

1 Pair

1. Schreibe eine generische Klasse `Pair<T,U>`. Sie soll folgendermaßen verwendbar sein:

```
1  Brot brot = new Brot();
2  Wurst wurst = new Wurst();
3  Pair<Brot, Wurst> wurstbrot = new Pair<>(bro, wurst);
4  Brot broRet = wurstbrot.getFirst(); // == bro
5  Wurst wurstRet = wurstbrot.getSecond(); // == wurst
```

2. Schreibe eine Methode `flip()` in `Pair`, welche ein Paar retourniert, das dieselben Werte in umgekehrter Reihenfolge enthält. Wird ein `Pair<Brot, Wurst>` geflippt, entsteht ein `Pair<Wurst, Brot>`.
3. Implementiere folgendes Interface mit der Klasse `PairMap` und verwende intern eine `LinkedList<Pair<K,V>>`. Zwei gleiche Keys dürfen nicht gespeichert sein. Zum Vergleich der Schlüssel verwende die Methode `equals()`.
Teste deine Implementierung mit beiliegendem Unit-Test.

```
1  public interface IMap<K,V> {
2      / **
3       * Fuegt ein neues Key-Valuepaar dieser Map hinzu.
4       * Sollte der Key schon existieren, wird der Value ersetzt
5       * und der alte Value wird retourniert.
6       * @return Den alten Value bzw. null, wenn der Key
7       * neu oder key gleich null ist.
8       * /
9      V put(K key, V value);
10
11     / **
12     * Liefert den zum Key key gehoerigen Value bzw. null,
13     * wenn der Key key nicht existiert.
14     * @return Den zu key gehoerigen Value bzw. null
15     * /
16     V get(K key);
17
18     / **
19     * Liefert die Anzahl der Key-Valuee-Paare in dieser Map.
20     * @return Anzahl der Eintraege in dieser Map
21     * /
22     int size();
23 }
```

2 Gemeinsame Token

Schreibe Sie ein Programm, das die in zwei Textdateien **gemeinsam** vorkommenden Token genau einmal **aufsteigend sortiert** ausgibt. Als Delimiter sollen dabei Whitespacezeichen

und die Satzzeichen [,.;?!] gelten.

Schreibe zu diesem Zweck eine Methode `public static Collection<String> getToken(String filename)`, welche einen Dateinamen erhält und die eine geeignete Collection retourniert, in der alle in der übergebenen Textdatei vorkommenden Token **genau einmal** gespeichert sind. Führen Sie dann eine geeignete Mengenoperation aus.

Hinweis: Eine Textzeile kann mit Hilfe des regulären Ausdrucks `[\\s,\\..;?!]` zerlegt werden.