

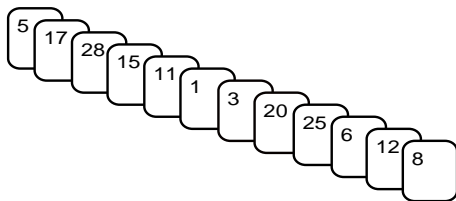
Schnelle Sortieralgorithmen

Helmut Alt
Freie Universität Berlin

20. März 2006

Aufgabe

Ein Meister gibt Dir einen Stapel nummerierter Karten



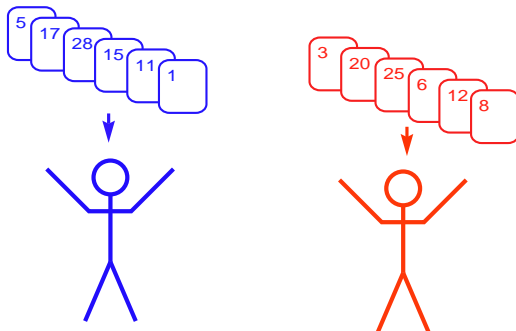
Aufgabe: Sortiere sie aufsteigend von unten nach oben

Algorithmus 1

Schritt 1: Aufteilen

Falls der Stapel nur aus einer Karte besteht, gib ihn sofort zurück, andernfalls:

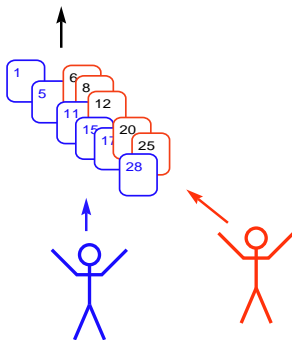
Teile den Stapel in zwei möglichst gleich große Teile und gib jeden Teil je einem Gehilfen mit der Bitte, ihn ebenfalls genau nach dem hier beschriebenen Verfahren zu sortieren.



Algorithmus 1

Schritt 2: Mischen

Warte bis dir beide Gehilfen die sortierten Teile zurückgegeben haben. Dann durchlaufe beide Stapel gleichzeitig von oben nach unten, und mische die Karten nach einer Art Reißverschlussprinzip zu einem sortierten Gesamtstapel zusammen. Gib diesen an deinen Meister zurück.



Algorithmus 2

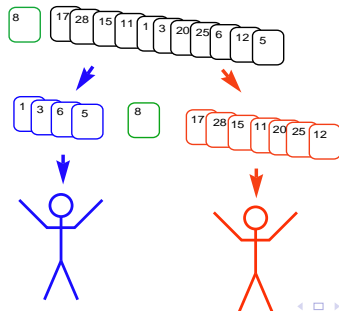
Schritt 1: Aufteilen

Falls der Stapel nur aus einer Karte besteht, gib ihn sofort zurück, andernfalls: Nimm die erste Karte aus dem Stapel. Durchlaufe die restlichen Karten und teile sie auf:

Stapel 1: alle mit Wert kleiner oder gleich dem der ersten Karte

Stapel 2: alle mit Wert größer als dem der ersten Karte.

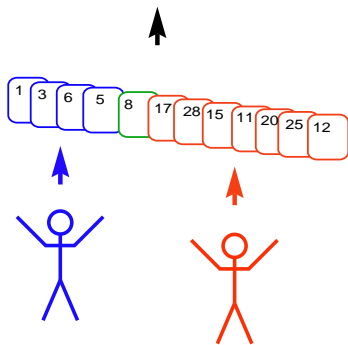
Teilstapel, wenn sie überhaupt Karten enthalten, an je einen Gehilfen geben, der sie ebenfalls genau nach dem hier beschriebenen Verfahren sortiert.



Algorithmus 2

Schritt 2: Zusammensetzen

Warte bis dir beide Gehilfen die sortierten Teile zurückgegeben haben, dann lege zuunterst den sortierten Stapel 1, darauf die anfangs gezogene Karte, darauf den sortierten Stapel 2 und gib das Ganze als sortiert zurück.



Namen der Algorithmen

Algorithmus 1 heißt

Mergesort

Erfinder unbekannt,
aber schon von **Johann von Neumann (1903 - 1957)** benutzt.

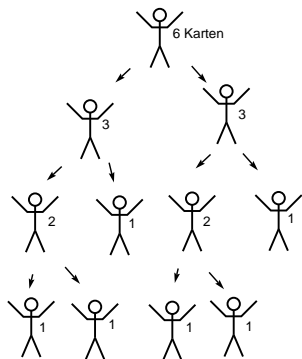
Algorithmus 2 heißt

Quicksort

Erfinder: **C.A.R. Hoare 1962**

Rekursion

Das Prinzip Gehilfen zu beauftragen, mit einem Teil der Daten den gleichen Algorithmus wieder auszuführen, heißt in der Informatik **Rekursion**. Die Gehilfen beauftragen ihrerseits wieder Untergehilfen usw. bis nur noch aus einer Karte bestehende Stapel übrigbleiben, z.B. bei Mergesort mit 6 Karten:



Realisierung bei der Programmierung: eine Prozedur ruft *sich selbst* wieder auf.

Laufzeit

Man kann beweisen: Die Anzahl der Vergleiche zwischen je zwei Karten, bei **Mergesort**, um n Karten zu sortieren, ist in etwa $n \log n$ (Logarithmus zur Basis 2).

Bei **Sortieren durch Einfügen** (siehe AdW 2) sind es dagegen $0,5n(n - 1)$ Vergleiche.

Deswegen ist Mergesort für große Eingaben schneller, z.B. bei $n = 1000$:

$1000 \log 1000 \approx 9965$ und $0,5 \cdot 1000 \cdot 999 = 499500$.

Quicksort benötigt im schlechtesten Fall (z.B. wenn die Eingabefolge bereits sortiert ist), auch $0,5n^2$, im Mittel aber etwa $1,38n \log n$ und gilt in der Praxis als der **schnellste** Sortieralgorithmus.