

Algorithmen und Datenstrukturen I

Blatt 7 (Abgabe 2)

Prof. Dr. Oliver Braun

Fakultät für Informatik und Mathematik
Hochschule München

Letzte Änderung: 09.12.2019 16:28

Achtung: Sollten Sie die erste Abgabe (Blatt 5) nicht bestanden haben, können Sie auch mit diesem Blatt den Schein nicht mehr bekommen.

Lesen Sie vor der Bearbeitung <https://ob.cs.hm.edu/exercises.html> und stellen Sie sicher, dass es *keine* Commits von Ihnen im master-Branch gibt.

Aufgabe 1

Diese Aufgabe kann alleine oder in einer Zweiergruppe bearbeiten und eingereicht werden. Über den [GitHub Classroom Link](#) können Sie ein Team erstellen oder einem bereits erstellten Team beitreten. Auch wenn Sie alleine arbeiten, müssen Sie das als “Einerteam” machen. Das Team bekommt ein gemeinsames, (fast) leeres Repository.

Zur Erfüllung dieser Aufgabe muss folgendes bis zum **13.01.2020, 10:00** umgesetzt sein:

1. Implementieren Sie einen AVL-Baum (siehe Ottmann & Widmayer, Algorithmen und Datenstrukturen, Abschnitt 5.2.1) als Bibliothek in C++. Der AVL-Baum soll **kein** Template sein, sondern enthält nur `ints` als Schlüssel.
2. Die stichprobenartige Qualitätssicherung der Bibliothek erfolgt mit Googletest.
3. Das Projekt muss inkl. Tests mit CMake gebaut werden können.
4. Der Build und die Tests werden mit einem [Jenkins-Job](#) ausgeführt, der bei jedem Push auf GitHub getriggert wird.

Zusätzlich muss, aber nur wenn Sie in einem Zweierteam arbeiten, folgendes umgesetzt werden:

1. Die Googletest-Testsuite sichert die Qualität der Bibliothek in einem sehr hohen Maß. Auf Deutsch: **Signifikant mehr Tests als alleine.**
2. Das öffentliche Interface des AVL-Baumes ist mit [Doxygen](#) dokumentiert und als [GitHub-Page](#) veröffentlicht.
3. Die Datei `README.md` enthält einen Link auf Ihre GitHub-Page.

Die Abgabe erfolgt durch fristgerechtes Erzeugen eines Pull-Requests gegen den `master`-Branch.

Im Januar wird diese Aufgabe teamweise während der Praktikumstermine von mir abgenommen. Bereiten Sie dazu ein Programm (`main()`) vor mit dem Sie die Nutzung Ihrer Bibliothek zeigen können. Veranschaulichen Sie dabei unbedingt auch das Balancieren des Baumes nach Einfügen oder Löschen beispielsweise durch eine aussagekräftige Ausgabe (textuell, dot, JSON, ...).

Tipps und Anmerkungen

- Kümmern Sie sich am Anfang gleich um die korrekte Jenkins-Anbindung (Stichwort `Jenkinsfile`).
- Schreiben Sie immer gleich Tests.
- Committen und pushen Sie so oft wie möglich.
- Sie dürfen den meinen und Ihren Code von Aufgabe 6 natürlich wieder verwenden.

und schreiben Sie **vernünftige** Commit-Messages!