Dijkstra Typisierung

Typisierung
von
Suchverfahren

Dijkstra Typisierung

- Der Algorithmus von Dijkstra ist

  kein blindes *)

  Suchverfahren.
- Die Reihenfolge, in der Nachfolgeknoten ausgewählt werden, ist eindeutig.
  Sie ergibt sich aus der Position in der Prioritätswarteschlange.

*) auch: "ein informiertes Suchverfahren"

Dijkstra Typisierung

- Zum Vergleich:
  Tiefensuche ist ein

  blindes *)

  Suchverfahren.
- Die Reihenfolge, in der die Nachfolger bearbeitet werden, wird zwar systematisch, aber willkürlich gewählt. Sie ist prinzipiell beliebig.

*) auch: "uninformiertes Suchverfahren"

Dijkstra Typisierung

- Ebenso:
  Breitensuche ist ein

  blindes *)

  Suchverfahren.
- Die Reihenfolge, in der die Nachfolger bearbeitet werden, wird zwar systematisch, aber willkürlich gewählt. Sie ist prinzipiell beliebig.
- Unterschied zur TS: Breite zuerst

*) auch: "uninformiertes Suchverfahren"

Dijkstra Typisierung

- Der Algorithmus von Dijkstra arbeitet auf einem Graphen mit

  bewerteten Kanten

- Anderenfalls wäre eine gezielte Auswahl nicht möglich und das Verfahren ginge in die Breitensuche über.

Dijkstra Typisierung

- Zum Vergleich:
  Tiefensuche und **Breitensuche** arbeiten (in der Regel) auf einem realen Graphen oder Suchgraphen mit

  unbewerteten Kanten *)

- Sind die Kanten des Graphen dennoch bewertet, wird diese Bewertung nicht für die Reihenfolge der Bearbeitung herangezogen.

*) auch: "uniforme Kantenbewertung"

Dijkstra Typisierung

- Tiefensuche, Breitensuche und der Algorithmus von Dijkstra arbeiten

  systematisch *)

- Alle drei arbeiten allein

  mit lokalen Informationen

  Es gibt also keine globale Sicht, "wohin man muss".

*) *also keine rein zufällige Auswahl*

Dijkstra Typisierung

- Der am Beginn betrachtete Greedy Algorithmus zum Füllen eines Containers arbeitet auf einem Suchgraphen mit

  bewerteten Kanten *)

- Anderenfalls wäre eine gezielte Auswahl nicht möglich.
- Das Suchverfahren ist

  nicht vollständig

*) das "größte", "schwerste",... zuerst

Dijkstra Typisierung

- Tiefensuche, Breitensuche und der Algorithmus von Dijkstra arbeiten (zumindest prinzipiell)

  vollständig *)

*) *jeder Knoten kann erreicht werden*