

Prüfung Programmieren II, WS 2013/14

Datum, Uhrzeit: 04.02.2014, 11:00 Uhr

Semester: WS 2013/14

Prüfungsnummern: 3910071, 4610071, 4710061

Prüfer: Prof. Dr. Thomas Wölfl, Prof. Dr. Jan Dünneweber

Dauer: 90 Minuten

Hilfsmittel: Keine

Die Prüfung umfasst 9 Seiten (bitte sofort nachprüfen). Die Lösungen für die Aufgaben sind auf den Aufgabenblättern einzutragen. Sie können auch die Rückseiten verwenden. Achten Sie bitte darauf, dass auf jedem Blatt Ihr Name und Ihre Immatrikulationsnummer vermerkt sind. Ist eine Aufgabenstellung Ihrer Meinung nach nicht vollständig oder mehrdeutig, so treffen Sie entsprechende Annahmen.

Prüfungsbewertung:

Aufgabe	1	2	3	4	Gesamt
Ergebnis					

Prüfungsergebnis:

Note	
------	--

1. Prüfer

2. Prüfer

f) Was bedeutet es, wenn eine Methode als „deprecated“ markiert ist?

g) Erklären Sie das Konzept des Zeitscheiben-Multiplexings mit Hilfe eines Beispiels.
Welches Ziel wird damit erreicht?

Aufgabe 2 (Klassenmodell, Programmierung)

Es soll eine Bibliotheksverwaltungssoftware erstellt werden. Die Software verwaltet Bücher und DVDs als Medien, die ausgeliehen werden können. Bücher und DVDs haben einen Titel und einen Einkaufspreis, der entweder in Euro oder in US-Dollar (nicht beides gleichzeitig) hinterlegt wird. Nur Bücher haben zusätzlich eine ISBN Nummer (eine Integer-Zahl). Nur DVDs haben zusätzlich eine Spieldauer (Länge des Filmes).

Die Benutzer der Bibliothek können Medien ausleihen. Jeder Ausgabevorgang enthält den Benutzer, welcher die Medien ausleiht, das Datum des Vorganges und die Medien, die ausgeliehen werden. Ein Benutzer hat eine Nummer und einen Namen.

Jedes Medium gibt es nur ein einziges Mal in der Bibliothek. Für jedes Medium ist bekannt, ob es gerade vorhanden und damit ausleihbar ist, oder ob es bereits ausgeliehen und damit nicht ausleihbar ist.

Es soll möglich sein, einen neuen Ausleihvorgang anzulegen, hierfür einen Benutzer zuzuordnen und Medien in den Ausleihvorgang einzubuchen. Ein Medium kann nur dann eingebucht werden, wenn es vorhanden (also nicht bereits ausgeliehen) ist.

- a) Erstellen Sie ein vereinfachtes Klassenmodell mit Attributen, Methoden und Datentypen, das für diese Programmieraufgabe geeignet ist.

- b) Programmieren Sie die Methode, welche das Einbuchen eines einzelnen Mediums im Rahmen eines Ausleihvorganges umsetzt. Wenn ein Medium eingebucht (d. h. ausgeliehen) werden soll, das nicht vorhanden ist, so gibt diese Methode eine `MediumNotAvailable` Exception zurück. Ergänzen Sie auch den Quellcode für diese Exception und arbeiten Sie dabei mit einer Checked Exception.
- c) Programmieren Sie eine Methode in der passenden Klasse, welche den Gesamt-Einkaufspreis aller im Rahmen eines Ausleihvorganges eingebuchten Medien in Euro und in US-Dollar (getrennt voneinander) ausgibt. Dafür werden alle Preise der eingebuchten Medien mit Euro-Preisen und alle Preise der eingebuchten Medien mit US-Dollar-Preisen summiert.

- d) Ein Benutzer kann nur dann existieren, wenn er einen Namen hat. Wie sichern Sie diese Eigenschaft in Ihrem Quellcode in der entsprechenden Klasse ab? Ergänzen Sie den hierfür nötigen Quellcode.

- e) Ergänzen Sie eine Klasse mit dem Namen Katalog. Diese Klasse enthält eine Liste aller Medien, die es in der Bibliothek gibt (d. h. ausgeliehene und nicht ausgeliehene Medien). Programmieren Sie den Quellcode für diese Klasse und ergänzen Sie eine Methode, die alle Medien alphabetisch absteigend (von „Z nach A“) nach dem Titel des Mediums sortiert und in dieser Reihenfolge in der Command-Line ausgibt.
Hinweis: Es bietet sich an, hierfür die Klasse Medium aus Aufgabe a) geeignet anzupassen. Nennen Sie hier auch die nötigen Änderungen der Klasse Medium.

Aufgabe 3 (Codeverständnis)

- a) Beschreiben Sie die Hauptaufgabe des folgenden Java-Programms in ca. 2 – 3 Sätzen. Nennen Sie außerdem einen Aufruf dieses Programms mit gültigen Parametern in der Command-Line.

```
1 import java.util.ArrayList;
2 import java.util.Collections;
3 import java.util.List;
4
5 public class WasMacheIch {
6
7     public static void main(String[] args) {
8         if (args.length < 1){
9             System.out.println("Kein Parameter angegeben");
10            return;
11        }
12        int x = 0;
13        try {
14            x = Integer.parseInt(args[0]);
15        } catch (NumberFormatException e){
16            System.out.println("Ungültiger Parameter angegeben");
17            return;
18        }
19        List<Integer> y = new ArrayList<Integer>();
20        for (int i = 1; i < 50; i++){
21            y.add(i);
22        }
23        Collections.shuffle(y);
24        for (int i = 0; i < x; i++){
25            System.out.println("Zahl " + (i + 1) + ": " + y.get(i));
26        }
27    }
28 }
```

- b) In der folgenden Klasse sind zwei Zugriffe auf Variablen nicht möglich. Nennen Sie diese beiden Variablen und begründen Sie Ihre Wahl.

```
1 public class MyClass {  
2  
3     private String currentArgument;  
4  
5     public static void main(String[] args) {  
6  
7         for ( int i = 0; i < args.length; i++ ) {  
8             currentArgument = args[i];  
9         }  
10  
11         if ( i > 3 ) {  
12             System.out.println("i > 3");  
13         } else {  
14             System.out.println("i <= 3");  
15         }  
16     }  
17 }
```


Aufgabe 4 (Threads)

- a) Programmieren Sie eine Klasse mit dem Namen Counter, die sich als Thread starten lässt. Wird ein solcher Thread gestartet, gibt er in der Command-Line die Zahlen von 1 bis 500.000 aus. Wenn mehrere Threads dieser Klasse gleichzeitig laufen, darf keine Zahl mehrfach ausgegeben werden. Es darf aber auch keine Zahl ausgelassen werden. Sichern Sie diese Eigenschaft in Ihrem Quellcode ab.
Bemerkung: Sie müssen die Threads nicht starten.