|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Eigenschaft**: | Zeitkomplexität | | Stabil  (ja / nein) | Offenkundig parallelisierbar  (ja / nein) | Spezielle Merkmale und Hinweise |
|  | Average Case | Worst Case |  |  |  |
| **Algorithmus**: |  |  |  |  |  |
| Direktes Einfügen (Insertion Sort) | O(n2) | O(n2) | ja | nein | sortierter/unsortierter Teil;  benachbartes Vertauschen |
| Direktes Auswählen (Selection Sort) | O(n2) | O(n2) | nein | nein | sortierter/unsortierter Teil;  Vertauschen über weite Entfernungen |
| Direktes Austauschen (Bubble Sort) | O(n2) | O(n2) | ja | nein | Vergleich von jedem Element mit jedem  Nur benachbartes Vertauschen von Elementen.  Option: Falls kein Vertauschen mehr stattgefunden hat, Sortieren abbrechen. |
| Shellsort | O(n1.5) Hibbard | O(n1.5) Hibbard | nein | Nein / ja | Insertion Sort mit verschiedenen Schrittweiten |
| Quicksort  (Arrays.sort()) | O(n \* log n) | O(n2) | nein | ja | Divide & Conquer;  Sortierung um Teilerelement |
| Mergesort (Collections.sort())  (Arrays.parallelSort()) | O(n \* log n) | O(n \* log n) | ja | ja | Divide & Conquer;  Reissverschlussprinzip;  Externes Sortieren |
| Heapsort | O(n \* log n) | O(n \* log n) | nein | nein | basiert auf Heap-Datenstruktur |

Quicksort:

* Worst Case: Trennelement immer an den Rand der Folge zu liegen