

# Aufgabe 4: Rekursion

In dieser Aufgabe werden AVL-Bäume näher beleuchtet. Zudem wird sich näher mit der Rekursion befasst.

## Aufgabenstellung

Sei  $zahlen$  eine endliche Folge von ganzen positiven Zufallszahlen gegeben, die in einen AVL-Baum einzufügen sind.

**Implementieren** Sie nun folgende Teile:

1. AVL-Baum als **höhenbalancierten binären Suchbaum**: Implementieren Sie den ADT **AVL-Baum** gemäss der Definition aus der Vorlesung. Implementieren Sie die beiden Operationen `einfügen` und `löschen`, so dass Sie jeweils mit einer Rekursion alle notwendigen Maßnahmen durchführen können! Implementieren Sie zudem die benötigten Rotationen `LinksRotation`, `RechtsRotation`, `DoppeltLinksRotation`, `DoppeltRechtsRotation`.
2. Beachten Sie: wegen den zufällig erzeugten Schlüsseln sollten Sie beim Einlesen gleiche Schlüssel ignorieren.
3. Erweitern Sie den ADT so, dass die Anzahl der jeweiligen Rotationen gezählt werden.
4. Implementieren Sie eine Ausleseoperationen, die den Baum in geeigneter Art und Weise ausgibt inklusive der beim Aufbau gezählten Rotationen und in einer Datei [avl.log](#) speichert.

Die 256 Zufallszahlen sind aus einer Datei [zzahlen.dat](#) einzulesen. Der Baum ist danach auszugeben. Dann sind von diesen 256 Zufallszahlen 25 zufällig ausgewählte zu löschen. Der Baum ist erneut auszugeben.

**Implementieren** Sie nun folgende Anwendungen:

1. `SummePost` als **Postorder Addition**: Implementieren Sie eine Methode, die die Summe der Schlüssel in einem gegebenen AVL-Baum in Postorderweise (Summe Linker Teilbaum + Summe Rechter Teilbaum + Knoten) bestimmt.
2. `SummePre` als **Preorder Addition**: Implementieren Sie eine Methode, die die Summe der Schlüssel in einem gegebenen AVL-Baum in Preorderweise (Knoten + Summe Linker Teilbaum + Summe Rechter Teilbaum) bestimmt.
3. `controlSum` als normale Addition (zur Kontrolle) auf der Datei `zahlen.dat`.

Die Endsummen sind in der Datei [avl.log](#) mit auszugeben. Sofern Sie etwas darin geübt sind, eine GUI zu erstellen, sollten Sie sich den Baum z.B. als graphische Darstellung ([AVL.jpg](#)) anzeigen lassen, da man dann ab einer bestimmten Grösse besser erkennen kann, was ggf. nicht korrekt implementiert wurde.

# Abnahme

Da die Aufgaben des ADP sehr frühzeitig im WWW zur Verfügung stehen, wird die Abnahme stark auf eine **vorbereitende Arbeit** aufgebaut.

**Bis Sonntag Abend** vor Ihrem Praktikumstermin ist eine erste [Skizze](#) der Aufgabe als \*.pdf Dokument ([Dokumentationskopf](#) nicht vergessen!) mir per E-Mail zuzusenden. Ggf. können offene Fragen mit gesendet werden. Die Skizze **muss** grob beschreiben, wie Sie sich die Realisierung denken. Als erfolgreich wird eine Skizze bewertet, wenn Ihre Kenntnisse bzgl. der gestellten Aufgabe eine erfolgreiche Teilnahme an dem Praktikumstermin in Aussicht stellen.

Am Tag des Praktikums findet eine Befragung von Teams statt. Die **Befragung muss erfolgreich absolviert werden**, um weiter am Praktikum teilnehmen zu können. Ist die Befragung nicht erfolgreich, gilt die Aufgabe als nicht erfolgreich bearbeitet. Als erfolgreich wird die Befragung bewertet, wenn Ihre Kenntnisse eine erfolgreiche Teilnahme an dem Praktikumstermin in Aussicht stellen. Bei der Befragung handelt es sich nicht um die Abnahme.

**Abgabe:** Unmittelbar am Ende des Praktikums, spätestens bis 19:00 Uhr am selben Tag, ist von allen Teams der Code abzugeben. Zu dem Code gehören die Sourcedateien, die ggf. erzeugten \*.log etc. Dateien, die während der Tests erzeugt wurden, und eine Readme.txt Datei, in der ausführlich beschrieben wird, wie die Software zu starten ist! Zudem ist der aktuelle Dokumentationskopf abzugeben. Die Dateien sind als \*.zip Ordner (mit cc an den/die Teamprätner\_in) per E-Mail abzugeben. Die Abgabe gehört zu den PVL-Bedingungen und ist einzuhalten, terminlich wie auch inhaltlich!

Wird eine Aufgabe nicht erfolgreich bearbeitet, gilt die **PVL als nicht bestanden**. Damit eine Aufgabe als erfolgreich gewertet wird, muß die Befragung als erfolgreich gewertet werden sowie die Abgabe abgenommen worden sein. **Alle gesetzten Termine sind einzuhalten**. Dies ist notwendig, da sonst erhebliche zeitliche Verzögerungen stattfinden würden..