

Python Musterprüfung – Grundlagen & Logik

Zeit: 45 Minuten

Punkte: 40

Hilfsmittel: keine

Aufgabe 1 – Code verstehen (10 Punkte)

```
def pruefe_werte(werte):
    ergebnis = []

    for w in werte:
        if w % 2 == 0:
            ergebnis.append(w * 2)
        else:
            ergebnis.append(w)

    return tuple(ergebnis)

zahlen = [1, 2, 3, 4]
resultat = pruefe_werte(zahlen)
print(resultat)
```

- a) Beschreibe, was die Funktion macht. (5 P)
- b) Nenne die Datentypen von *werte*, *ergebnis* und dem Rückgabewert. (3 P)
- c) Was wird ausgegeben? (2 P)

Aufgabe 2 – while-Schleife (8 Punkte)

```
def sammle_positive(daten):
    i = 0
    result = set()

    while i < len(daten):
        if daten[i] > 0:
            result.add(daten[i])
        i += 1

    return result

zahlen = [3, -1, 3, 5, 0, -2]
print(sammle_positive(zahlen))
```

- a) Warum wird ein Set verwendet? (3 P)
- b) Welche Werte können im Ergebnis vorkommen? (3 P)
- c) Was passiert, wenn *i += 1* fehlt? (2 P)

Aufgabe 3 – Dictionary & items() (10 Punkte)

```
def berechne_noten(daten):
    result = {}

    for name, noten in daten.items():
        summe = 0
        anzahl = 0
```

```

    for n in noten:
        if n >= 4:
            summe += n
            anzahl += 1

    if anzahl > 0:
        result[name] = round(summe / anzahl, 1)
    else:
        result[name] = 0

    return result

schueler = {
    "Anna": [5, 4, 3],
    "Ben": [3, 2],
    "Clara": [6, 5]
}

print(berechne_noten(schueler))

```

- a) Beschreibe den Ablauf der Funktion. (6 P)
- b) Warum ist die Abfrage *anzahl > 0* nötig? (2 P)
- c) Welchen Datentyp hat *noten* und woher weiss man das? (2 P)

Aufgabe 4 – Theorie (12 Punkte)

- 1) Was bedeutet *dynamisch typisiert*?
- 2) Unterschied *Liste* und *Set*
- 3) Unterschied *for x in liste* vs. *for i in range(len(liste))*
- 4) Was liefert *.items()* zurück?
- 5) Was passiert, wenn eine Funktion kein *return* hat?
- 6) Was ist *Tuple-Unpacking*?