

Aufgabe 5 - (Teil 2 von SCHIFFE VERSENKEN)

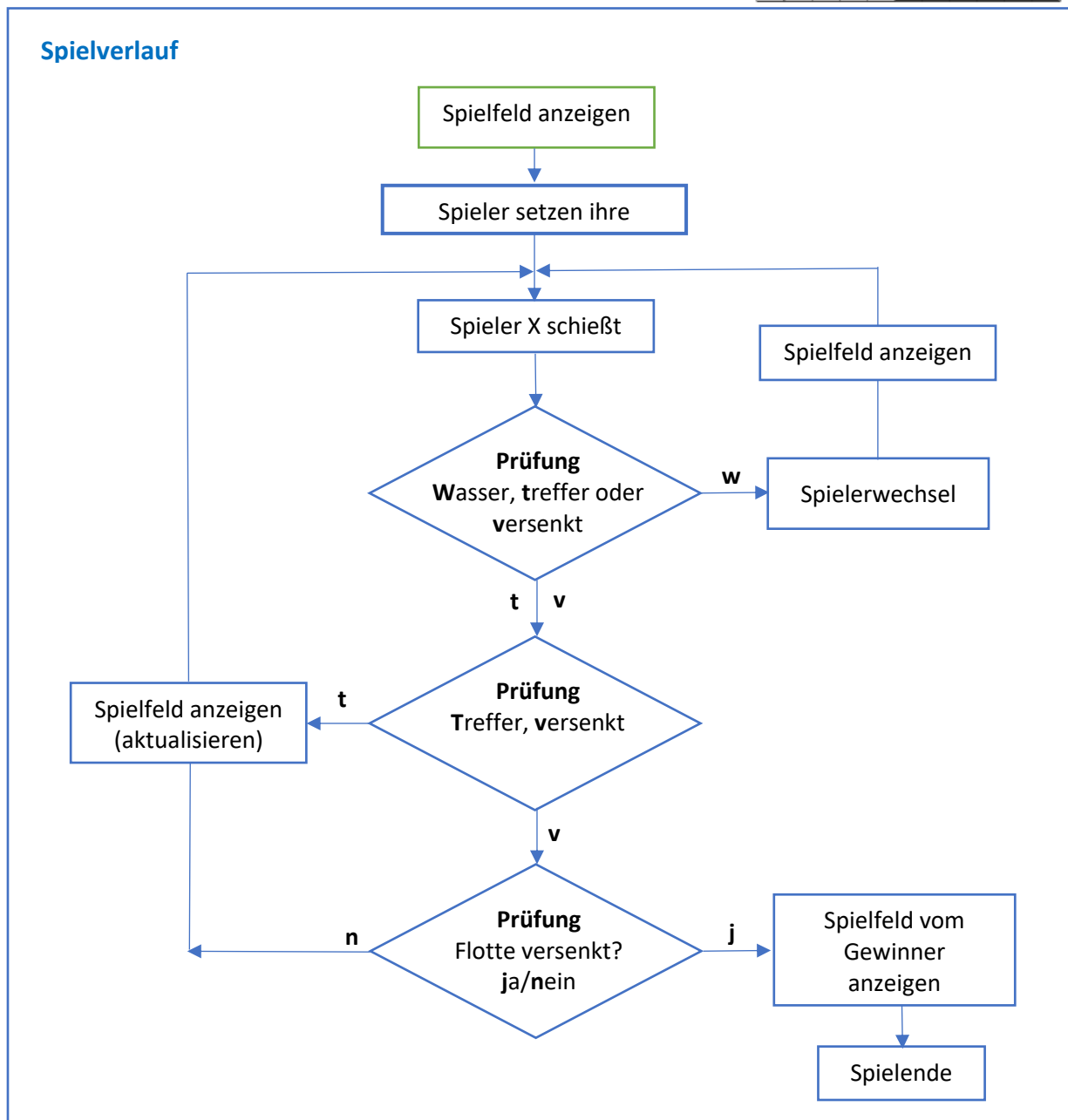
Situation: Laut einer aktuellen VuMA-Umfrage spielen mehr als 40 Prozent der Deutschen Computer- und Videospiele. In der Altersgruppe der 14- bis 29-Jährigen beträgt der Anteil der Videospiele sogar über 70 Prozent. Die meisten Gamer interessieren sich für Action-Spiele/ Ego-Shooter, Abenteuer-Spiele und Geschicklichkeitsspiele.



Es sollte das bekannte Spiel **SCHIFFE VERSENKEN** als Python-Programm erstellt werden.

Zuerst betrachten wir den **Spielverlauf** und überlegen uns, welche Prozesse wir als **Funktionen** umsetzen können.

| | A | B | C | D | E | F | G | H | I | J |
|----|---|---|---|---|---|---|---|---|---|---|
| 1 | | | | | | | | | | |
| 2 | | | | | | | | | | |
| 3 | | | | | | | | | | |
| 4 | | | | | | | | | | |
| 5 | | | | | | | | | | |
| 6 | | | | | | | | | | |
| 7 | | | | | | | | | | |
| 8 | | | | | | | | | | |
| 9 | | | | | | | | | | |
| 10 | | | | | | | | | | |



Folgende Funktionen leiten wir vom Spielverlauf ab:

| | |
|---------------------------------------|----------------------|
| Spielfeld anzeigen | spielfeld_anzeigen() |
| Spieler setzt seine Schiffe | spieler_flotte() |
| Spieler schießt | spieler_schiesst() |
| Spielerwechsel | spieler_wechsel() |
| Prüfung Wasser, Treffer oder versenkt | spieler_pwtv() |
| Prüfung Treffer, versenkt | spieler_ptv() |
| Prüfung Flotte versenkt | spieler_pfv() |

In der heutigen Aufgabe werden wir das Spielfeld mit den Schiffen der beiden Spieler füllen. Dafür werden wir den Datentyp „Liste“ einsetzen.

b.) Spieler setzt seine Schiffe

Beachten Sie das die beiden Spieler nach einander Ihre Flotte setzen. Zuerst der Spieler A, bis er alle seine Schiffe bzw. komplette Flotte gesetzt hat.

| | | |
|--------------------------|---------------------|-------------|
| Eine Flotte besteht aus: | Vier Einzelschiffen | 4 x O, |
| | Drei U-Boote | 3 x OO, |
| | Zwei Zerstörer | 2 x OOO und |
| | Einem Kreuzer | 1 x OOOO |

Die Flotte hat insgesamt 10 Schiffe (10=4+3+2+1).

Folgende Regeln gelten für das Setzen der Schiffe:

- Die Schiffe dürfen nicht aneinander stoßen.
- Die Schiffe dürfen nicht über Eck gebaut sein oder Ausbuchtungen besitzen.
- Die Schiffe dürfen auch am Rand liegen.
- Die Schiffe dürfen nicht diagonal n (entweder vertikal oder horizontal) aufgestellt werden.

Wir werden erstmal auf die Prüfung der Regeln für das Setzen der Schiffe verzichten und konzentrieren uns auf das **Abspeichern der gesetzten Schiffe** und der **Anzeige der Schiffe** auf dem Spielfeld. Für das Anzeigen der Schiffe auf dem Spielfeld X vom Spieler X benutzen Sie „O“.

Sobald der Spieler A seine 10 Schiffe gesetzt hat (=Flotte A), setzt der Spieler B seine 10 Schiffe (Flotte B). Danach kann dann mit dem Schießen begonnen werden.

Erstellen Sie hierzu die Funktion: **spieler_flotte()**

Datentyp **Liste**

Eine Liste ist eine Sequenz von Objekten. Sie kann Elemente unterschiedlichen Objekttyps enthalten. Man kann sich eine Liste als ein- oder mehrdimensionales Feld vorstellen (Array).

| | | |
|---|--|-------------------------------|
| 1 | <code>z = [3, 6, 12.5, -8, 5.5]</code> | ← Liste mit 5 Elementen |
| 2 | <code>print(z)</code> | gesamte Liste |
| 3 | <code>print(z[0])</code> | ein Element |
| 4 | <code>print(z[0:3])</code> | Ein Teilbereich (engl. Slice) |

Ausgabe: [3, 6, 12.5, -8, 5.5]
3
[3, 6, 12.5]

Achten Sie darauf, der Listen **Index** beginnt immer mit **0!**

Ihr Programm... `afg5_spielerschiffe.py`

```
def spielfeld_anzeigen():
    print(" | A | B | C | D | E | F | G | H | I | J |")
    print(" 0 | | | | | | | | | |")
    ....
    print(" 9 | | | | | | | | | |")

def spieler_flotte():
    Anweisungen...

#Hauptprogramm
spielfeld_anzeigen()
```