

# Datenstrukturen

**Stack:** Es werden Objekte gespeichert die nur in umgekehrter Reihenfolge wieder gelesen werden können. Also LIFO. Zu einem solchen Speicher gehören die Operationen `push` und `pop`. In Python implementiert man einen solchen Speicher am besten als Liste.

**Queue:** Es werden Objekte gespeichert die nur in der gleichen Reihenfolge wieder gelesen werden können. Also FIFO. Zu einem solchen Speicher gehören die Operationen `enqueue` und `dequeue`.

**Heap:** Sind Priority Queues. Dabei ist jedem Element eine Priorität zugeordnet. Zu einem solchen Speicher gehören die Operationen `insert`, `remove` und `extractMin`, wobei die letzte einfach das Element mit höchster Priorität entfernt und zurückgibt. Oft wird ein Heap als Binärbaum implementiert, wobei jeder Knoten einen Schlüssel und das Element selber speichert und die *Heap-Eigenschaft* gilt. Also bei

- einem Max-Heap, die Schlüsselwerte auf jedem Weg von der Wurzel zu einem Blatt monoton fallend sind.
- einem Min-Heap, die Schlüsselwerte auf jedem Weg von der Wurzel zu einem Blatt monoton steigend sind. Also der kleinste Schlüssel immer in der Wurzel ist.

**Graph:** Ein Graph als Datentyp sollte es mindestens erlauben eine Kante und Knoten einzufügen, zu löschen und zu finden. Implementiert wird ein Graph meist als Adjazenzmatrix oder als Adjazenzliste(= `dict<int,list<int>>>`).

**Baum:** Ein Baum ist ein Zusammenhängender, kreisfreier Graph mit  $n - 1$  Kanten. Ein Gerichteter Graph ist genau dann ein Baum wenn er ungerichtet auch ein Baum wäre. Die Wurzel existiert nur in einem gerichteten Graphen und ist der Knoten von dem aus alle anderen Knoten erreichbar sind. Blätter sind Knoten mit Grad 1. Der Maximale Ausgangsgrad eines gerichteten Baums wird als Ordnung bezeichnet. Tiefe eines Knotens ist die Länge des längstens Weges von der Wurzel zu ihm.

Ein Partiiell geordneter Baum ist ein Baum, dessen Knoten markiert sind und indem für jeden Teilbaum gilt, dass die Markierung aller Knoten größer oder gleich der jeweiligen Wurzel ist. Bei Binärbäumen hat jeder Knoten höchstens zwei Kinder.

Bei einem balancierten Binärbaum unterscheiden sich außerdem die Länge der Pfade von der Wurzel zu allen Blättern höchstens um eins.

Bei Suchbäumen sind die Elemente der Baumstruktur geordnet abgelegt, sodass man schnell Elemente im Baum finden kann.

Man kann einen Baum auf drei Arten durchgehen:

- in order: Also L - W - R.
- pre order: Also W - L - R.
- post order: Also L - R - W.