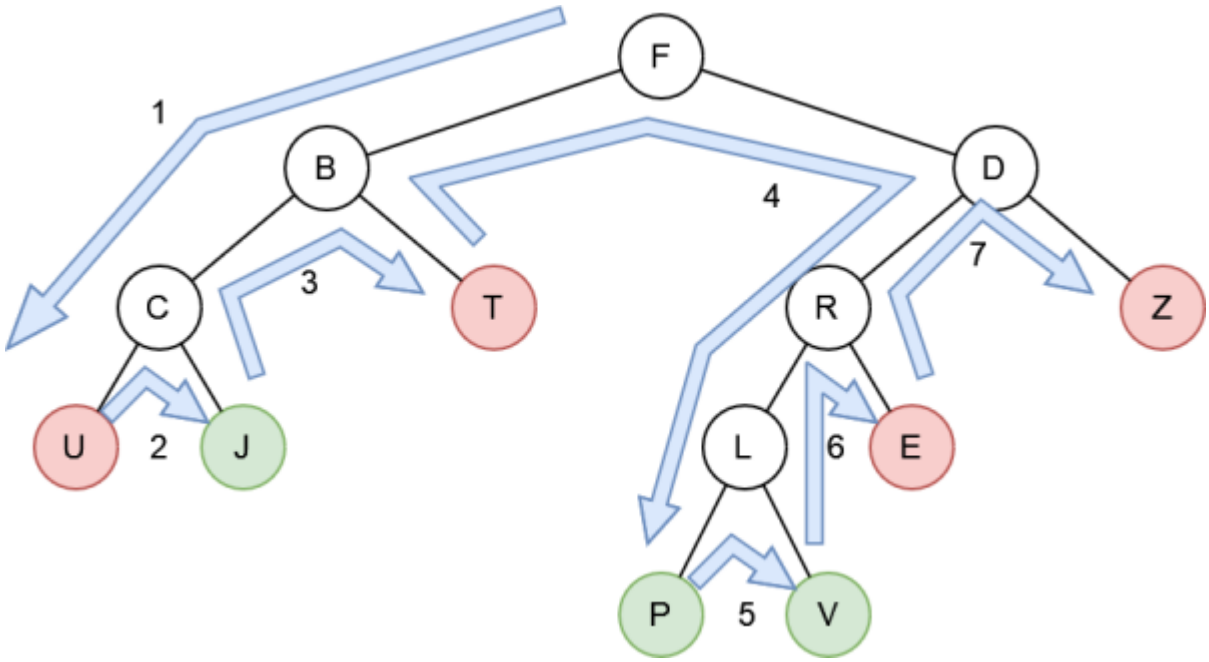
Backtracking

Das Backtracking ("Rückverfolgung") ist ein weiteres Paradebeispiel dafür, wie extrem "mächtig" Programmierung mit Rekursion sein kann. Backtracking ist eine bekannte Vorgehensweise zum Problemlösen, die besonders dafür geeignet ist, ein Problem mit sehr vielen potenziellen Kombinationsmöglichkeiten systematisch zu durchsuchen, und nicht funktionierende Kombinationen direkt auszuschließen.

Backtracking arbeitet dabei nach dem Prinzip der Tiefensuche, durchsucht also gewissermaßen verschiedene mögliche Pfade zum Ziel. Sobald ein Pfad aufgrund einer Regelverletzung nicht mehr zur Lösung führen kann, so wird der nächste benachbarte Pfad ausprobiert. Mit Backtracking können alle möglichen Lösungen ("Pfade") gefunden werden, nicht nur eine Lösung.



Am rechten Baum sieht man, wie man sich das Backtracking vorstellen kann. Man beginnt an der Wurzel und probiert zunächst immer die linken Pfade aus.
In diesem Fall gibt es beim U die erste (rot dargestellte) Regelverletzung (die ist diesem Beispiel nicht genauer erklärt sei). Man ist daher gezwungen, wieder einen Knoten zurück zu gehen und den nächsten Nachbarknoten (sofern vorhanden) zu testen. J ist eine legitime Lösung. Auch jetzt geht man wieder so weit zurück zu den Elternknoten, bis es einen weiteren Nachbarzweig gibt, den man ausprobieren kann. Am Ende hat man herausgefunden, dass J, P und V legitime Lösungen darstellen.

Die folgenden komplexen Probleme sind Beispiele, die sich mit Backtracking mit erstaunlich wenig Code schnell lösen lassen. 8-Damen-Problem

From:
https://www.info-bw.de/ -

Permanent link:
https://www.info-bw.de/faecher:informatik:oberstufe:algorithmen:rekursion:backtracking:start?rev=1719487602

Last update: **27.06.2024 11:26**