

Tutorium Praktische Informatik 2

Jonathan Köhn

Technische Hochschule Köln

Fakultät für Informations-, Medien- und Elektrotechnik

Institut für Nachrichtentechnik

14. Juni 2016

Dynamische Datenstrukturen

1. Definieren Sie die folgenden Datenstrukturen
 - I. Menge

Dynamische Datenstrukturen

1. Definieren Sie die folgenden Datenstrukturen
 - I. Menge
- A: Keine Reihenfolge, kein Element mehrfach

Dynamische Datenstrukturen

1. Definieren Sie die folgenden Datenstrukturen
 - I. Menge
 - A: Keine Reihenfolge, kein Element mehrfach
 - II. Liste

Dynamische Datenstrukturen

1. Definieren Sie die folgenden Datenstrukturen
 - I. Menge
 - A: Keine Reihenfolge, kein Element mehrfach
 - II. Liste
 - A: Hat Reihenfolge (linear, ein Nachfolger), mehrfaches Vorkommen möglich

Dynamische Datenstrukturen

1. Definieren Sie die folgenden Datenstrukturen
 - I. Menge
 - A: Keine Reihenfolge, kein Element mehrfach
 - II. Liste
 - A: Hat Reihenfolge (linear, ein Nachfolger), mehrfaches Vorkommen möglich
 - III. Assoziativer Array

Dynamische Datenstrukturen

1. Definieren Sie die folgenden Datenstrukturen
 - I. Menge
 - A: Keine Reihenfolge, kein Element mehrfach
 - II. Liste
 - A: Hat Reihenfolge (linear, ein Nachfolger), mehrfaches Vorkommen möglich
 - III. Assoziativer Array
 - A: Zu jedem gespeicherten Wert existiert ein eindeutiger (einmaliger) Schlüssel der frei gewählt werden kann.
2. Wie heißen die Implementierungen der genannten Datenstrukturen in Java?

Dynamische Datenstrukturen

Wiederholung

Programmieren

1. Definieren Sie die folgenden Datenstrukturen

I. Menge

A: Keine Reihenfolge, kein Element mehrfach

II. Liste

A: Hat Reihenfolge (linear, ein Nachfolger), mehrfaches Vorkommen möglich

III. Assoziativer Array

A: Zu jedem gespeicherten Wert existiert ein eindeutiger (einmaliger) Schlüssel der frei gewählt werden kann.

2. Wie heißen die Implementierungen der genannten Datenstrukturen in Java?

A: Interfaces: Set, List, Map

A: Implementierungen: HashSet, TreeSet, ArrayList, LinkedList, Stack, HashMap, Hashtable...

Dynamische Datenstrukturen

3. Welche Exception muss beim Umgang mit Listen hauptsächlich berücksichtigt werden?

Dynamische Datenstrukturen

3. Welche Exception muss beim Umgang mit Listen hauptsächlich berücksichtigt werden?

A: `IndexOutOfBoundsException`

Dynamische Datenstrukturen

3. Welche Exception muss beim Umgang mit Listen hauptsächlich berücksichtigt werden?
- A: `IndexOutOfBoundsException`
4. Wofür stehen die Abkürzungen FIFO und LIFO?

Dynamische Datenstrukturen

3. Welche Exception muss beim Umgang mit Listen hauptsächlich berücksichtigt werden?

A: `IndexOutOfBoundsException`

4. Wofür stehen die Abkürzungen FIFO und LIFO?

A: **F**irst **i**n, **f**irst **o**ut und **L**ast **i**n, **f**irst **o**ut.

Dynamische Datenstrukturen

3. Welche Exception muss beim Umgang mit Listen hauptsächlich berücksichtigt werden?

A: `IndexOutOfBoundsException`

4. Wofür stehen die Abkürzungen FIFO und LIFO?

A: **F**irst **i**n, **f**irst **o**ut und **L**ast **i**n, **f**irst **o**ut.

5. Was bedeuten sie?

Dynamische Datenstrukturen

Wiederholung

Programmieren

3. Welche Exception muss beim Umgang mit Listen hauptsächlich berücksichtigt werden?

A: `IndexOutOfBoundsException`

4. Wofür stehen die Abkürzungen FIFO und LIFO?

A: **F**irst **i**n, **f**irst **o**ut und **L**ast **i**n, **f**irst **o**ut.

5. Was bedeuten sie?

A: FIFO: Das zuerst hinzugefügte Element wird zuerst verarbeitet. (Warteschlange, Puffer)

A: LIFO: Das zuletzt hinzugefügte Element wird zuerst verarbeitet. (Kellerspeicher, Stack)

Dynamische Datenstrukturen

Wiederholung

Programmieren

1. Schreiben Sie in einer Java-Klasse Hilfsklasse eine Methode `intEinlesen()` und eine Methode `stringEinlesen()`. Diese sollen jeweils eine Ganzzahl bzw Zeichenkette einlesen und mittels `return` zurückgeben.
 2. Schreiben Sie eine Java-Klasse `Datenstrukturen`. Diese soll die Hauptmethode enthalten.
 3. Schreiben Sie mithilfe der `intEinlesen()` und einem `switch-case`-Konstrukt ein Menü entsprechend der folgenden Aufgaben.
- case 1: Eine `HashMap` wird die als Wert und als Schlüssel je einen `String` erwartet, wird vom Anwender gefüllt. Nutzen Sie dazu die Methode `stringEinlesen()`.

Dynamische Datenstrukturen

- case 2: Der Benutzer wird aufgefordert, einen Key einzugeben. Falls zu diesem ein Wert in der HashMap existiert, wird der Wert ausgegeben, ansonsten eine entsprechende Meldung angezeigt.
- case 3: Eine ArrayList für Zeichenketten wird mit Werten gefüllt.
- case 4: Alle Elemente der ArrayList werden nacheinander ausgegeben.
- case 5: Der Anwender kann eine Zeichenkette eingeben. Falls diese in der ArrayList enthalten ist, wird die Position in der Liste angegeben. (Mehrfachvorkommen muss nicht geprüft werden, es genügt ggf die erste Position.)
- case 6: Ein HashSet für Ganzzahlen wird gefüllt.
- case 7: Der Anwender kann ein weiteres Element hinzufügen. Dieses wird nur akzeptiert, falls es noch nicht in der Menge enthalten ist. *Prüfen Sie dies mit der entsprechenden Methode.*

Dynamische Datenstrukturen

- case 8: Der Anwender wird aufgefordert, eine Ganzzahl einzugeben. Ein Stack wird in einer Schleife mit den den Zahlen von 0 bis zur angegebenen Zahl gefüllt.
- case 9: Der Inhalt des Stacks wird in ein eine ArrayList übergeben.