

BE:

/ 29

Note

Unterschrift

① Schleifen

/ 4

- a) In der Programmierung unterscheidet man zwischen kopfgesteuerten und fußgesteuerten Schleifen. Erkläre den Unterschied zwischen diesen Schleifenarten. (2P.)
- b) Die folgende Python-Funktion enthält eine Schleife, die nicht endet. Erkläre, warum die Schleife nicht terminiert und wie man den Fehler beheben könnte. (2P.)

```
1 def count_down(n):
2     while n > 0:
3         print(n)
4         n += 1
```

② Verzweigungen

/ 12

Ein Freizeitpark bietet unterschiedliche Ticketpreise an:

- Kinder unter 6 Jahren zahlen nichts.
- Kinder von 6 bis 17 Jahren zahlen 10 Euro.
- Erwachsene von 18 bis 64 Jahren zahlen 20 Euro.
- Senioren ab 65 Jahren zahlen 15 Euro.

- a) Schreibe eine **Python-Funktion** *ticketpreis(alter)*, die den Preis für eine Person anhand ihres Alters berechnet und zurückgibt. (5P.)

Beispielaufruf:

```
1 print(ticketpreis(5))    # Erwartete Ausgabe: 0
2 print(ticketpreis(12))   # Erwartete Ausgabe: 10
3 print(ticketpreis(30))   # Erwartete Ausgabe: 20
4 print(ticketpreis(70))   # Erwartete Ausgabe: 15
```

- b) Falls eine Gruppe von mindestens 5 Personen gemeinsam ein Ticket kauft, erhält **jede** Person 10 % Rabatt auf ihren individuellen Ticketpreis. Erweitere die Funktion, indem du einen zweiten Parameter *ticketpreis(alter, personen)* erstellst. (4P.)

Beispielaufruf:

```
1 print(ticketpreis(30, 1)) # Erwartete Ausgabe: 20
2 print(ticketpreis(30, 5)) # Erwartete Ausgabe: 18
```

- c) Jeden Mittwoch ist „Mittwoch-Madness 🍷“. An diesem Tag wird von dem Gesamtpreis noch einmal 5€ abgezogen. Der Preis darf jedoch nicht unter 0 fallen. Erweitere den Code so, dass nun zusätzlich nach dem Wochentag gefragt wird. (3P.)

Beispielaufruf:

```
1 print(ticketpreis(30, 1, "Mittwoch")) # 15 €
2 print(ticketpreis(30, 5, "Mittwoch")) # 13 €
```

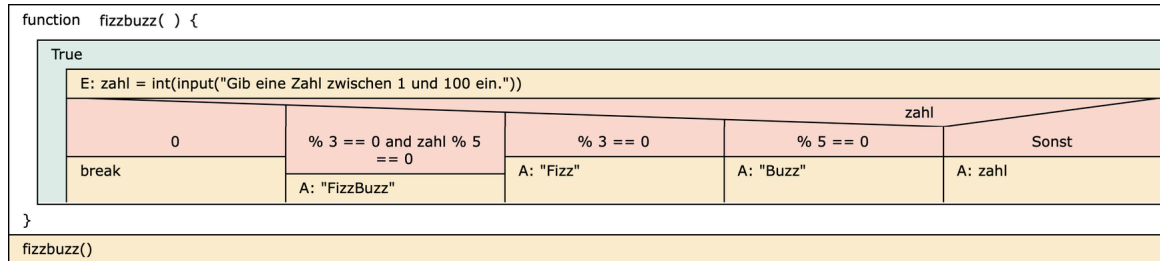
Tipp: Mit *deinString.lower()* werden alle Buchstaben kleingeschrieben, was wiederum den Vergleich der Strings vereinfacht.

③ Struktogramme

/ 13

Fizz buzz ist ein Gruppen-Wortspiel für Kinder.

Das folgende Struktogramm zeigt dieses Spiel.



- Erkläre anhand des gegebenen Struktogramms, welcher mathematische Aspekt den Kindern beigebracht werden soll. (1P.)
- Erstelle ein Python-Programm, welches dieses Spiel umsetzt. Orientiere dich dabei am gegebenen Struktogramm. (10P.)
- Der Benutzer darf nur Zahlen zwischen 1 und 100 eingeben. Falls eine ungültige Eingabe gemacht wird (z. B. -5 oder 150), soll das Programm eine Fehlermeldung ausgeben und erneut nach einer Zahl fragen. Erweitere das Programm entsprechend. (2P.)

Lösung/ Notizen: