

Übung zur Vorlesung EidP (WS 2020/21)

Blatt 9

Block grün

Es können 4 Punkte erreicht werden.

Abgabedatum: 28. Januar 2021, 23:59 Uhr

Hinweise

- Bitte beachten Sie die aktuellen Hinweise unter

<https://ls11-www.cs.tu-dortmund.de/teaching/ep2021uebung/>

- Die Aufgaben sind **in Dreiergruppen** zu bearbeiten.
- Für die Abgabe sind die jeweils genannten Dateien zu erstellen.
- Stellen Sie sicher, dass alle von Ihnen abgegebene Dateien reine Textdateien im UTF-8-Format sind.
- Die Verwendung von zusätzlichen **Bibliotheken** zur Lösung der Aufgaben ist **nicht erlaubt!**
- **Die Präsenzaufgabe entfällt auf diesm Blatt, sodass stattdessen Aufgaben der Probeklausur besprochen werden können.** Bitte teilen Sie Ihrem Übungsgruppenleiter rechtzeitig mit, welche Aufgaben Sie besprechen möchten.
- Dies ist das erste Blatt im dritten, grünen Block.

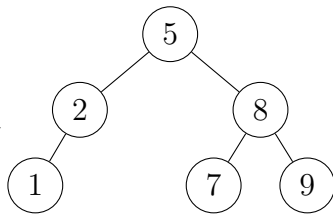
Aufgaben

Aufgabe 1: ADT Binärer Suchbaum (2 Punkte)

In der Vorlesung und in der vorherigen Übung haben Sie den ADT *binärer Suchbaum* für den effizienten Zugriff auf sortierbare Elemente kennengelernt. Die Implementierung des ADT *binärer Suchbaum* sowie die Testeingabe sind auf der Website der Übung zu finden. Laden Sie für Ihre Abgabe die Dateien `Aufgabe_09_1.h` und `Aufgabe_09_1.cpp` von der Website der Übung herunter. Erweitern Sie den ADT um die folgende Methode:

Die Methode `void preorder()` soll für den Suchbaum, für den sie aufgerufen wird, die Werte der Knoten des Baums in Preorder-Reihenfolge auf der Konsole ausgeben. Die Erweiterung um eigene Methoden ist hierbei möglich.

Beispiel:



Der Aufruf `preorder()` gibt auf der Konsole die Werte 5 2 1 8 7 9 aus.

Kompilieren Sie Ihr Programm, überprüfen Sie Ihre Ergebnisse und kopieren Sie diese als Block-Kommentar an das Ende der cpp-Datei `Aufgabe_09_1.cpp`.

Aufgabe 2: Kopierkonstruktor und Zuweisungsoperator (2 Punkte)

In der Vorlesung und in der vorherigen Übung haben Sie den ADT *binärer Suchbaum* für den effizienten Zugriff auf sortierbare Elemente kennengelernt. In der Vorlesung wurden jedoch der Kopierkonstruktor und der Zuweisungsoperator nicht implementiert, was bei einer Klasse mit dynamischem Speicher ja eigentlich notwendig wäre. Erweitern Sie den ADT binärer Suchbaum um Kopierkonstruktor und Zuweisungsoperator so, wie es für andere ADTs in der Vorlesung vorgemacht wurde.

Hinweis: Die Implementierung des ADT *binärer Suchbaum* sowie die Testeingabe sind auf der Website der Übung zu finden. Laden Sie für Ihre Abgabe die Dateien `Aufgabe_09_2.h` und `Aufgabe_09_2.cpp` von der Website der Übung herunter. Speichern Sie Ihre Methoden in der h-Datei `Aufgabe_09_2.h`.