

Prim Funktionen

Entwicklung
der Funktionen
eines Programms
zum
Algorithmus von Prim

Prim Funktionen

Beginne mit einem Baum, der **allein aus einem Knoten** besteht.

Ordne alle Kanten von diesem Knoten aus nach ihren Kosten. (→ **Prioritätswarteschlange**)

Füge nun jeweils immer aus dieser jeweils die nächste **Kante mit den geringsten Kosten** ein, die **keinen Zyklus** erzeugt.

Dann füge alle von der **freiem Knoten** ausgehenden zulässigen Kanten in die Prioritätswarteschlange ein.

Brich damit ab, wenn **alle Knoten** des Graphen **zum Baum** gehören.

Formulierung ist nicht funktional!

Prim Funktionen

Funktionen zur Übersichtlichkeit des Aufrufs
[also prinzipiell unnötig]

- aktualisiere-besuchliste
 - fügt den freien Knoten der Kante der Besuchliste hinzu
- kante-hinzufuegen
 - fügt die Kante der Liste kuerzeste hinzu
- aktualisiere-priows
 - fügt die Nachfolgekanten in die Prioritätswarteschlange ein

Prim Funktionen

- vor?
 - bestimmt, ob eine Kante vor einer anderen in die Prioritätswarteschlange einzuordnen ist
- zyklus?
 - prüft, ob der freie Knoten der Kante in der Besuchliste enthalten ist
- anzahl-knoten
 - bestimmt die Anzahl der Knoten aus der Kantenliste
- laenge-netz
 - berechnet die Länge zum fertigen Baum

Prim Funktionen

- nachfolge-kanten
 - gibt zu einem Knoten alle Kanten zurück, die vom Knoten ausgehen
- fuege-einen-ein und fuege-alle-ein
 - zur Verwaltung der Prioritätswarteschlange

Standardfunktion:

- member
 - testet, ob das Element in einer Liste enthalten ist; gibt anderenfalls #f zurück