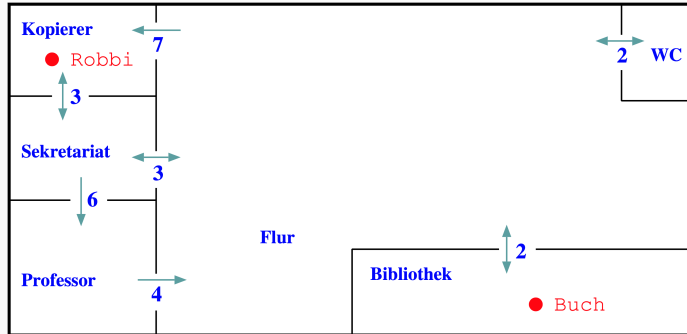


Suche mit Tiefensuche

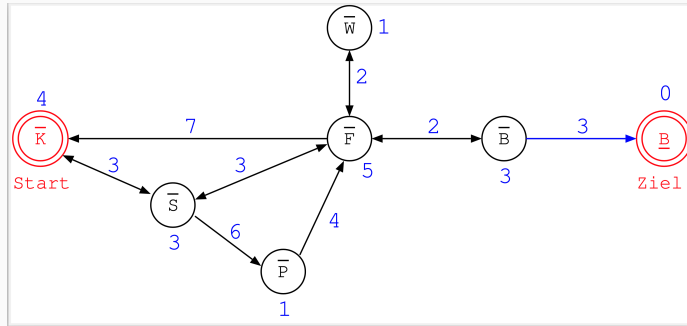
Carsten Gips (FH Bielefeld)

Unless otherwise noted, this work is licensed under CC BY-SA 4.0.

Hole das Buch



Problemgraph zum Kopiererbeispiel



Uninformierte (“blinde”) Suche:

Keine Informationen über die Kosten eines Pfades: Nur die **Pfadlänge** (Anzahl der Schritte) zählt.

Tiefensuche (*TS*, *DFS*)

1. Füge Startknoten in leere Datenstruktur (Stack, Queue, ...) ein
2. Entnehme Knoten aus der Datenstruktur:
 - Knoten ist gesuchtes Element: Abbruch, melde "*gefunden*"
 - Expandiere alle Nachfolger des Knotens und füge diese in die Datenstruktur ein
3. Falls die Datenstruktur leer ist: Abbruch, melde "*nicht gefunden*"
4. Gehe zu Schritt 2

=> Was passiert, wenn wir einen **Stack** einsetzen?

Tiefensuche (rekursive Variante)

1. Startknoten ist gesuchtes Element: Abbruch, melde "*gefunden*"
2. Für jeden Nachfolger des Startknotens:
 - Rufe Tiefensuche für aktuellen (Nachfolger-) Knoten auf
 - Ergebnis "*gefunden*": Abbruch, melde "*gefunden*"
3. Abbruch, melde "*nicht gefunden*"

- Uninformierte Suchverfahren
 - Keine weiteren Pfadkosten (nur Anzahl der Schritte)
 - Tiefensuche: Verfolge einen Pfad zuerst in die Tiefe
 - Backtracking bei Sackgassen (automatisch durch den Stack)

LICENSE



Unless otherwise noted, this work is licensed under CC BY-SA 4.0.