



Eigenschaften zur Qualitätsbestimmung von Sortieralgorithmen

Sortierverfahren unterscheiden sich hauptsächlich in den folgenden Eigenschaften:

- Zeitkomplexität
- Platzkomplexität
- Stabilität
- vergleichsbasiert/nicht-vergleichsbasiert

Zeitkomplexität von Sortieralgorithmen

Das wichtigste Kriterium bei der Auswahl eines Sortierverfahrens ist in den meisten Fällen dessen Geschwindigkeit insbesondere, wie die Geschwindigkeit funktional von der Anzahl der zu sortierenden Elemente abhängt.

Man gibt die Zeitkomplexität eines Algorithmus in der "Landau- oder [O-Notation](#)" (englisch: "Big O notation") an.

Die folgenden Klassen von Zeitkomplexitäten sind für Sortieralgorithmen:

- [\$O\(n\)\$ \("O von n"\)](#): **linearer Aufwand** – doppelt so viel Elemente benötigen doppelt so lange;
- [\$O\(n \log n\)\$](#) **quasi-linearer Aufwand** – fast so gut wie linearer Aufwand;
- [\$O\(n^2\)\$ **quadratischer Aufwand**](#) – für doppelt so viele Elemente benötigt die Sortierfunktion viermal so lange, für 10mal so viele Elemente 100mal so lang...

Bei quadratischer Komplexität stößt man relativ schnell an die Leistungsgrenzen heutiger Hardware. Quadratische Komplexität sollte also, wenn möglich, vermieden werden.

Platzkomplexität von Sortieralgorithmen

Diese gibt an, wie viel zusätzlichen Speicherplatz der Algorithmus in Abhängigkeit von der Anzahl der zu sortierenden Elemente benötigt. Damit ist nicht der Speicherplatz gemeint, der für die zu sortierenden Elemente benötigt wird, sondern der benötigter Platz für z. B. Hilfsvariablen, Schleifenzähler und temporäre Arrays.

Platzkomplexität wird mit den gleichen Klassen angegeben wie Zeitkomplexität. Hier treffen wir noch auf eine weitere Klasse:

- [\$O\(1\)\$ \(ausgesprochen "O von 1"\)](#): **konstanter Aufwand** – bei vielen Sortieralgorithmen ist der zusätzliche Speicherbedarf unabhängig von der Anzahl der zu sortierenden Elemente.

Stabile und nicht-stabile Sortierverfahren

Bei stabilen Sortierverfahren wird die relative Reihenfolge von Elementen, die den gleichen Sortierschlüssel haben, beibehalten. Bei nicht-stabilen Sortierverfahren wird dies nicht garantiert.

Was bedeutet das?

In folgendem Beispiel haben wir eine Namensliste mit zweimal den gleichen Nachnamen. Die Liste ist zunächst nach Vornamen sortiert:

Annika Weigert, Fabio Müller, Gertrud Selig, Jonathan Heydrich, Mathias Müller, Waltraud Birke

Diese Liste soll nun – ohne die Vornamen zu betrachten – nach Nachnamen sortiert werden. Wenn wir dafür ein stabiles Sortierverfahren anwenden, ist das Ergebnis immer: Waltraud Birke, Jonathan Heydrich, **Fabio Müller, Mathias Müller**, Annika Weigert, Gertrud Selig d. h. die Reihenfolge von Fabio und Mathias bleibt bei einem stabilen Sortierverfahren immer unverändert.

Vergleichsbasierte und nicht-vergleichsbasierte Sortierverfahren

Die meisten der bekannten Sortierverfahren basieren auf dem Vergleich zweier Elemente auf *kleiner*, *größer* oder *gleich*. Es existieren jedoch auch nicht-vergleichsbasierte Sortieralgorithmen.