

Wintersemester 2024/25  
Prof. Dr. J. Rexilius

Abgabetermin: 12.12.2024, 08:00

### Aufgabe 8 (9 Punkte)

*Beachten Sie die aktuelle Version zu den allgemeinen Abgabehinweisen (Praktikumsordner im ILIAS).*

#### (a) Poststation (5 Punkte)

Eine Poststation belädt Fahrräder mit Säcken und schickt sie dann in Gruppen auf unterschiedliche Routen. In Zukunft möchte die Station seine Fahrradgruppen elektronisch kontrollieren. Bei einem Gespräch mit dem Betreiber wurde das folgende Vorgehen festgelegt: Eine Gruppe besteht aus einer Anzahl Fahrrädern. Die Fahrräder einer Gruppe fahren immer in einer Reihe hintereinander. Nach jedem Fahrrad folgt genau ein nachfolgendes Fahrrad. Nur nach dem letzten Fahrrad gibt es kein weiteres mehr.

Ein einzelnes Fahrrad hat folgende private Parameter und Methoden (plus die entsprechenden set- und get-Methoden):

- `Fahrrad next`  
Nachfolge-Fahrrad in einer Gruppe. Zu Beginn ist der Nachfolger `null`.
- `int ladung`  
Die Anzahl an Säcken, die ein Fahrrad im Moment transportiert. Mit jedem Sack nimmt die Reisegeschwindigkeit gegenüber der Maximalgeschwindigkeit um einen Kilometer pro Stunde ab. Für `ladung >= maxSpeed` bleibt das Fahrrad stehen.
- `int maxSpeed`  
Anzahl der Kilometer, die das Fahrrad ohne Ladung pro Stunde zurücklegt. Der Parameter wird über den Konstruktor gesetzt.
- `int speed()`  
Liefert die aktuelle Geschwindigkeit des Fahrrads unter Berücksichtigung seiner Ladung.

Die Fahrradgruppe hat mindestens folgende Eigenschaften und Methoden:

- `Fahrrad startFahrrad`  
Eine Gruppe hat eine Referenz auf das erste Fahrrad der Gruppe. Diese Referenz ist initial `null`. Zunächst gehören also noch keine Fahrräder zur Gruppe.
- `addFahrrad(Fahrrad f)`  
Ein neues Fahrrad wird vor dem ersten Fahrrad in die Gruppe eingefügt
- `int reiseGeschwindigkeit()`  
Die Gruppe kann ihre Reisegeschwindigkeit bestimmen und ausgeben. Diese wird vom langsamsten Fahrrad in der Gruppe bestimmt.
- `addLadung(int anzahl)`  
Erhält die Gruppe `anzahl` neue Säcke für den Transport werden diese so auf alle Fahrräder verteilt, dass die Reisegeschwindigkeit der Gruppe möglichst hoch bleibt.

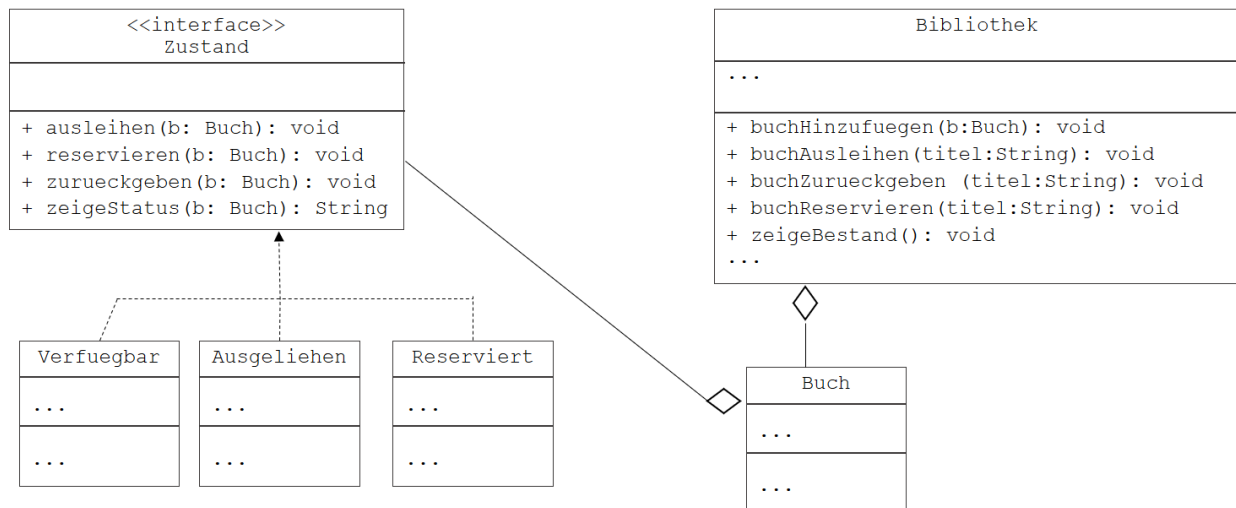
Implementieren Sie die zugehörigen Klassen und Methoden und schreiben Sie zusätzlich eine Testklasse, in der mindestens eine Gruppe mit unterschiedlichen Fahrrädern ( $\geq 3$  mit unterschiedlichen Maximalgeschwindigkeiten) und Beladungen betrieben werden, d.h. Gruppen mit Fahrrädern ausstatten, Ladung hinzufügen, und die Reisegeschwindigkeit bestimmen. Verwenden Sie für Ihre Implementierung keine Klassen aus dem Java Collection Framework und keine Arrays.

Wintersemester 2024/25  
Prof. Dr. J. Rexilius

Abgabetermin: 12.12.2024, 08:00

### Bibliothek (4 Punkte)

Eine Bibliothek benötigt ein neues System für die Verwaltung der Bücherausleihe. Jedes Buch hat einen Titel und einen Autor, wobei jedes Buch in dieser Kombination nur einmal vorkommt. Bücher können ausgeliehen, reserviert und zurückgegeben werden. Der Status („verfügbar“, „reserviert“, „ausgeliehen“) kann dabei für jedes Buch abgefragt werden. Für die Gesamtarchitektur wurde ein erstes Klassendiagramm wie folgt entworfen:



Implementieren Sie die Klassen wie oben beschrieben. Verwenden Sie für alle Klassen geeigneten Methoden und Variablen. Beachten Sie folgende Dinge bei der Implementierung

- Wenn ein Buch angelegt wird, ist es zunächst automatisch verfügbar.
- `zeigeBestand()`  
Nutzen Sie einen Iterator um den Bestand an Büchern auszugeben. Neben Autor und Titel sollen dabei auch jeweils der Status ausgegeben werden.  
Bsp: M. Mustermann: Buchtitel (verfügbar)
- `buchAusleihen()`, `buchZurueckgeben()`, `buchReservieren()`  
Die jeweilige Aktion soll nur ausgeführt werden, wenn das Buch auch vorhanden ist. Sonst wird eine Fehlermeldung auf der Konsole ausgegeben.

Testen Sie Ihre Implementierung. Fügen Sie dazu unterschiedliche Bücher der Bibliothek hinzu und rufen Sie die Methoden für ausleihen, reservieren und zurückgeben auf. Geben sie zusätzlich mindestens einmal den Gesamtbestand an Büchern über die entsprechende Methode aus.