
Grundlagen: Algorithmen und Datenstrukturen

Abgabetermin: Jeweilige Tutorübung in der Woche vom 9. bis 13. Juni

Wiederholung einiger Definitionen: Ein Knoten eines gewurzelten Baumes ist ein Blatt, wenn er keine Kinder hat. Andernfalls ist er ein innerer Knoten.

(In der Literatur wird ein gewurzelter Baum, der nur aus der Wurzel besteht, nicht einheitlich gehandhabt - in einigen Fällen ist dieser einzelne Knoten per Definition ein innerer Knoten, in anderen Fällen ein Blatt, und in wieder anderer Literatur ein Spezialfall, der weder innerer Knoten, noch Blatt ist. Bei unserer Definition ist ein solcher Knoten ein Blatt.)

Die Tiefe eines Knotens ist die Länge des (eindeutigen) Pfades von dem Knoten zur Wurzel, gemessen in der Zahl der Kanten des Pfades. D.h. insbesondere, dass die Wurzel Tiefe 0 hat. (Dies ist nicht einheitlich in der Literatur: in einigen Fällen wird in der Zahl der Knoten des Pfades gemessen, was dann gerade 1 mehr ist als die Zahl der Kanten).

Die Baumtiefe eines gewurzelten Baumes ist das Maximum der Tiefen aller enthaltenen Knoten. Das i -te Level eines gewurzelten Baumes besteht aus allen Knoten, die Tiefe i besitzen.

Ein gewurzelter Baum ist ein Binärbaum, wenn jeder innere Knoten maximal zwei Kinder hat. Wenn jeder innere Knoten genau zwei Kinder hat, ist er ein echter Binärbaum. Ein Binärbaum ist vollständig, wenn er ein echter Binärbaum ist und alle Blätter dieselbe Tiefe haben.

Sei t die Tiefe eines Binärbaumes T . Der Baum T ist ein fast vollständiger Binärbaum, wenn T entweder nur aus der Wurzel besteht, oder wenn die ersten $t - 1$ Level gemeinsam einen vollständigen Binärbaum bilden, und die Knoten von T so angeordnet werden können, dass es auf dem Level t einen Knoten e mit dem Vaterknoten v gibt, sodass gilt:

- Alle Knoten auf dem Level $t - 1$, die „links“ von v stehen, haben zwei Kinder.
- Alle Knoten auf dem Level $t - 1$, die „rechts“ von v stehen, haben keine Kinder.

Die folgende Abbildung zeigt einen fast vollständigen Binärbaum mit Baumtiefe 4.

- `delMin()`,
- `decreaseKey(27, 10)`,
- `delete(10)`.
- `insert(11)`,
- `insert(1)`

Hausaufgabe 2

Ein Einfügeknoten eines Binärbaums ist eine Position, an der ein weiteres Blatt angehängt werden kann, sodass der resultierende Baum wieder ein Binärbaum ist. Mit anderen Worten, jedes Blatt besitzt zwei Einfügeknoten, ein innerer Knoten mit einem Kind (Halbblatt) hat genau einen Einfügeknoten, und ein innerer Knoten mit zwei Kindern hat keinen Einfügeknoten. Der leere Baum hat zudem per Definition einen Einfügeknoten.

Für einen Binärbaum sei n seine Knotenzahl und e die Zahl seiner Einfügeknoten. Zeigen Sie $e = n + 1$ mit (starker) vollständiger Induktion.

Hausaufgabe 3

Implementieren Sie in der Klasse `UIaHeap` einen adressierbaren binären Heap, der neben den Basisoperationen auch die folgenden Operationen bereitstellt:

- `insertH(e)`: Fügt ein Element e in den Heap ein und gibt ein Integer als Handle zurück
- `removeH(h)`: Entfernt das Element mit Handle h
- `decreaseKeyH(h, k)`: Verringert das Element mit Handle h auf den Wert k .

Verwenden Sie für Ihre Implementierung die auf der Übungswebseite bereitgestellten Klassen und verändern Sie für Ihre Implementierung *ausschließlich* die Klasse `UIaHeap`.

Achten Sie bei der Abgabe Ihrer Aufgabe darauf, dass Ihre Klasse `UIaHeap` heißt und auf den Rechnern der Linuxhalle (`lxhalle.informatik.tu-muenchen.de`) mit der bereitgestellten Datei `main_ah` kompiliert werden kann. Anderenfalls kann eine Korrektur nicht garantiert werden. Achten Sie darauf, dass Ihr Quelltext ausreichend kommentiert ist.

Schicken Sie die Lösung per Email mit dem Betreff `[GAD] Gruppe <Gruppennummer>` an Ihren Tutor.

Hinweis: Hier ist die Verwendung der von Java bereitgestellten dynamischen Arrays sinnvoll.

Anmerkung: Handles müssen im Allgemeinen keine Integers sein. Oft ist es sinnvoller, Objekte als Handles zu verwenden, die Referenzen auf die tatsächlichen Daten in der Datenstruktur bzw. deren Wrapper besitzen.