



Praktikum OOSE

Praktikumsunterlagen 01

SoSe 2022

Dr. Felix Boes

Hinweis. In jeder der drei Praktikumsphasen muss jede:r Praktikumssteilnehmer:in erfolgreich einen Zwischenstand und den Endstand des Projekts präsentieren.

Story Nachdem der Handel mit schnell verderblichen Gütern die Aufmerksamkeit von sehr interessierten Mitbewerbern und Behörden wahrgenommen wurde, hat sich die Boes Inc. anderen profitablen Geschäftszweigen zugewandt. Nach ersten Erfolgen auf dem Softwaremarkt, wurden die ersten Praktikant:innen eingestellt. Die Firmenleitung sucht mithilfe des Praktikums die besten potentiellen jungdynamischen Mitarbeiter:innen.

Das Praktikum ist in drei Praktikumsphasen eingeteilt. In der ersten Projektphase sollen die Praktikant:innen nachweisen, dass Sie mit den Grundzügen der objektorientierten Softwareentwicklung, der Projektkoordination, dem verteilten Arbeiten und den dazugehörigen Werkzeugen vertraut sind. Deshalb werden wöchentlich neue Teilziele durch den Vice President der Boes Inc. bekannt gegeben.

Ziel der ersten Phase Das übergeordnete Ziel ist es im Team die folgenden generischen Datentypen verteilt zu entwickeln. Die Entwickelte Software soll in einem Packet mit dem Namen `inc.boes.praktikum` organisiert werden. Das gesamte Projekt soll aussagekräftig und umfänglich dokumentiert sein. Dabei sollen alle in der Vorlesung genannten Dokumentationsansätze verwendet werden. Außerdem sollen Buildsysteme verwendet werden, Unit Tests umgesetzt werden und die Performance der Implementierung im Vergleich zur Performance der Standardbibliothek gemessen und diskutiert werden.

Zur konkreteren Aufgabenstellung der ersten Woche Lösen Sie die unten genannten Aufgaben mit den Werkzeugen und Methodiken der verteilten Softwareentwicklung. Dazu müssen Sie die Aufgaben selbstständig in Teilaufgaben zerlegen und auf Ihre Gruppenmitglieder verteilen. Aus Ihrer Dokumentation muss (wie in der Vorlesung erklärt) hervorgehen welche Teilaufgaben es gibt und welches Gruppenmitglied für welche Teilaufgaben zuständig ist (oder war).

Aufgabe 1 (Abstrakte Datentypen als Interfaces). Entwerfen Sie für die folgenden generischen Datentypen jeweils ein Interface. Dabei soll jeder Interfacename mit dem Namen **Abstract** beginnen. Entwerfen Sie alle hier genannten Interfaces in einer nachvollziehbare Interface-Hierarchie. Fügen Sie dabei ggf. weitere Interfaces hinzu um die Interface-Hierarchie feiner (und damit nachvollziehbarer) zu gliedern.

Für die folgenden generischen Datentypen sollen die zu speichernden Typen generisch sein.

- Einfach verkettete Liste
- Doppelt verkettete Liste
- Queue
- Stack

Für die folgenden generischen Datentypen sollen die zu speichernden Typen generisch Zahltypen sein.

- Min Heap
- Binärer Suchbaum
- AVL Baum

Aufgabe 2 (Iteratoren). In der obigen Aufgabe haben Sie Interfaces für Ansammlungen von generischen Daten- oder generischen Zahltypen definiert. Um die Elemente einer solchen Ansammlung zu durchlaufen (z.B. um die Elemente nacheinander formatiert auf der Standardausgabe auszudrucken) verwendet man Iteratoren.

Lesen Sie sich in die Funktionsweise von Iteratoren ein. Erweitern Sie die Interfaces der oben konstruierten Interface-Hierarchie so dass sie iterierbar werden.

Aufgabe 3 (Konkrete Datentypen als Klassen). Geben Sie für jeden der oben genannten abstrakten Datentypen eine konkrete Implementierung an die das zugehörige Interface implementiert.