

Autoren: Marius Birk
Pieter Vogt
Tutor: Florian Brandt

Abgabe: 16.06.2020, 12:00 Uhr

Smileys:

A1	A2	A3	Σ
2	6.5	4.5	13

Objektorientierte Modellierung und Programmierung

Abgabe Übungsblatt Nr.08

(Alle allgemeinen Definitionen aus der Vorlesung haben in diesem Dokument bestand, es sei den sie erhalten eine explizit andere Definition.)

Aufgabe 1 / Aufgabe 2

```
1  import java.io.*;
2  import java.nio.charset.Charset;
3  import java.util.ArrayList;
4  import java.util.List;
5
6  public class Lecture {
7      private String number = "";
8      private String title = "";
9      private String shortTitle = "";
10     private String semester = "";
11     private List<Lecturer> lecturers = new ArrayList<>();
12     private List<Date> schedule = new ArrayList<>();
13
14     public Lecture(String number, String title, String
        shortTitle, String semester) {
15         super();
16         this.number = number;
17         this.title = title;
18         this.shortTitle = shortTitle;
19         this.semester = semester;
20     }
21
22     public String getNumber() {
23         return number;
24     }
25
26     public void setNumber(String number) {
27         this.number = number;
28     }
29
30     public String getTitle() {
31         return title;
32     }
33
34     public void setTitle(String title) {
```

```
35     this.title = title;
36 }
37
38 public String getShortTitle() {
39     return shortTitle;
40 }
41
42 public void setShortTitle(String shortTitle) {
43     this.shortTitle = shortTitle;
44 }
45
46 public String getSemester() {
47     return semester;
48 }
49
50 public void setSemester(String semester) {
51     this.semester = semester;
52 }
53
54 public List<Lecturer> getLecturers() {
55     return lecturers;
56 }
57
58 public List<Date> getSchedule() {
59     return schedule;
60 }
61
62 @Override
63 public String toString() {
64     StringBuilder result = new StringBuilder();
65     result.append(number);
66     result.append(": ");
67     result.append(title);
68     result.append(" (");
69     result.append(shortTitle);
70     result.append("), ");
71     result.append(semester);
72     result.append("\n\t");
73     for (int i = 0; i < lecturers.size(); i++) {
74         if (i > 0) {
75             result.append(", ");
76         }
77         result.append(lecturers.get(i));
78     }
79     for (Date date : schedule) {
80         result.append("\n\t- ");
81         result.append(date);
82     }
83     result.append("\n");
```

```
84     return result.toString();
85 }
86 public static Lecture load(String filename) throws
    IOException {
87     Lecture result = null;
88     InputStream in = null;
89     try {
90         in = new FileInputStream(filename);
91         result = load(in);
92     } finally {
93         if (in != null) {
94             in.close();
95         }
96     }
97     return result;
98 }
99 public static Lecture load(InputStream in) throws
    IOException {
100     String number;
101     String title;
102     String shortTitle;
103     String semester;
104     StringBuilder builder = new StringBuilder();
105
106     int ch;
107     int point = 33;
108     boolean bool=false;
109     String compare="";
110     int[] replace = new int
        []{1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25
111
112     try{
113         while((ch = in.read()) != -1){
114             if(ch==0){
115                 continue;
116             }else{
117                 for(int i = 0; i<=replace.length-1;i++){
118                     if(ch==replace[i]){
119                         ch=33;
120                         bool=true;
121                     }else{
122                         bool=false;
123                     }
124                 }
125                 builder.append((char) ch);
126             }
127         } catch (IOException e) {
128             e.printStackTrace();
```

```
129     }
130     String[] split = builder.toString().split("!");
131     number =split[1];
132     title = split[2]+" "+split[3]+" "+split[4]+" "+split[5]+"
        "+split[6];
133     shortTitle= split[7];
134     semester= split[8];
135
136     Lecture lect = new Lecture(number, title, shortTitle,
        semester);
137
138     Lecturer snake = new Lecturer(split[10],split[11]);
139     lect.lecturers.add(snape);
140     Lecturer umbridge = new Lecturer(split[12], split[13]);
141     lect.lecturers.add(umbridge);
142     Lecturer lupin= new Lecturer(split[14], split[15]);
143     lect.lecturers.add(lupin);
144     return lect;
145 }
146
147 public static void saveText(String filename, Lecture data)
    throws IOException {
148     PrintWriter out = null;
149     try{
150         out = new PrintWriter(new FileOutputStream(filename));
151         out.print(data.getNumber()+"\n"+data.getTitle()+"\n"+data
            .getShortTitle()+"\n"+data.getSemester()+"\n"+data.
            getLecturers()+"\n"+data.getSchedule());
152     }catch(FileNotFoundException e){
153         e.printStackTrace();
154     }finally {
155         out.close();
156     }
157 }
158
159 public static Lecture loadText(String filename) throws
    IOException {
160     BufferedReader in = null;
161     Lecture lect= null;
162     try{
163         in=new BufferedReader(new FileReader(filename));
164         String zeile = null;
165         ArrayList<String> tmp = new ArrayList<>();
166         while((zeile=in.readLine())!=null){
167             tmp.add(zeile);
168         }
169         lect = new Lecture(tmp.get(0), tmp.get(1), tmp.get(2),
            tmp.get(3));
170         String[] tmp2 = tmp.get(4).split(",");
```



1) 2/5

a) 2.5/3

```

171     tmp2[0]=tmp2[0].substring(1);
172     tmp2[tmp2.length-1]=tmp2[tmp2.length-1].substring(0, tmp2
        [tmp2.length-1].length()-1);
173
174     for(int i = 0; i<=tmp2.length-1; i++){
175         String[] tmp3= tmp2[i].split(" ");
176         if(tmp3.length>2) {
177             lect.lecturers.add(new Lecturer(tmp3[tmp3.length-2],
                tmp3[tmp3.length-1]));
178         }else{
179             lect.lecturers.add(new Lecturer(tmp3[0], tmp3[1]));
180         }
181     }
182     }finally {
183         in.close();
184     }
185     return lect;
186 }
187 }

```



b) 4/6

Aufgabe 3

c) 0/1

Klasse A3

```

1  import java.io.BufferedReader;
2  import java.io.IOException;
3  import java.net.URL;
4  import java.net.URLConnection;
5
6  public class A3 {
7      public static void main(String[] args) throws IOException {
8          String url = "https://uol.de/en/computingscience/se/
                publications";
9          Connect connect = new Connect(url);
10         BufferedReader in = null;
11         in = connect.connect();
12         connect.count(in);
13     }
14 }

```

Klasse Connect

```

1  import java.io.*;
2  import java.net.URL;
3  import java.net.URLConnection;
4  import java.util.ArrayList;
5

```

```
6 public class Connect {
7     private String url;
8     private int proceeding;
9     private int artitel;
10    private int phdthesis;
11    private String input;
12    public Connect(String url) {
13        this.url =url;
14        proceeding =0;
15        artitel = 0;
16        phdthesis = 0;
17    }
18    public BufferedReader connect() throws IOException {
19        URL web = new URL(url);
20        BufferedReader in=null;
21        StringBuilder inputLine = new StringBuilder();
22        try{
23            URLConnection connect = web.openConnection();
24            in = new BufferedReader(new InputStreamReader(
25                connect.getInputStream()));
26        }catch(IOException e){
27            e.printStackTrace();
28        }finally{
29            return in;
30        }
31    }
32    }
33    public void count(BufferedReader in) throws IOException {
34        ArrayList<String> tmp = new ArrayList<>();
35        while((input=in.readLine())!= null){
36            tmp.add(input);
37        }
38        for(int i = 0; i<tmp.size();i++){
39            if(tmp.get(i).contains("inproceedings")){
40                proceeding+=1;
41            }
42            if(tmp.get(i).contains("[article]")){
43                artitel+=1;
44            }
45            if(tmp.get(i).contains("[phdthesis]")){
46                phdthesis+=1;
47            }
48        }
49        System.out.print("proceeding: "+proceeding+", article
50            : "+artitel+", phdthesis: "+phdthesis);
51    }
```