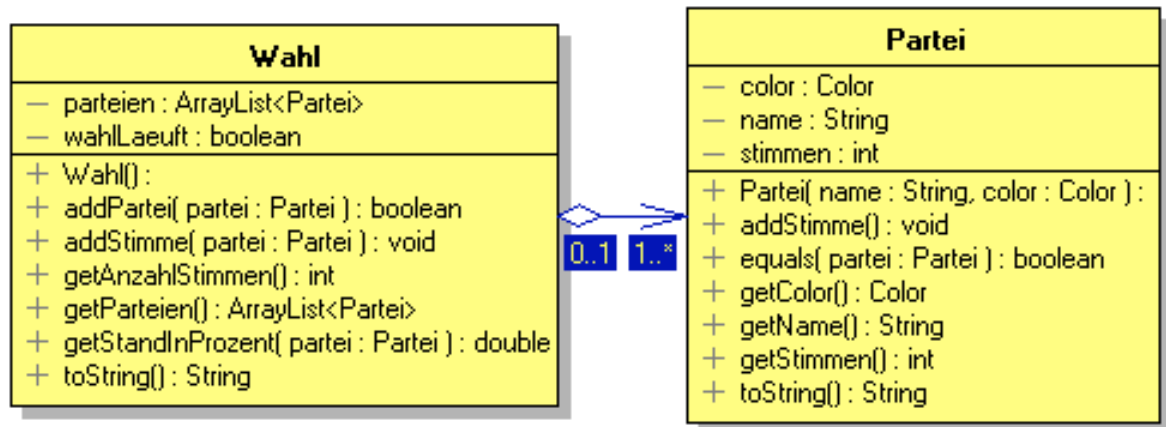

Modellierung und Programmierung 1

Übungsserie 6

Lösungsvorschläge

1. Modellierung

- a) **Wahl.java**: `public boolean addPartei(Partei partei)`
Partei.java: `public boolean equals(Partei partei)`
- b) **Wahl.java**: `public void addStimme(Partei partei)`
Partei.java: `public void addStimme()`
- c) **Wahl.java**:
`public double getStandInProzent(Partei partei)`
`public int getAnzahlStimmen()`



2. Implementierung

Wahl:

- a) Siehe **Partei.java** und **Wahl.java**.
- b) Siehe **WahlTest.java** und **WahlTest100.out** bzw. **WahlTest50000.out**.

3. GUI / MVC - Architektur

WahlMVC:

- a) Siehe **WahlModel.java**.
WahlModel wird überwacht: `public class Wahl extends Observable`
Änderungen werden angezeigt durch `setChanged()` und `notifyObservers()` in:
`public void addStimme(Partei partei)`
`public boolean addPartei(Partei partei) (optional)`
- b) Siehe **WahlView.java**.
- c) Siehe **WahlController.java**.
- d) Siehe **WahlMain.java**.
- e) **WahlDoc**: Online-Dokumentation.