

## Aufgabe 7

- (a) Führen Sie „Sortieren durch Einfügen“ lexikographisch aufsteigend und *in-situ* (*in-place*) so in einem Schreibtischlauf auf folgendem Feld (Array) aus, dass gleiche Elemente ihre relative Abfolge jederzeit beibehalten (also dass z. B.  $A_1$  stets vor  $A_2$  im Feld steht). Jede Zeile stellt den Zustand des Feldes dar, nachdem das jeweils nächste Element in die Endposition verschoben wurde. Der bereits sortierte Teilbereich steht vor |||. Gleiche Elemente tragen zwecks Unterscheidung ihre „Objektidentität“ als Index (z. B. `"A1".equals("A2")` aber `"A1" != "A2"`)

L	A <sub>1</sub>	B <sub>1</sub>	F	A <sub>2</sub>	B <sub>2</sub>
---	----------------	----------------	---	----------------	----------------

Lösungsvorschlag

L	A <sub>1</sub>	B <sub>1</sub>	F	A <sub>2</sub>	B <sub>2</sub>
A <sub>1</sub>	L	B <sub>1</sub>	F	A <sub>2</sub>	B <sub>2</sub>
A <sub>1</sub>	B <sub>1</sub>	L	F	A <sub>2</sub>	B <sub>2</sub>
A <sub>1</sub>	B <sub>1</sub>	F	L	A <sub>2</sub>	B <sub>2</sub>
A <sub>1</sub>	A <sub>2</sub>	B <sub>1</sub>	F	L	B <sub>2</sub>
A <sub>1</sub>	A <sub>2</sub>	B <sub>1</sub>	B <sub>2</sub>	F	L

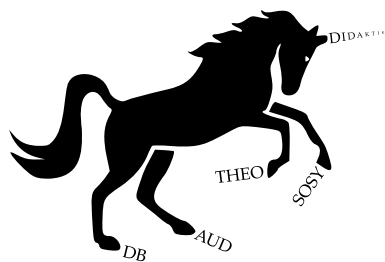
- (b) Ergänzen Sie die folgende Methode so, dass sie die Zeichenketten im Feld `a` lexikographisch aufsteigend durch Einfügen sortiert. Sie muss zum vorangehenden Ablauf passen, d. h. sie muss *iterativ* sowie *in-situ* (*in-place*) arbeiten und die relative Reihenfolge gleicher Elemente jederzeit beibehalten. Sie dürfen davon ausgehen, dass kein Eintrag im Feld null ist.

```
void sortierenDurchEinfuegen(String[] a) {
    // Hilfsvariable:
    String tmp;
}
```

Lösungsvorschlag

```
static void sortierenDurchEinfuegen(String[] a) {
    // Hilfsvariable:
    String tmp;
    for (int i = 1; i < a.length; i++) {
        tmp = a[i];
        int j = i;
        while (j > 0 && a[j - 1].compareTo(tmp) >= 1) {
            a[j] = a[j - 1];
            j = j - 1;
        }
        a[j] = tmp;
    }
}
```

Code-Beispiel auf Github ansehen: [src/main/java/org/bschlangaul/examen/examen\\_46115/jahr\\_2017/herbst/InsertionSort.java](https://github.com/src/main/java/org/bschlangaul/examen/examen_46115/jahr_2017/herbst/InsertionSort.java)



## Die Bschlangaul-Sammlung

### Hermine Bschlangaul and Friends

Eine freie Aufgabensammlung mit Lösungen von Studierenden für Studierende zur Vorbereitung auf die 1. Staatsexamensprüfungen des Lehramts Informatik in Bayern.



Diese Materialsammlung unterliegt den Bestimmungen der Creative Commons Namensnennung-Nicht kommerziell-Share Alike 4.0 International-Lizenz.

Hilf mit! Die Hermine schafft das nicht allein! Das ist ein Community-Projekt! Verbesserungsvorschläge, Fehlerkorrekturen, weitere Lösungen sind herzlich willkommen - egal wie - per Pull-Request oder per E-Mail an [hermine.bschlangaul@gmx.net](mailto:hermine.bschlangaul@gmx.net). Der  $\text{\LaTeX}$ -Quelltext dieses Dokuments kann unter folgender URL aufgerufen werden: