

Übungsblatt 9

Ausgabe: Mi, 22.12.99

Abgabe: Di, 11.01.00, 18 Uhr

Aufgabe 1: Verkettete Listen [10 Punkte]

Implementieren Sie eine Adressverwaltung unter Verwendung einer verketteten Liste zur Speicherung der Adressen. Es gelten die Anforderungen aus Aufgabe 1b), Übungsblatt 7, mit folgender Ausnahme: Die Elemente innerhalb der Adressenliste sollen zu jedem Zeitpunkt nach Namen aufsteigend sortiert sein, d.h. beim Einfügen von Adressen wird die Sortierung berücksichtigt. Das `print`-Kommando liefert dementsprechend eine sortierte Ausgabe der Adressen.

Hinweise: Verwenden und erweitern Sie folgende Klassendeklarationen:

```
class Adresselement {
    ... // siehe Aufgabe 1, Blatt 7
}

class Listenelement {
    public Listenelement nachfolger = null; // Referenz auf das nachfolgende
                                           // Listenelement
    public Adresselement adresse = null;   // Adressdaten
}

public class AdressenListeVListe {
    private Listenelement listenAnfang = null; // Referenz auf das erste
                                               // Element der Liste
    private Listenelement listenEnde = null; // Referenz auf das letzte
                                               // Element der Liste
    ...
    // Methoden einfüegen, loeschen, suchen
    ...
    public static void main (String[] args) {
        ...
    }
}
```

Speichern Sie alle drei Klassen in der Datei `AdressenListeVListe.java` (dieses ist möglich, da nur eine Klasse `public` deklariert wurde) und geben Sie Ihre Implementierung per `abox` ab. Für Tests können Sie die auf Blatt 7 verwendete Datei `kommandos.txt`¹ verwenden.

Aufgabe 2: Binärbäume [10 Punkte]

Implementieren Sie eine Adressverwaltung unter Verwendung eines Binärbaumes zur Speicherung der Adressen. Es gelten die Anforderungen aus Aufgabe 1b), Übungsblatt 7. Ausnahme: Das `print`-Kommando soll eine nach Namen aufsteigend sortierte Ausgabe der Adressen liefern.

¹<http://www.informatik.uni-mannheim.de/informatik/pi4/stud/veranstaltungen/ws199900/pi1/ueb/blatt7/kommandos.txt>

Hinweise: Verwenden und erweitern Sie folgende Klassendeklarationen:

```
class AdressElement {
    ... // siehe Aufgabe 1, Blatt 7
}

class BaumElement {
    public BaumElement linkesKind = null; // Verkettung mit den
    public BaumElement rechtesKind = null; // nachfolgenden Elementen
    public AdressElement adresse = null; // Adressdaten
}

public class AdressenListeBBaum {
    private BaumElement wurzel = null; // Referenz auf die Wurzel des
                                        // binaeren Baums

    ...
    // Methoden einfuegen, loeschen, suchen
    ...
    public static void main (String[] args) {
        ...
    }
}
```

Speichern Sie alle drei Klassen in der Datei `AdressenListeBBaum.java` und geben Sie Ihre Implementierung per abox ab. Für Tests können Sie die bereits auf Blatt 7 verwendete Datei `kommandos.txt` verwenden.