



RWTH AACHEN
Lehrstuhl für Informatik X
Programmierung für Alle (Java)

Übung 6

Abgabe der Lösungszettel: 05. Dezember 2006 vor der Übung

Deadline für die Abgabe der Programme per *E-Mail* beim Tutor:
05. November 2006, 8:00 Uhr

Tragen Sie hier Ihre **Gruppennummer** ein:

Tragen Sie hier den **Namen Ihres Tutors** ein:

Die Abgabe der Übung erfolgt in **Dreiergruppen**. Tragen Sie dazu alle Namen und Matrikelnummern Ihrer Gruppenmitglieder hier ein. Der / Die erste in der Tabelle gibt die Übung beim Tutor ab.

NAMEN	MATRIKELNUMMERN

Punktetabelle für die Korrektur:

	Punkte
1 (10)	
2 (8)	
Total (18)	

Diese 6. Übung schließt den ersten Übungsblock ab. Sie müssen in allen Übungen 1-6 zusammen mindestens 50% der Übungspunkte erreichen, damit Sie zu der Schein-/Bachelorklausur Teil 1 am 15. Dezember 2006 zugelassen werden.

Die Anmeldung zu der Teilklausur 1 erfolgt für alle Studiengänge über die Lehrstuhlwebseite zwischen dem 04. – 08. Dezember 2006 um 17:00 Uhr.

Bachelorstudenten aller Studiengänge müssen sich bereits im Dezember für die Bachelorklausur Teil 2 am 13. Februar 2007 anmelden.

Die Anmeldung zu der Vordiplom-/Zwischenprüfungsklausur am 27. März 2007 findet ebenfalls im Dezember statt.

Beachten Sie bitte unbedingt die Anmeldetermine beim ZPA bzw. auf der Lehrstuhlwebseite.

Die Ergebnisse der Schein-/Bachelorklausur Teil 1 werden für diejenigen Studenten, die sich für die Bachelorklausur Teil 2 und für die Vordiplom-/ Zwischenprüfungsklausur beim ZPA anmelden müssen, an das ZPA weitergeleitet.

1. **Wilde Tiere** Teil3: (Programmieraufgabe)

Der Direktor löst nun *seine dritten Wünsche* ein:

- (a) [1 Punkt] Die Klasse Zoo und die spezifischen Tier-Unterklassen sind noch in einer einzigen Datei enthalten und können nur innerhalb eines Paketes benutzt werden (wie wir Pakete erstellen ist jetzt nicht wichtig). Es ist an der Zeit, jede der Klassen für alle anderen in der Welt öffentlich zu machen und jeweils in einer eigenen Dateien zu schreiben. Die spezifischen Unterklassen von Tier sollen außerdem von überall die geerbten Instanzvariablen direkt verändern können. Und dann ist noch etwas ganz Furchtbares am Wochenende passiert! Der Zoo-Direktor hatte einen furchtbaren Albtraum (über den er eigentlich nicht so offen sprechen möchte): Er hatte abstrakte Tiere in seinem Zoo herumlaufen! Oh Mann, wie er sich erschrocken hat. Da hilft nur eins: Lassen Sie abstrakte Tiere niemals in unserem Zoo vorkommen!
Tipp: In der letzten Vorlesung haben wir *Zugriffsmodifier* kennengelernt (Folie 10 und 11 in 05-Kap7.pdf).
- (b) [1 Punkt] Setter-Methoden sollen die Instanzvariablen mit sinnvollen Werten initialisieren. Bauen Sie Tests ein, damit das Alter eines Tieres auf 0 gesetzt wird, wenn ein ungültiges Alter eingegeben wird. Die Methode *length()* der Klasse *String* liefert die Länge eines Strings zurück. Initialisieren Sie String-Instanzvariablen mit “unbekannt”, falls die Setter-Methoden ein leeres String-Argument erhalten. Das Geschlecht eines Tieres sollte z.B. “maennlich”, “weiblich” oder “unbekannt” sein. Passen Sie den Rest des Programmes entsprechend an.

- (c) [1 Punkt] Erweitern Sie die Klasse `Tier`, damit alle Instanzvariablen für ein Tier vor überall aus mit nur einem Methodenaufruf gesetzt werden können. Benutzen Sie innerhalb dieser Methode die “sicheren” Setter-Methoden.
Tipp: Schreiben Sie dazu eine Methode *initialisieren* (*Instanzvariable1*, *Instanzvariable2*, ...).
- (d) [1 Punkt] Wilde Tiere machen Geräusche, aber unsere Tiere sind noch stumm. Erweitern Sie alle Unterklassen von `Tier`, damit jedes Tier sein spezifisches Geräusch brüllt, schreit, zwitschert, etc. (mit anderen Worten: am Bildschirm ausgibt).
- (e) [3 Punkte] In der letzten Übung haben Sie auf Wunsch des Zoo-Direktors alle vier Tiere in ein Array gesteckt und damit endlich für Ordnung im Zoo gesorgt. Erweitern Sie das Programm, damit beliebig viele Tiere aller vier Tierarten dynamisch hinzugefügt werden können. Setzen Sie für jedes neue Tier die geerbten Instanzvariablen mit der vorher geschriebenen Initialisierungsmethode. Sobald das Programm startet, sollen unsere bisherigen vier Tiere wie immer automatisch dem Zoo hinzugefügt werden.
Tipp: Benutzen Sie eine private `ArrayList`-Instanzvariable *alleTiere*.
- (f) [3 Punkte] Viele Zoo-Besucher sind großzügig und möchten dem Direktor ein Tier schenken. Erweitern Sie das Programm, damit Tiere nicht nur berührt, sondern auch an den Zoo verschenkt werden können. Schreiben Sie für den Zoo eine Methode *public void neuzugang()*, die den Besucher nach der Art des Tieres und nach den Instanzvariablen des Tieres fragt. Sobald ein Tier der passenden Art dem Zoo geschenkt und mit passenden Werten initialisiert wurde (über die Initialisierungsmethode), wird es in die `ArrayList` *alleTiere* mit aufgenommen. Fremde Tiere, wie z.B. Elefanten, können wir leider noch nicht aufnehmen. Wenn sich im Zoo nach einiger Zeit z.B. zwei Affen befinden und ein Zoo-Besucher einen Affen berührt, dann sollen alle Affen aus dem Zoo nacheinander über sich erzählen und artgerecht Geräusche machen.
Tipp: Um nicht auf Groß-/Kleinschreibung zu achten, können Sie auch die Methode *boolean equalsIgnoreCase* (*String anotherString*) der Klasse `String` benutzen.

2. Lotto Teil 2: (Programmieraufgabe)

Benutzen Sie für die Lösung das beigelegte Programmgerüst.

- (a) [3 Punkte] Die Lottogesellschaft hat bemerkt, dass einige Spielertipps doppelte Zahlen enthalten. Schreiben Sie für die Lottogesellschaft eine Methode *pruefeDoppelteZahlen (...)*, die den Tipp eines Spielers erwartet und entscheidet, ob der Tipp doppelte Zahlen enthält. Selbstverständlich darf die Lottogesellschaft auch keine doppelten Gewinnzahlen ziehen!
- (b) [2 Punkte] Ändern Sie das LottoSpiel, damit nur noch Tipps im gültigen Wertebereich und ohne doppelte Zahlen in die *ArrayList alleTipps* aufgenommen werden. Geben Sie einen entsprechenden Hinweis aus, wenn der Tipp eines Spielers doppelte Zahlen enthält. Weisen Sie den Spieler ebenfalls darauf hin, wenn sein Tipp Zahlen außerhalb des gültigen Wertebereiches enthält.
- (c) [3 Punkte] Was nützt eine Lottogesellschaft, die nicht die Gewinner ermittelt? Schreiben Sie eine Methode *getAnzahlTreffer(...)*, die den Tipp eines Spielers erwartet und die Anzahl der Treffer als int-Wert zurückliefert. Nach der Ziehung der Gewinnzahlen sollten alle Spieler, die mindestens 3 Treffer haben, benachrichtigt werden.
Tipp: Um die Implementierung von *getAnzahlTreffer(...)* zu vereinfachen, könnte die LottoGesellschaft z.B. auf eine private Methode *zahlIstVorhanden (...)* zurückgreifen, die für eine beliebige int-Zahl entscheidet, ob sie in den gezogenen Gewinnzahlen enthalten ist oder nicht.

Hinweise zur Übungsabgabe

Bitte geben Sie zu jeder Übung das *Deckblatt*, angeheftet vor Ihren Lösungen, mit ab. *Programmieraufgaben* werden per E-Mail **und** der ausgedruckte Code vor Übungsbeginn beim Tutor abgegeben. *Verständnisaufgaben* werden **handschriftlich** beim Tutor abgegeben.

Sollten Sie Probleme oder Fragen haben, wenden Sie sich bitte an Ihren Tutor oder besuchen Sie die *Betreuungsstunden* im Lila Raum (4U15) im ZIP-Pool der Informatik: Donnerstag von 16:00 bis 18:00 und Freitag von 10:00 bis 12:00.

Häufige Fragen finden Sie auf der *FAQ Seite*:

http://media.informatik.rwth-aachen.de/programmierung_faq.html