

Alle Klassendefinitionen müssen gemäß der in den Tutorien besprochenen Javadoc-Konventionen kommentiert sein, da ansonsten nicht die volle Punktzahl erreicht werden kann.
--

**Aufgabe 5:** (5 Punkte) Realisieren Sie den ADT Queue in Java unter Verwendung zweier Objekte der Klasse `Stapel<T>`. Verwenden Sie hierzu die im LearnWeb bereit gestellten Schnittstellen `ADTStack<T>` und `ADTQueue<T>` sowie die Klasse `Stapel<T>`.

Geben Sie Testfälle an, mit denen Sie Ihre Implementierung testen können.

**Aufgabe 6:** (BA, 1+4=5 Punkte) Ein Programmierer überlegt sich bei der Lektüre von Folie 1.64, dass es sinnvoll sein könnte, die Methode `addNumbers` über den Datentyp der zu addierenden Zahlen zu parametrisieren, und schlägt folgenden Signatur vor:

```
public <T extends Number> T addNumbers(java.util.List<T> zahlen)
```

- (a) Was bedeutet `<T extends Number>` in der obigen Signatur?
- (b) Kann die Methode mit dieser Signatur (und der passenden Semantik) in Java implementiert werden? Wenn ja: wie muss die Implementierung aussehen? Wenn nein: worin besteht das Problem?

**Aufgabe 7:** (BA, 5 Punkte) Realisieren Sie den ADT Queue unter Verwendung eines zirkulären Felds. Die Größe des Felds soll bei der Konstruktion als Parameter übergeben werden und sich nicht mehr ändern. Dies bedeutet insbesondere, dass—wie auf Folie 1.28 angemerkt—diese Realisierung nicht mehr vollständig der Spezifikation entspricht; dies soll im Rahmen dieser Aufgabe jedoch kein Problem darstellen.

Verwenden Sie die im LearnWeb bereit gestellten Schnittstelle `ADTQueue<T>`.

Geben Sie Testfälle an, mit denen Sie Ihre Implementierung testen können.

**Aufgabe 8:** (2+1+2=5 Punkte) Eine *double-ended queue* (*Deque*) kombiniert die Eigenschaften einer FIFO-Queue und einer LIFO-Queue: man kann an beiden Enden der Warteschlange Elemente anfügen und löschen.

- (a) Geben Sie die Signatur für den ADT Deque an. (Semantikgleichungen sind hier nicht gefordert.)
  - (b) Geben Sie eine Java-Schnittstellen-Definition für Realisierungen des ADT Deque an.
  - (c) Implementieren Sie eine Klasse *Doppelwarteschlange*, die den ADT Deque realisiert.
-