



1. Zulassungsklausur (A) zur Veranstaltung
Einführung in die Informatik II - Vertiefung
und **Allgemeine Informatik 2**

im Sommersemester 2022

Prüfer: Dr. Jens Kohlmeyer

Fakultät Ingenieurwissenschaften, Informatik, Psychologie

4.06.2022, 10 Uhr

Bearbeitungszeit: 45 min

Nachname:	Vorname:	Matrikelnummer:
Studiengang und Abschluss:		Account:
<p>Hiermit erkläre ich, dass ich prüfungsfähig bin. Sollte ich nicht auf der Liste der angemeldeten Studierenden aufgeführt sein, dann nehme ich hiermit zur Kenntnis, dass diese Prüfung nicht gewertet werden wird.</p> <p>_____</p> <p>Datum, Unterschrift des Prüfungsteilnehmers</p>		

Zur allgemeinen Beachtung:

- Füllen Sie das Deckblatt vollständig und korrekt aus.
- Lesen Sie sich zunächst die Klausur sorgfältig durch (sie besteht aus 2 Seiten).
- Zum Erreichen der Vorleistung benötigen Sie **50** der insgesamt **100** Punkte, die Sie über die 2 Zulassungsklausuren hinweg erreichen können.
- Im Ordner ~/materials finden Sie das zu bearbeitende Projekt sowohl als ZIP als auch entpackt.
- Sie dürfen im Projekt beliebig Hilfsmethoden und Klassen hinzufügen. Die Signaturen der vorgegebenen Methoden dürfen nicht verändert werden.
- **Vor Beginn der Bearbeitungszeit** haben Sie 5 Minuten Zeit um einen Editor/eine IDE Ihrer Wahl zu starten und sich in den Materialien zurecht zu finden.
- **Nach Ablauf der Bearbeitungszeit** haben Sie 5 Minuten Zeit um ihre Lösung im Ordner ~/export abzulegen. **Bleiben Sie sitzen!**
- Abgaben sind als Projeklexport in Zip-Form abzugeben.

Punkteverteilung				
1	2	3	Σ	Note
von 7	von 11	von 12	von 30	
				Korrektur

Das Projekt

Das Ihnen zur Verfügung gestellte Projekt setzt ein Bankensystem um. In diesem System soll es jeder Bank möglich sein die Vaults aller Banken einzusehen, aber nur die eigenen Vaults zu verändern. Diese Funktionalität wird durch die Verwendung einer zentralen Verwaltungseinheit (VaultingManager) umgesetzt.

Das Projekt besteht aus **5** Klassen und einem Interface. Im folgenden eine kurze Übersicht über die einzelnen Klassen:

ei di 2. sose2022. administration_exam.group01.Main: Diese Klasse enthält keinen funktionalen Inhalt. Sie können diese verwenden um Ihre Implementierungen zu testen.

ei di 2. sose2022. administration_exam.group01.Bank: Diese Klasse stellt die im System zu verwaltenen Banken dar. Alle Funktionalität ist bereits umgesetzt.

ei di 2. sose2022. administration_exam.group01.VaultingManager: Diese Klasse stellt die zentrale Verwaltungseinheit des Projektes dar. In ihr sollen verschiedene Funktionalitäten umgesetzt werden. Die Klasse besitzt ein einzelnes Attribut banksAndVaults. Dieses Attribut soll die Zuordnung aller Banken zu deren Accounts beinhalten.

ei di 2. sose2022. administration_exam.util.Tuple: Diese Klasse wird dazu verwendet eine Zuordnung zwischen den zwei darin enthaltenen Objekten zu ermöglichen.

ei di 2. sose2022. administration_exam.storage.IStoreable: Dieses Interface definiert welche Methoden jedes in Vaults enthaltene Objekt implementieren muss.

ei di 2. sose2022. administration_exam.storage.Vault: Diese Klasse bildet Bankaccounts im System ab. Sie enthält eine eindeutige ID und eine Liste von Objekten die im Account gespeichert werden.

Aufgabe 1 - Vault**7 Punkte**

Implementieren Sie die Methode `removeValue(int value)` der Klasse `Vault`. Diese soll das erste Element, welches im Vault gespeichert ist, dessen Wert dem übergebenen Wert entspricht aus den `contents` löschen. Sollte kein Element dem übergebenen Wert entsprechen, so soll kein Element gelöscht werden. Die Methode soll den Gesamtwert des Vaults nach der Löschoperation zurückgeben. Sollte kein Element aus dem Vault gelöscht werden, soll trotzdem der Gesamtwert des Vaults zurückgegeben werden. Dieser entspricht dann dem Gesamtwert der schon vor dem Aufruf der Methode galt.

Hinweis: Beachten Sie die Definition von `Vault` und den Typ des `contents` Attributs.

Aufgabe 2 - FindBankTuple**11 Punkte**

Implementieren Sie die Methode `findBankTuple(Bank bankToFind)` der Klasse `VaultingManager`. Die Methode soll aus der Liste aller Banken und Vaults das Tupel zurückliefern, welches an erster Stelle die übergebene Bank (`bankToFind`) enthält. Sollte die übergebene Bank nicht in der Liste existieren, soll `null` zurückgeliefert werden.

Hinweis: Sie dürfen davon ausgehen, dass der Methode niemals `null` übergeben wird. Ein Vergleich auf Objektidentität für das Auffinden der übergebenen Bank ist ausreichend.

Aufgabe 3 - RemoveValueFromBankVault**12 Punkte**

Implementieren Sie die Methode `removeValueFromBankVault(Bank bankToFind, Long vaultID, int valueToRemove)` der Klasse `VaultingManager`. Die Methode soll bei dem Vault mit der ID `vaultID` das erste Element mit dem Wert `valueToRemove` löschen. Hierbei soll sichergestellt werden, dass der zu ändernde Vault in der Liste der Vaults liegt, die der übergebenen Bank `bankToFind` zugeordnet sind. Sollte dies nicht der Fall sein, so soll keine Änderung vorgenommen werden und `false` zurückgeliefert werden. Sollte der gesuchte Vault der übergebenen Bank zugeordnet sein, so soll dann `true` zurückgeliefert werden, wenn sich der Wert des Vaults durch den Löschauftrag ändert. Ansonsten soll `false` zurückgegeben werden.

Hinweis: Die Methode `findBankTuple` kann zur Überprüfung der Zuordnung von Vault mit ID und übergebener Bank verwendet werden.

