21 Wörterbuch als sortierte lineare Liste

```
class Dictionary {
    public static void main(String[] args) {
        Dictionary dict = new Dictionary();

        dict.insert("better", "besser");
        Out.println(dict.toString());

        dict.insert("apple", "Apfel");
        Out.println(dict.toString());

        dict.insert("deft", "geschickt");
        Out.println(dict.toString());

        dict.insert("control", "Kontrolle");
        Out.println(dict.toString());

        Out.format("Lookup 'better': %s%n", dict.lookup("better"));
    }

    private Entry head = null;

    void insert(String term, String translation) {
        Entry cur = head, prev = null;
        Entry e = new Entry(term, translation);

        while (cur != null && term.compareTo(cur.term) > 0) {
            prev = cur;
            cur = cur.next;
        }

        if (cur != null && cur.term == term)
            return;

        if (prev == null) {
            head = e;
        } else prev.next = e;

        e.next = cur;
    }

    void delete(String term) {
        Entry cur = head, prev = null;

        while (cur != null && !term.equals(cur.term)) {
            prev = cur;
            cur = cur.next;
        }

        if (cur == null)
            return;
```

```
51
52          prev.next = cur.next;
53      }
54
55      String lookup(String term) {
56          Entry cur = head;
57
58          while (cur != null && !term.equals(cur.term)) {
59              cur = cur.next;
60          }
61
62          return (cur != null ? cur.translation : null);
63      }
64
65      @Override
66      public String toString() {
67          Entry cur = head;
68          StringBuilder builder = new StringBuilder();
69
70          while (cur != null) {
71              builder.append(cur.toString() + '\n');
72
73              cur = cur.next;
74          }
75
76          return builder.toString();
77      }
78
79      private static class Entry {
80          String term;
81          String translation;
82
83          Entry next = null;
84
85          Entry(String term, String translation) {
86              this.term = term;
87              this.translation = translation;
88          }
89
90          @Override
91          public String toString() {
92              return this.term + " - " + this.translation;
93          }
94      }
95  }
```

```
$ java Dictionary
better - besser

apple - Apfel
better - besser
```

```
apple - Apfel
better - besser
deft - geschickt

apple - Apfel
better - besser
control - Kontrolle
deft - geschickt

Lookup 'better': besser
```

22 Bibliotheksbestand als binärer Suchbaum

```
ffinal class Library {
    public static void main(String args[]) {
        final StockTree st = new StockTree();

        In.open(args[0]);

        while (!In.isEof()) {
            int amount = In.readInt();
            st.insert(In.readLine(), amount);
        }

        st.printOrdered();
    }
}

final class StockTree {
    static final class Node {
        Stock stock;
        Node left = null, right = null;

        Node(Stock stock) {
            this.stock = stock;
        }

        public void insert(Node node) {
            final int order = node.stock.getTitle()
                .compareTo(this.stock.getTitle());

            if (order < 0) {
                if (left == null)
                    left = node;
                else
                    left.insert(node);
            } else if (order > 0) {
                if (right == null)
                    right = node;
                else
                    right.insert(node);
```

```
39              } else {
40                  this.stock.increment(1);
41              }
42          }
43
44          public Stock find(String title) {
45              final int order = title.compareTo(this.stock.getTitle());
46
47              if (order == 0)
48                  return this.stock;
49
50              if (order < 0) {
51                  if (left != null)
52                      return left.find(title);
53              } else {
54                  if (right != null)
55                      return right.find(title);
56              }
57
58              return null;
59          }
60
61          @Override
62          public String toString() {
63              return String.format("%d %s", this.stock.getAmount(),
64                  this.stock.getTitle());
65          }
66      }
67
68      Node root = null;
69
70      public void insert(String title, int amount) {
71          final Node nn = new Node(new Stock(title, amount));
72
73          if (root == null)
74              root = nn;
75
76          root.insert(nn);
77      }
78
79      public int amount(String title) {
80          final Stock item = root.find(title);
81
82          return item == null ? 0 : item.getAmount();
83      }
84
85      public void printOrdered() {
86          printOrdered(root);
87      }
88
89      private void printOrdered(Node node) {
90          if (node.left != null)
```

```
91              printOrdered(node.left);
92
93          Out.println(node.toString());
94
95          if (node.right != null)
96              printOrdered(node.right);
97      }
98  }
99
100 final class Stock {
101     private int amount;
102     private final String title;
103
104     public Stock(String title, int amount) {
105         this.title = title;
106         this.amount = amount;
107     }
108
109     public String getTitle() {
110         return title;
111     }
112
113     public int getAmount() {
114         return amount;
115     }
116
117     public void increment(int value) {
118         amount += value;
119     }
120
121     public String toString() {
122         return String.format("[%s, %d]", title, amount);
123     }
124 }
```

```
$ java Dictionary input.txt # the example input given in the exercise
3  A Terrible Revenge
7  Analytic Philosophy
3  Disaster Law and Policy
2  Eternity's Sunrise
6  HSPT Flashcard Study System
5  Hunting the Rockies
2  It's St. Patrick's Day
1  Life Reimagined
9  Microbiologically Safe Foods
10  Mobile & Social Game Design
10  On the Backroad to Heaven
1  Paraguay (Bradt Travel Guide)
2  Plastics
10  Promises Kept
6  Ready, Freddy!
6  Strands of Sorrow
```

9 The Modern Coral Reef Aquarium
5 The Slums of Aspen
5 Treating Trauma and Traumatic Grief
3 Vocabulary for the College Bound Student