

FMuD

Termin 5

FMuD

Sortierung von Arrays und Listen

Aufgabe 17

Erstellen Sie eine Klasse „Person“ mit den Attributen „Name“ und „Alter“. Erzeugen Sie 5 Objekte und speichern Sie diese in einem Array und in einer Liste.

Sortierung

Bei der Sortierung werden Werte miteinander verglichen und geordnet. Das geschieht durch Algorithmen wie Bubblesort, Mergesort, Quicksort ... usw.

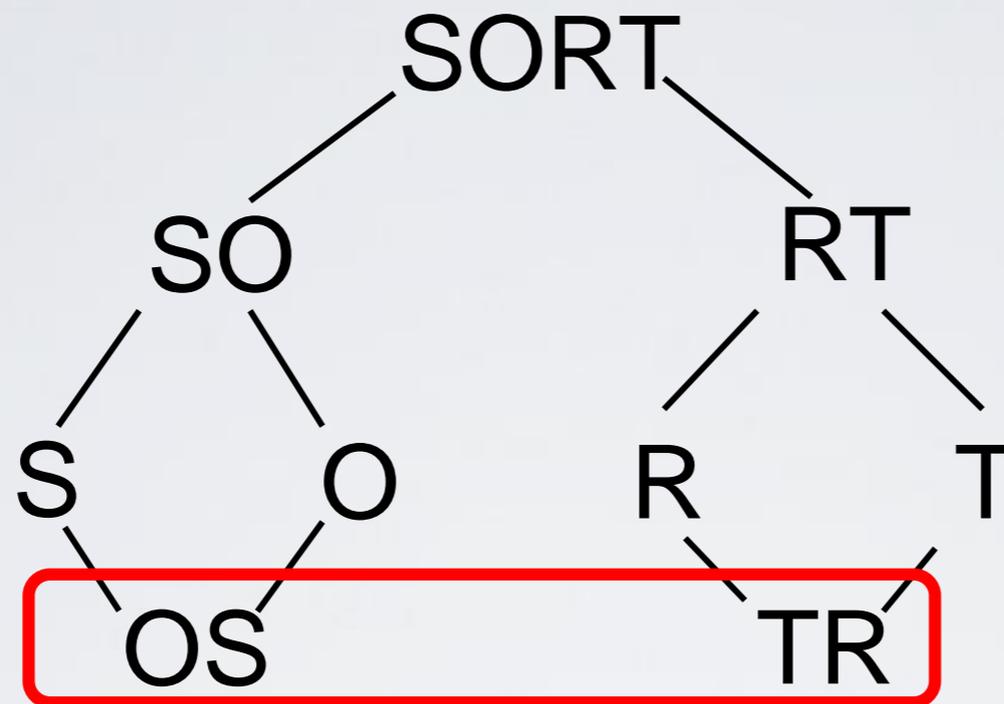
(vgl. https://en.wikipedia.org/wiki/Sorting_algorithm)

Aufgabe 18

Schreiben Sie Schritt für Schritt auf, wie Sie die Zahlenfolge ,96179' mit dem Bubblesort Algorithmus aufsteigend sortieren.

(vgl. <http://tobiasstraub.com/bubblesort>)

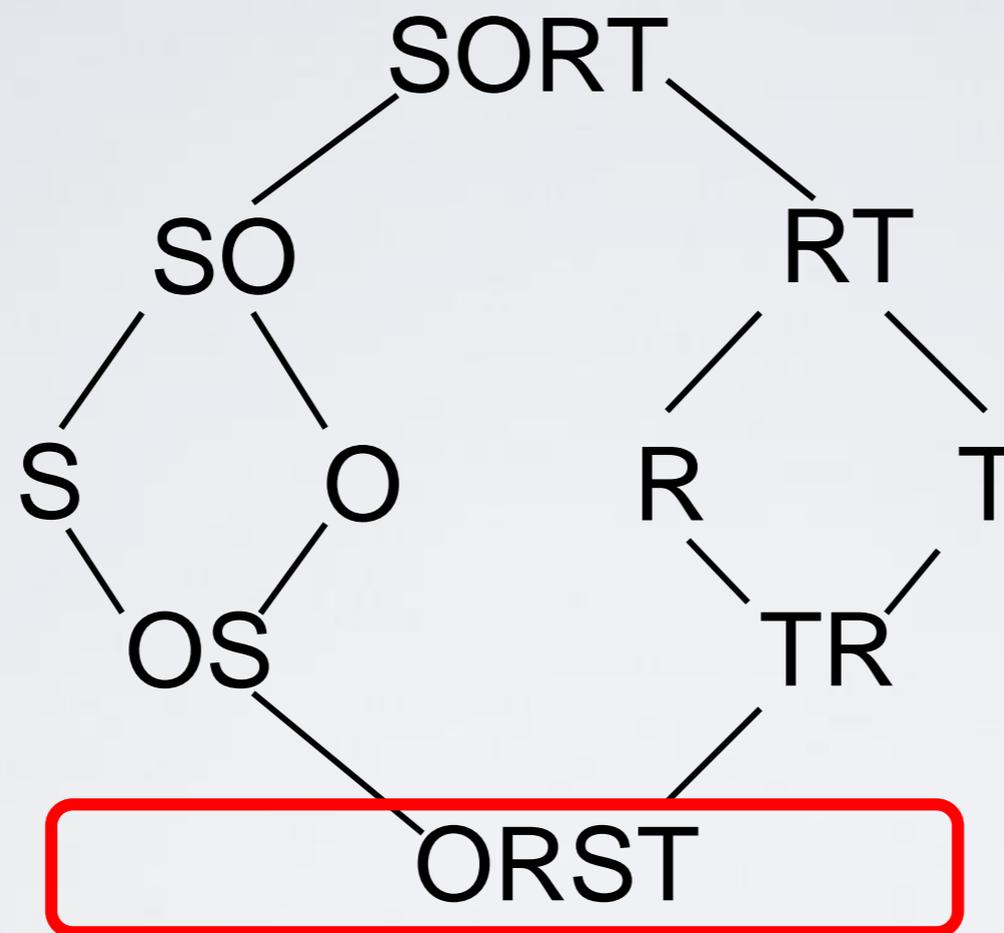
MergeSort – 1. Durchlauf



Vorgehensweise

- Teillisten sind bereits vorsortiert

MergeSort – 2. Durchlauf



Vorgehensweise

- O und R vergleichen
- S und R vergleichen
- S und T vergleichen

Sortieren in Java

Java hat einige Sortierfunktionen bereits implementiert.

Array sortieren mit: `Arrays.sort(arrayName);`

Liste sortieren mit: `Collections.sort(liste);`

Veränderte Sortierung

Das sortieren von Listen kann einfach angepasst werden. Hierbei nutzt man die Interfaces Comparable und Comparator.

Comparable wird in der zu sortierenden Klasse verwendet, man vergleicht damit das Objekt der Klasse selbst mit einem weiteren.

Comparator vergleicht zwei Objekte einer Klasse und muss nicht in der Klasse von der man das Objekt vergleichen möchte, implementiert werden.

Comparable

In der Klasse:

```
... Klassenname implements Comparable <Datentyp>{  
...  
@ Override  
public int compareTo(Datentyp objName){  
    return this.attrName.compareTo(objName.attrName);  
}
```

Sortierung:

```
Collections.sort(nameDerAList);
```

Comparator

Extra Klasse:

klassenNameComparator implements

Comparator<Datentyp>{

...

@Override

```
public int compare(Datentyp obj1, Datentyp obj2) {  
    return obj1.attrName.compareTo(obj2.attrName);  
}
```

Sortierung:

```
Collections.sort(nameDerAList,  
new klassenNameComparator());
```

Aufgabe 19

Sortieren Sie die in Aufgabe 17 erstellten Personen nach dem Namen aufsteigen und geben Sie diesen aus. Geben Sie danach die Personen nach dem Alter sortiert (Alter zuerst) aus.

Bonus Probieren Sie das ganze mit Comparator / Comparable noch einmal.