

Übungsblatt 9

Abgabe: 12.1.2009

Eine im Vergleich zur Liste einfachere Datenstruktur zur Verwaltung von Nutzdaten ist der *Stack*, bei dem jeweils nur auf ein ausgezeichnetes Element (das oberste) zugegriffen werden kann. Auch die Implementierung ist einfacher.....

Aufgabe 1 Stack from scratch (30%)

Implementiert also einen Stack, der die folgenden Operationen zur Verfügung stellt:

- Erzeugung eines Stacks mit Hilfe eines passenden Konstruktors.
- `isEmpty` - Abfrage, ob ein Stack leer ist.
- `push` - Ablegen eines Elementes oben auf den Stack.
- `top` - Auslesen des obersten Elements des Stacks, ohne dieses zu entfernen.
- `pop` - Auslesen und Entfernen des obersten Elements vom Stack.

(mit passenden Parametern und Rückgabetypen)

Beim Dokumentieren sollen für alle Methoden die Pre- und Postconditions angegeben werden.

Aufgabe 2 Stack simuliert (30%)

Da wir schon viel Mühe in die Implementierung der Datenstruktur Liste gesteckt haben, wollen wir diese nutzen um die Stackoperationen zu implementieren. Schreibt eine zweite Implementierung eines Stacks, bei der die Methoden für die oben angegebenen Operationen auf die Listenoperationen des Beispiels aus der Vorlesung zugreifen.

Aufgabe 3 Liste einfach(er?) (40%)

Eine *einfach verkettete Liste* hat nur einen Referenz-Typ als Attribut, der auf das erste Listenelement (so vorhanden) zeigt. Jedes Listenelement hat eine Referenz auf seinen Nachfolger; gibt es kein Folgeelement, so ist diese Referenz `null`.

Implementiert die Methoden, die im Programm `MyList.java` für die Ringliste geschrieben wurden, für eine einfach verkettete Liste. Diskutiert jeweils die Vor- und Nachteile Eurer Implementierung gegenüber der Ringliste.