

Übungsblatt 5

Abgabe: 25.06.2023

Ab diesem Übungsblatt wird nicht mehr zwischen implementierender und testender Person unterschieden, obwohl beides auf diesem Übungsblatt immer noch gemacht werden muss.

Aufgabe 1 Ahorn, Vogelkirsche, Lerche (20 %)

Fügt die Buchstaben *A*, *B*, *D*, *E*, *F*, *C* in einen leeren AVL-Baum ein. Löscht danach das *D*. Zeichnet die jeweiligen Zustände des Baums (inkl. der Knotenneigungen). Verdeutlicht die Rotationen, die notwendig sind.

Aufgabe 2 Wirklich Rot-Schwarz oder tut er nur so? (40 %)

Das beigelegte Projekt ist eine etwas gekürzte Fassung des Codes aus der Vorlesung. Es wurde alles entfernt, was nicht für die Rot-Schwarz-Bäume benötigt wird, mit Ausnahme der *toString*-Methoden, die vielleicht für ein Debugging ganz hilfreich sein können. Ergänzt wurde die Klasse *IsRedBlackTree* mit der Methode *check*, die ihr implementieren sollt. Diese überprüft bei einem übergebenen Baum, ob dieser die in der Vorlesung vorgestellten Eigenschaften eines Rot-Schwarz-Baums erfüllt. Als Ergebnis liefert sie einen Wert aus einem Aufzählungstyp zurück, der entweder sagt, dass alle Kriterien erfüllt sind (bereits vordefiniert: *OK*) oder ein verletztes Kriterium benennt. Sind mehrere verletzt, wird ein beliebiges davon gemeldet.

Erweitert den Aufzählungstyp *Result* um Symbole für die Eigenschaften, die verletzt sein könnten.¹ Implementiert dann *check* so, dass es die Eigenschaften prüft und den passenden Wert zurück liefert.²

Aufgabe 3 Test testen (40 %)

Implementiert Tests in der Klasse *IsRedBlackTreeTest*, die überprüfen, ob die Methode *check* korrekt funktioniert. Hierzu wurde der Code aus der Vorlesung so erweitert, dass ihr auch von Hand Rot-Schwarz-Bäume aus *RBNode*-Objekten erzeugen könnt. Somit könnt ihr fehlerhafte Bäume konstruieren.³

Implementiert zusätzlich einen oder mehrere Tests, die mit Hilfe von *check* überprüfen, ob ein *RBTree* überhaupt richtig funktioniert, d.h. die Rot-Schwarz-Eigenschaften nach Einfüge- und Löschoperationen noch erfüllt sind und eingefügte Werte auch wieder gefunden und gelöschte nicht mehr gefunden werden.

Die *toString*-Methoden brauchen nicht getestet zu werden.

¹Die Eigenschaft „Blätter sind schwarz“ ist per Definition immer erfüllt und kann nicht geprüft werden, da Blätter *null* sind.

²Beachtet, dass einige Referenzen auf Knoten vom Typ *Node<E>* sind und ihr eine Typumwandlung nach *RBNode<E>* machen müsst, um an das Attribut *color* zu gelangen.

³Werte von Aufzählungstypen können übrigens auch *null* sein, d.h. es gibt für eine trivial aussehende Rot-Schwarz-Eigenschaft durchaus eine Möglichkeit, sie zu verletzen.