



Grundlagen der Informatik 2

Praktikumsaufgabe 3

Abgabe: 23.06.2010, 12:00

Ratequiz. Dieses Mal soll ein einfaches interaktives Ratespiel mit Hilfe von Binärbäumen implementiert werden. Der Benutzer denkt an ein Tier, und der Computer versucht, es mit einer Folge von Ja/Nein Fragen zu erraten. Wenn der Computer falsch liegt, bringt ihm der Benutzer das neue Tier und eine neue Frage bei, mit der er das Tier von dem falsch geratenen unterscheiden kann. Einen Beispielauflauf des Spiels finden Sie im Kopf der Datei `Quiz.java` in der Quelltextvorgabe `Project3.zip`.

Binärbäume. Die Fragen und Tiere werden in einem Binärbaum gespeichert, der im Laufe des Spiels durchlaufen wird. Eine Folge von Ja/Nein Antworten führt schließlich zu einem Blatt mit einem Tier. Eine eventuelle neue Frage und ein neues Tier erweitern den Baum. Über mehrere Spiele hinweg kann der Baum in *Preorder* in einer Textdatei gespeichert und aus ihr geladen werden.

Für den Baum sollen die verkettete und die sequentielle Darstellung aus der Vorlesung implementiert werden. Die generische Klasse `BinaryTree` abstrahiert beide Darstellungen und bietet mit `BinaryTreeSearch` ein generisches Interface um beide Darstellungen zu durchlaufen.

- `LinkedBinaryTree`: In einer verketteten Darstellung eines Binärbaums enthält jeder Knoten ein Datenobjekt und zwei Verweise auf seine Kinder. Es bietet sich an, eine *verschachtelte Klasse* für Knoten innerhalb von `LinkedBinaryTree` anzulegen.
- `SequentialBinaryTree`: In der sequentiellen Darstellung sind die Knoten Einträge eines Arrays. Die Kinder eines Knoten an Position i befinden sich an den Positionen $2i$ bzw. $2i + 1$. Die Wurzel befindet sich an Position 1.

Zu implementieren. Wie immer sind die zu bearbeitenden Stellen im Code mit **TODO**: markiert. Im einzelnen:

1. `Quiz.java`: Implementieren Sie die Methode `run`, die eine Runde des Quiz durchführt. Interaktion mit dem User (und den Unit-Tests) ist mit den Methoden `askYesNo`, `askString` und `tell` der Klasse `QuizIoAdapter` möglich.

2. `LinkedBinaryTree` und `SequentialBinaryTree`: Implementieren Sie die Datenstrukturen und jeweils eine Methode `createFromPreorder`, die aus einer Liste von Strings in Preorder den entsprechenden Binärbaum erzeugt (zum Laden des Spielstands). Überlegen Sie sich, wie Sie in der Preorder für das Quiz innere Knoten von Blattknoten unterscheiden können.

Implementieren Sie jeweils eine entsprechende Methode `preorder`, die den Baum in eine Liste von Strings in Preorder umwandelt (zum Speichern des Spielstands).

3. Als Interface zum Quiz bieten beide Klassen die Methode `binaryTreeSearch`, die eine *anonyme* Klasse vom Typ `BinaryTreeSearch` für die Suche in einem Binärbaum erzeugt (ähnlich einem Iterator). Implementieren Sie die Methoden der anonymen Klassen für beide Baumdarstellungen. Die Methoden zum Abstieg im Baum `nextLeftChild` und `nextRightChild` sollen dabei `NoSuchElementExceptions` werfen, wenn das entsprechende Kind nicht existiert.

Um das Spiel mit beiden Baumdarstellungen zu probieren, können Sie die letzte Zeile in `load in Quiz.java` ändern.

4. Testen Sie Ihre Implementierungen mit den vorgegebenen Testfällen. *Hinweis:* Öffnen Sie `QuizTest.java` im Editor. Wählen Sie im Menü: *Run, Run As, JUnit Test*. Alle Tests die mit *linkedTree* beginnen, testen die Klasse `LinkedBinaryTree`, die Tests mit *sequentialTree* die Klasse `SequentialBinaryTree` und die Tests mit *quiz* die Methode `run` in `Quiz.java`.

Abgabe. Reichen Sie Ihre drei fertig bearbeiteten Dateien `Quiz.java`, `LinkedBinaryTree.java` und `SequentialBinaryTree.java` bis zum Abgabetermin in moodle ein. Die Abgabe ist in Gruppen von bis zu drei Personen möglich. Ein Mitglied der Gruppe reicht die Lösung zusammen mit einem Kommentar der folgenden Form ein: *Die Übung wurde bearbeitet von der Gruppe X, Y, Z; die Abgabe erfolgte durch X. Jedes andere* Mitglied der Gruppe reicht *anstelle* der Lösung nur diesen (identischen) Kommentar ein.

Bitte stellen Sie sicher, dass Sie und ggf. Ihre Gruppenmitglieder diese Regeln befolgen; wird weder eine Lösung noch der obenstehende Kommentar eingereicht, so vergibt der Tutor auch keine Punkte!

Hinweis: Der Fachbereich Informatik misst der Einhaltung der Grundregeln der wissenschaftlichen Ethik großen Wert bei. Mit der Abgabe einer Programmieraufgabe bestätigen Sie, dass Sie bzw. Ihre Lerngruppe die alleinigen Autoren sind.