

Algorithmenkonstruktion

Rekursions-Arten: Ein Algorithmus kann sein

- linear rekursiv: in der Funktion kommt genau ein weiterer Aufruf zustande
- nicht linear rekursiv: es kommt zu mehrfachen Aufrufen einer Funktion in einer Funktion
- endrekursiv: der Rekursionsaufruf ist die letzte Operation in der Funktion. Das Ergebnis des Aufrufes wird also eins zu eins weiter returned. Heißt beim Erreichen der Abbruchbedingung liegt bereits das Ergebnis vor.

Man spricht dabei von direkter Rekursion, wenn eine Funktion sich selbst aufruft. Und indirekt wenn die Funktion einer andere aufruft die dann wieder die Ursprungsfunktion aufruft.

Greedy-Algorithmus: Ein Greedy Algorithmus probiert nach jedem Schritt den nächst besten Schritt zu nehmen. Somit kann es passieren, dass der Algorithmus in einem lokalen Optimum stecken bleibt.

Divide and Conquer: Teile das Problem in gleich große Teilprobleme. Löse die Teilprobleme auf die selbe Art. Wiederhole bis, die Teilprobleme klein genug sind (Herrsche) und bricht so die Rekursion ab. Vereinige die Lösungen dann zu einer Gesamtlösung. Z.B. Merge-Sort.

Backtracking: Dabei werden erreichte Teillösungen zu einer Gesamtlösung aufgebaut. Wenn klar ist, dass eine Teillösung nicht zu einer Gesamtlösung führt werden die Schritte wieder rückgängig gemacht. Also quasi systematisches Ausprobieren.