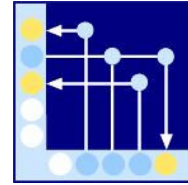




Hochschule Aalen

*Fakultät Elektronik und Informatik
Studienbereich Informatik*



Objektorientierte Programmierung

Vorlesung im Wintersemester 2023/2024

Prof. Dr. habil. Christian Heinlein

1. Übungsblatt (19. – 25. Oktober 2023)

Aufgabe 1: Gebührenpflichtige Konten

Erweitern Sie das Konto-Beispiel der Vorlesung (Skript, § 2.1 und § 2.2) wie folgt um gebührenpflichtige Konten:

- Definieren Sie eine Datenstruktur `struct ChargedAccount`, die alle Datenfelder von `struct Account` „erbt“ und zusätzlich ein Datenfeld `count` besitzt, in dem die Anzahl der ausgeführten Buchungen pro Konto gespeichert wird (Anfangswert 0)!
- Implementieren Sie Funktionen `newChargedAccount` und `initChargedAccount` zur Erzeugung und Initialisierung gebührenpflichtiger Konten! `newChargedAccount` soll als Parameter den Kontoinhaber erhalten.
- „Überschreiben“ Sie die Funktionen `deposit` und `withdraw` für gebührenpflichtige Konten so, dass bei jeder solchen Buchung das Datenfeld `count` um eins erhöht wird! Zur Ausführung der eigentlichen Buchung soll die entsprechende Funktion für gewöhnliche Konten aufgerufen werden.
(Die Funktion `transfer` muss nicht überschrieben werden. Warum?)
- Implementieren Sie eine Funktion `charge`, die als Parameter ein gebührenpflichtiges Konto sowie einen Centbetrag erhält und für jede bis jetzt auf diesem Konto ausgeführte Buchung diesen Betrag als Buchungsgebühr vom aktuellen Kontostand abzieht! (Für diesen Vorgang selbst wird jedoch keine Gebühr abgezogen.) Anschließend soll der Buchungszähler `count` auf 0 zurückgesetzt werden, um die bis jetzt ausgeführten Buchungen zu „vergessen“.
- Schreiben Sie, analog zu § 2.1.7 und § 2.2.8, ein Hauptprogramm, in dem auf gewöhnlichen, limitierten und gebührenpflichtigen Konten diverse Buchungen ausgeführt werden, u. a. Überweisungen zwischen unterschiedlichen Arten von Konten! Verfolgen Sie durch Ausgabeanweisungen jeweils den genauen Programmablauf! Rufen Sie am Schluss für jedes gebührenpflichtige Konto die Funktion `charge` mit Parameterwert 10 Cent auf und geben Sie dann alle Kontostände aus!

Orientieren Sie sich bei der Lösung dieser Aufgabe an der Implementierung limitierter Konten (Skript, § 2.2)!

Der Code zur Implementierung gewöhnlicher und limitierter Konten aus der Vorlesung steht auf der Vorlesungs-Webseite zur Verfügung und kann übernommen werden.