



4. Übungsblatt zur Vorlesung Einführung in die Informatik

(In dieser Übung ist ein Teil der Aufgaben in natürlicher Sprache und/oder mit Diagrammen zu beantworten. Bitte machen Sie auch diese Abgaben per daphne/svn, zum Beispiel durch leserliches Abfotografieren/Scannen, oder mit einem geeigneten Programm. Akzeptierte Formate sind .jpg, .pdf und .txt.)

Aufgabe 1: Verkettete Liste

Betrachten Sie folgenden Code:

```
import Prog1Tools.IOTools;

public class List {
    public static void main(String[] params) {
        String nodeContent = null;
        ListNode listNode = new ListNode();
        ListNode listHead = listNode;
        while (true) {
            nodeContent = IOTools.readString("Bitte geben Sie einen Text als
            Inhalt des nächsten Knotens ein (oder 'ende' wenn Sie aufhören
            wollen) ");
            if (nodeContent.equals("ende")) {
                break;
            }
            listNode.nextNode = new ListNode();
            listNode.nextNode.contents = nodeContent;
            listNode = listNode.nextNode;
        }

        listNode = listHead;
        while (listNode.nextNode != null) {
            listNode = listNode.nextNode;
            System.out.println(listNode.contents);
        }
    }
}
```

```

class ListNode {
    public ListNode nextNode;
    public String contents;
}

```

Was tut dieses Programm, wenn es ausgeführt wird? Zeichnen Sie den Zustand der Objekte und ihrer Beziehungen am Programmpunkt direkt vor der zweiten While-Schleife.

Aufgabe 2: Vererbung

Entwerfen Sie ein Klassendiagramm, das eine einfache geografische Datenbank zu beschreibt. Diese soll folgende Eigenschaften der Welt abbilden:

Es gibt Koordinaten. Eine Koordinate beschreibt (zweidimensional) eine Position auf der Karte. Es gibt Straßen. Straßen haben Namen. Einer Straße ist eine Liste von Koordinaten zugeordnet, die ihren Verlauf beschreiben. Außerdem soll gespeichert werden, wo Straßen aufeinandertreffen. An Straßen stehen Gebäude. Ein Gebäude kann ein Wohnhaus, ein Bürogebäude oder ein Industriegebäude sein. Jedes Gebäude hat einen Eingang. Jeder Eingang hat eine Hausnummer. Jedem Eingang ist eine Koordinate zugeordnet.

Ein Benutzer soll durch Eingabe einer Adresse der Form `<Straße> <Hausnummer>`, herausfinden können, an welcher Koordinate sich diese Adresse befindet und welche Art von Gebäude sich dort befindet. Außerdem soll es möglich sein, auf Basis der von Ihren Klassen bereitgestellten Daten eine Route zwischen zwei Adressen zu berechnen. Weiterhin soll, wenn der Benutzer eine Koordinate eingibt, angezeigt werden können, was sich dort befindet.

Aufgabe 3: Laufzeit

Schreiben Sie eine Klasse `LittleGauss.java`, das die ersten n positiven natürlichen Zahlen aufsummiert, und zwar auf drei verschiedene Arten:

- per Rekursion (also wie im vorigen Übungsblatt)
- mit einer Schleife
- mithilfe der Formel $\sum_{i=1}^n i = \frac{n \cdot (n+1)}{2}$

Lagern Sie jede Berechnung in eine eigene Funktion aus. Messen Sie, wie lange die Berechnung für $n = 10^i$ dauert, für $i \in \{1, \dots, 9\}$. Notieren Sie ihre Beobachtungen in einem Kommentar im Quelltext. (Benutzen sie die Methode `System.nanoTime()` für Ihre Messungen.)