

Übungen zu Informatik II (Sommersemester 2008) Aufgabenblatt 7

Aufgabe 25 (Präsenzaufgabe)

In einer *zyklischen Liste* werden Objekte kreisförmig angeordnet gespeichert und es gibt zu jedem Zeitpunkt ein aktuelles Element in der Liste. Der ADT der zyklischen Liste stellt dazu die folgenden Operationen bereit:

- `isEmpty()`: Gibt an, ob die Liste leer ist.
- `goToNext()`: Das auf das aktuelle Element folgende Element wird zum aktuellen Element.
- `insert(o)`: Zwischen dem aktuellen Element und dem darauf folgenden Element wird ein neues Element mit dem Objekt o in die Liste eingefügt.
- `set(o)`: Das Objekt des aktuellen Elements der Liste wird durch o ersetzt.
- `get()`: Gibt das Objekt, das zum aktuellen Element der Liste gehört, zurück.
- `delete()`: Entfernt das auf das aktuelle Element folgende Element aus der Liste.

Geben Sie eine Implementierung **LinkedCyclicList** des ADT an, indem Sie wie in der Vorlesung Bestandteile, einen Konstruktor und Methoden angeben.

Aufgabe 26 (Präsenzaufgabe)

Der ADT der *Schlange mit beschränkter Kapazität* bietet dieselben Operationen wie der ADT der Schlange, allerdings wird dem Konstruktor die maximale Anzahl der in der Schlange zu speichernden Elemente übergeben. Sind bereits n Elemente in einer Schlange mit Kapazität n gespeichert, so führt ein weiterer Aufruf der `enqueue()`-Operation zu einem Fehler.

Geben Sie eine Implementierung **ArrayBoundedQueue** des ADT der Schlange mit beschränkter Kapazität an, die keine Container, sondern ein Feld für die Speicherung der Objekte nutzt. Die Laufzeit jeder Operation soll $\theta(1)$ sein, der Speicherbedarf einer Schlange mit Kapazität n soll $\theta(n)$ betragen.

Aufgabe 27 (Hausaufgabe)

In der Vorlesung wurden die abstrakten Datentypen *ungeordnete* und *geordnete Menge* vorgestellt. Erweitern Sie beide Datentypen jeweils um Methoden `union(s)` und `intersection(s)`, die die Vereinigung und den Durchschnitt einer Menge mit der Menge s als neue (!) Menge zurückgeben. Wird die Methode also beispielsweise für zwei geordnete Mengen s_1 und s_2 mit

```
setze  $r = s_1.union(s_2)$ 
```

aufgerufen, so ist r danach eine geordnete Menge mit allen Elementen aus s_1 und s_2 .

Bei Vereinigung oder Durchschnitt zweier geordneter Mengen s_1 und s_2 sollen Ihre Methoden eine Laufzeit von $O(|s_1| + |s_2|)$ haben.

Aufgabe 28 (Hausaufgabe)

Der ADT *Keller mit Minimum* erweitert einen Keller, der Objekte mit einem Vergleichsschlüssel speichert, um eine Methode `getMin()`, die ein im Keller gespeichertes Element zurückgibt, das einen minimalen Schlüssel besitzt. Das Element soll dabei nicht vom Keller entfernt werden.

Implementieren Sie den ADT Keller mit Minimum derart, dass die Laufzeit jeder Operation nach wie vor $\theta(1)$ und der Speicherbedarf $\theta(n)$ ist, wobei n die Anzahl der verwalteten Objekte bezeichnet.

Sie dürfen die Implementierung **LinkedStack** aus der Vorlesung benutzen. Erinnern Sie sich an den Vorschlag Ihres Kommilitonen aus der Vorlesung am 23.05.2008.

Ausgabe am 27.05.2008.

Besprechung der Präsenzaufgaben in den Übungen ab dem 28.05.2008.

Abgabe der Hausaufgaben bis zum 03.06.2008 um 12 Uhr im Schrein (HRS3).

Rückgabe und Besprechung der Hausaufgaben ab dem 04.06.2008.

Stichprobenkontrolle der Hausaufgaben ab dem 11.06.2008.