

Prüfung Programmieren II, WS 2014/15

Datum, Uhrzeit: 05.02.2015, 11:00 Uhr

Semester: WS 2014/15

Prüfungsnummern: 3910071, 4610071, 4710061, 2410030 (SB)

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

Dauer: 90 Minuten

Hilfsmittel: Keine

Die Prüfung umfasst 8 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

Aufgabe 2 (Klassenmodell, Programmierung)

An einem Post-Packstationsautomaten können Kunden hinterlegte Sendungen abholen. Eine Sendung kann dabei ein Brief oder ein Paket sein. Diese werden zu einem bestimmten Zeitpunkt in die Fächer der Station eingelegt. Dieser Zeitpunkt wird bei der Sendung gespeichert. Für jeden Brief werden eine eindeutige Nummer und das Gewicht gespeichert. Bei Paketen wird neben dem Gewicht und der eindeutigen Nummer auch zusätzlich das Volumen des Paketes gespeichert.

Die Station hat genau 30 Fächer. Jedes Paket bzw. jeder Brief wird in ein eigenes Fach der Station gelegt. Beim Hinterlegen in der Station wird dem Post-Mitarbeiter ein beliebiges freies Fach vorgeschlagen, sofern noch freie Fächer vorhanden sind. Größenbeschränkungen werden hier nicht betrachtet.

Für jede Sendung wird weiterhin eine zufällig erzeugte Zahl als Passwort gespeichert, sobald die Sendung in ein Fach der Station gelegt wird. Dieses Passwort muss der Kunde nennen, um die Sendung aus der Station entnehmen zu können. Für die vorliegende Aufgabe können Sie annehmen, dass die Kunden das Passwort per SMS erhalten.

Von jeder Sendung ist bekannt, welchem Kunden die Sendung gehört. Ein Kunde hat einen Namen, eine Kundennummer und eine Telefonnummer.

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

- b) Programmieren Sie die Methode(n), welche das Einlegen einer Sendung in die Station abbildet bzw. abbilden. Dabei soll das zufällige Passwort erzeugt und bei der Sendung gespeichert werden.
- c) Ergänzen und programmieren Sie eine Methode, die alle Sendungsnummern zusammen mit dem Namen des jeweiligen Kunden in der Command-Line anzeigt, die zum Zeitpunkt des Aufrufes der Methode länger als 5 Tage in der Station liegen.

- d) Programmieren Sie weiterhin eine Methode, welche den Benutzer in der Command-Line dazu auffordert, das Passwort einer Sendung einzugeben. Die Methode sucht die zum Passwort passende Sendung in der Station und entfernt sie aus der Station, sofern sie gefunden wird. Wenn keine passende Sendung gefunden wird, erhält der Benutzer eine entsprechende Meldung.

Anmerkung: Sie können davon ausgehen, dass jedes erzeugte Zufallspasswort nur ein einziges Mal existiert.

Aufgabe 3 (Codeverständnis)

- a) Beschreiben Sie die Hauptaufgabe des folgenden Java-Programms in zwei bis drei Sätzen. Nennen Sie eine Beispieleringabe und das zugehörige Ergebnis.

```
1 import java.util.Scanner;
2
3 public class WasMacheIch {
4
5     private static int size = 2;
6     private static long[][] array;
7
8     public static void main(String[] args) {
9         array = new long[size][size];
10        Scanner sc = new Scanner(System.in);
11        for (int i = 0; i < size; i++) {
12            for (int j = 0; j < size; j++) {
13                System.out.println("Geben Sie Zahl " + i + "/" + j + " ein: ");
14                array[i][j] = sc.nextLong();
15            }
16        }
17        System.out.print("Ergebnis: ");
18        System.out.println(anonymous(array));
19    }
20
21    private static float anonymous(long[][] array) {
22        long result = 0;
23        for (int i = 0; i < array.length; i++) {
24            long curNumber = array[i][i];
25            if (curNumber % 2 == 0) {
26                result += curNumber;
27            }
28        }
29        return result;
30    }
31 }
```

- b) In der folgenden Klasse sind zwei Fehler enthalten. Nennen Sie die beiden Fehler und begründen Sie Ihre Wahl.

```
1  import java.util.List;
2
3  public class Fehlerhaft {
4
5      public static void main(String[] args){
6
7          List<Integer> list = new int[10];
8
9          for (Integer element : list){
10             if (element > 10){
11                 list.remove(element);
12                 System.out.println("Removed");
13             }
14         }
15     }
16 }
```


Aufgabe 4 (Threads)

- a) Erklären Sie den Begriff der „quasi-parallelen“ Ausführung mehrerer Threads auf einem Einprozessorsystem.
- b) Programmieren Sie eine Klasse mit dem Namen „Test“, die sich als Thread starten lässt und 5000 Zufallszahlen in der Command-Line ausgibt. Ergänzen Sie eine main-Methode, welche zwei dieser Threads startet. Sobald beide Threads fertig sind, soll die Meldung „fertig“ in der Command-Line angezeigt werden.