

Eine sprachliche Beschreibung
für zwei Funktionen zur

Verwaltung der kürzesten Wege bei Dijkstra

- Bei der Expansion des nächsten Suchschritts beim Algorithmus von Dijkstra werden alle Nachfolgewege eines aktuellen Wegs benötigt.
- Wege sind Listen aus der Länge und seinen enthaltenen Knoten (rückwärts).
- Muster eines Wegs von I über B zu X:
(60 X B I)

Die Aufgabe wird in zwei Funktionen aufgeteilt. Die erste bekommt alle Nachfolgekanten zum aktuellen Knoten von der Kantenliste.

- **alle-nachfolge-wege**
gibt zum aktuellen Weg alle Wege zurück, die sich aus den Nachfolgekanten ergeben.
- Sie benötigt
 - den aktuellen Weg: `weg`
 - und die Liste `kanten` aller Kanten des Graphen.

Die Aufgabe wird in zwei Funktionen aufgeteilt.

Die zweite gibt zu einem Weg und zu einer Kante den verlängerten Weg zurück:

- **neuer-weg**
 - bestimmt dazu die neue Gesamtlänge des Wegs
 - und fügt sie am Kopf des neuen Wegs ein,
 - der an zweiter Position zusätzlich den Zielknoten der Kante eingefügt bekommt.

neuer-weg

- erfordert daher keine Rekursion,
- sondern
 - setzt nur die Ergebnisliste für den verlängerten Weg aus den beiden Teilen zusammen

alle-nachfolge-wege arbeitet mit einer inneren Funktion, die alle Nachfolgekanten erhält.

- Beschreibung des Abbruchfalls:
 - Wenn diese Liste leer ist, werden die neuen Wege zurück gegeben.
- Ist der Zielknoten im Weg enthalten,
 - arbeitet **alle-nachfolge-wege** ohne diese Kante weiter.
- sonst ...

alle-nachfolge-wege arbeitet mit einer inneren Funktion, die alle Nachfolgekanten erhält.

- Beschreibung des Abbruchfalls:
 - ...
- Ist der Zielknoten im Weg enthalten,
 - ...
- sonst
 - wird **alle-nachfolge-wege** mit dem Rest der Nachfolger und dem von **neuer-weg** gelieferten Weg in **neue-wege** aufgerufen.