:

```
# Python
anzahl_pet = 10
anzahl_glas = 5
pfand_pet = anzahl_pet * 0.25
pfand_glas = anzahl_glas * 0.08
gesamtpfand = pfand_pet + pfand_glas
print(f"Ihr Gesamtpfand beträgt: {gesamtpfand:.2f} €")
```

```
// JavaScript
let anzahl_pet = 10;
let anzahl_glas = 5;
let pfand_pet = anzahl_pet * 0.25;
let pfand_glas = anzahl_glas * 0.08;
let gesamtpfand = pfand_pet + pfand_glas;
console.log(`Ihr Gesamtpfand beträgt: ${gesamtpfand.toFixed(2)} €`);
```

```
// Java
int anzahl_pet = 10;
int anzahl_glas = 5;
double pfand_pet = anzahl_pet * 0.25;
double pfand_glas = anzahl_glas * 0.08;
double gesamtpfand = pfand_pet + pfand_glas;
System.out.printf("Ihr Gesamtpfand beträgt: %.2f €\n", gesamtpfand);
```

Lösung zu Szenario 2: Ticketpreis-Kalkulator

Top-Down-Zerlegung:

- **Ebene 1:** Korrekten Kinoticket-Preis ermitteln.
- **Ebene 2:**
 - Teilproblem 2.1: Alter des Kunden einlesen.
 - Teilproblem 2.2: Altersgruppe des Kunden bestimmen und Preis zuweisen.
 - Teilproblem 2.3: Preis ausgeben.
- **Ebene 3 (Zerlegung von 2.2):**
 - Prüfe, ob das Alter unter 16 ODER 65 und älter ist.

Pseudocode:

```
START
  EINGABE von alter
  SETZE preis AUF 0

  WENN alter < 16 ODER alter >= 65 DANN
    SETZE preis AUF 9.00
  SONST
    SETZE preis AUF 12.00
  ENDEWENN

  AUSGABE von "Der Ticketpreis beträgt: " + preis + " €"
ENDE
```

Code-Umsetzung:

```
# Python
alter = 25
preis = 0
if alter < 16 or alter >= 65:
    preis = 9.00
else:
    preis = 12.00
print(f"Der Ticketpreis beträgt: {preis:.2f} €")
```

```
// JavaScript
let alter = 25;
let preis = 0;
if (alter < 16 || alter >= 65) {
    preis = 9.00;
} else {
    preis = 12.00;
}
console.log(`Der Ticketpreis beträgt: ${preis.toFixed(2)} €`);
```

```
// Java
int alter = 25;
double preis = 0;
if (alter < 16 || alter >= 65) {
    preis = 9.00;
} else {
    preis = 12.00;
}
System.out.printf("Der Ticketpreis beträgt: %.2f €\n", preis);
```



Lösung zu Szenario 3: Bonuspunkte-System

Top-Down-Zerlegung:

- **Ebene 1:** Bonuspunkte für einen Einkauf berechnen.
- **Ebene 2:**
 - Teilproblem 2.1: Notwendige Daten einlesen (Einkaufswert, Premium-Status).
 - Teilproblem 2.2: Basispunkte berechnen.
 - Teilproblem 2.3: Möglichen Verdopplungs-Bonus anwenden.
 - Teilproblem 2.4: Möglichen Premium-Bonus anwenden.
 - Teilproblem 2.5: Gesamtpunkte ausgeben.

Pseudocode:

```
START
  EINGABE von einkaufswert
  EINGABE von ist_premium

  SETZE bonuspunkte AUF RUNDE_AB(einkaufswert)

  WENN einkaufswert > 100 DANN
    SETZE bonuspunkte AUF bonuspunkte * 2
  ENDEWENN

  WENN ist_premium == "ja" DANN
    SETZE bonuspunkte AUF bonuspunkte + 50
  ENDEWENN

  AUSGABE von "Sie erhalten " + bonuspunkte + " Bonuspunkte."
ENDE
```

Code-Umsetzung:

```
# Python
einkaufswert = 120.50
ist_premium = "ja"
bonuspunkte = int(einkaufswert) # int() rundet ab
if einkaufswert > 100:
    bonuspunkte = bonuspunkte * 2
if ist_premium == "ja":
    bonuspunkte = bonuspunkte + 50
print(f"Sie erhalten {bonuspunkte} Bonuspunkte.")
```

```
// JavaScript
let einkaufswert = 120.50;
let ist_premium = "ja";
let bonuspunkte = Math.floor(einkaufswert);
if (einkaufswert > 100) {
    bonuspunkte *= 2; // Kurzschreibweise für bonuspunkte = bonuspunkte * 2
}
if (ist_premium === "ja") {
    bonuspunkte += 50;
}
console.log(`Sie erhalten ${bonuspunkte} Bonuspunkte.`);
```

```
// Java
double einkaufswert = 120.50;
```

```
String ist_premium = "ja";
int bonuspunkte = (int) einkaufswert;
if (einkaufswert > 100) {
    bonuspunkte *= 2;
}
if (ist_premium.equals("ja")) {
    bonuspunkte += 50;
}
System.out.println("Sie erhalten " + bonuspunkte + " Bonuspunkte.");
```

Lösung zu Szenario 4: Versandkosten-Rechner

Top-Down-Zerlegung:

- **Ebene 1:** Versandkosten für ein Paket berechnen.
- **Ebene 2:**
 - Teilproblem 2.1: Paketgewicht einlesen.
 - Teilproblem 2.2: Gewicht prüfen und Kosten zuweisen.
 - Teilproblem 2.3: Kosten ausgeben.
- **Ebene 3 (Zerlegung von 2.2):**
 - Prüfe, ob Gewicht ≤ 2 kg.
 - Wenn nicht, prüfe, ob Gewicht ≤ 5 kg.
 - Wenn nicht, muss es in die teuerste Kategorie fallen.

Pseudocode:

```
START
  EINGABE von gewicht
  SETZE kosten AUF 0

  WENN gewicht <= 2 DANN
    SETZE kosten AUF 5
  SONST WENN gewicht <= 5 DANN
    SETZE kosten AUF 8
  SONST
    SETZE kosten AUF 12
  ENDEWENN

  AUSGABE von "Die Versandkosten betragen: " + kosten + " €"
ENDE
```

Code-Umsetzung:

```
# Python
gewicht = 3.5
kosten = 0
if gewicht <= 2:
    kosten = 5
elif gewicht <= 5:
    kosten = 8
else:
    kosten = 12
print(f"Die Versandkosten betragen: {kosten} €")
```

```
// JavaScript
let gewicht = 3.5;
let kosten = 0;
if (gewicht <= 2) {
    kosten = 5;
} else if (gewicht <= 5) {
    kosten = 8;
} else {
    kosten = 12;
}
console.log(`Die Versandkosten betragen: ${kosten} €`);
```

```
// Java
double gewicht = 3.5;
int kosten = 0;
if (gewicht <= 2) {
    kosten = 5;
} else if (gewicht <= 5) {
    kosten = 8;
} else {
    kosten = 12;
}
System.out.println("Die Versandkosten betragen: " + kosten + " €");
```

Lösung zu Szenario 5: Passwort-Validierung

Top-Down-Zerlegung:

- **Ebene 1:** Passwort auf Gültigkeit prüfen.
- **Ebene 2:**
 - Teilproblem 2.1: Passwort einlesen.
 - Teilproblem 2.2: Passwort auf Bedingungen prüfen.
 - Teilproblem 2.3: Ergebnis der Prüfung ausgeben.
- **Ebene 3 (Zerlegung von 2.2):**
 - Prüfe, ob die Länge mindestens 8 ist.
 - Prüfe, ob eines der Sonderzeichen `!`, `?` oder `#` enthalten ist.
 - Beide Bedingungen müssen zutreffen.

Pseudocode:

```
START
  EINGABE von passwort

  SETZE laenge AUF die Anzahl der Zeichen in passwort
  SETZE hat_sonderzeichen AUF FALSCH

  WENN passwort enthält "!" ODER passwort enthält "?" ODER passwort enthält "#" DANN
    SETZE hat_sonderzeichen AUF WAHR
  ENDEWENN

  WENN laenge >= 8 UND hat_sonderzeichen == WAHR DANN
    AUSGABE von "Passwort ist gültig."
  SONST
    AUSGABE von "Passwort ist ungültig."
  ENDEWENN

ENDE
```

Code-Umsetzung:

```
# Python
passwort = "Sicher!123"
laenge = len(passwort)
hat_sonderzeichen = "!" in passwort or "?" in passwort or "#" in passwort
if laenge >= 8 and hat_sonderzeichen:
    print("Passwort ist gültig.")
else:
    print("Passwort ist ungültig.")
```

```
// JavaScript
let passwort = "Sicher!123";
let laenge = passwort.length;
let hat_sonderzeichen = passwort.includes("!") || passwort.includes("?") ||
passwort.includes("#");
if (laenge >= 8 && hat_sonderzeichen) {
    console.log("Passwort ist gültig.");
} else {
    console.log("Passwort ist ungültig.");
}
```

```
// Java
String passwort = "Sicher!123";
int laenge = passwort.length();
boolean hat_sonderzeichen = passwort.contains("!") || passwort.contains("?") ||
passwort.contains("#");
if (laenge >= 8 && hat_sonderzeichen) {
    System.out.println("Passwort ist gültig.");
} else {
    System.out.println("Passwort ist ungültig.");
}
```