

Alte und neue Versionen des generischen Suchalgorithmus

Kapitel IV – Graphen; Algorithmen

- Generischer Suchalgorithmus (alt)

Initialisierung;

while die Worklist nicht leer ist

entferne einen Knoten v aus der Worklist;

für alle Nachbarn u von v

wenn u noch nicht besucht wurde dann

markiere u als besucht;

(trage die zus. Inf. für u in die Tabelle ein);

füge u zur Worklist hinzu.

Kapitel IV – Graphen; Algorithmen

- Generischer Suchalgorithmus (neu)

Initialisierung;

while die Worklist nicht leer ist

 wähle einen Knoten v aus der Worklist;

 falls v mindestens einen unbesuchten Nachbarn hat

 wähle einen unbesuchten Nachbarn u von v ;

 markiere u als besucht;

 (trage die zus. Inf. für u in die Tabelle ein);

 trage u in die Worklist ein

 andernfalls entferne v aus der Worklist

Kapitel IV – Graphen; Algorithmen

- Erweiterter Suchalgorithmus (alt)

Initialisierung;

$n \leftarrow 1$; $n[s] \leftarrow n$; $d[s] \leftarrow 0$; für alle $v \in V \setminus \{s\}$ $n[v], d[v] \leftarrow \infty$;

while die Worklist nicht leer ist

entferne einen Knoten v aus der Worklist;

für alle Nachbarn u von v

wenn u noch nicht besucht wurde dann

markiere u als besucht;

$n \leftarrow n+1$; $n[u] \leftarrow n$; $d[u] \leftarrow d[v]+1$; $\text{pred}[u] \leftarrow v$;

füge u zur Worklist hinzu.

Kapitel IV – Graphen; Algorithmen

- Erweiterter Suchalgorithmus (neu)

Initialisierung;

$n \leftarrow 1$; $n[s] \leftarrow n$; $d[s] \leftarrow 0$; für alle $v \in V \setminus \{s\}$ $n[v], d[v] \leftarrow \infty$;

while die Worklist nicht leer ist

 wähle einen Knoten v aus der Worklist;

 falls v mindestens einen unbesuchten Nachbarn hat

 wähle einen unbesuchten Nachbarn u von v ;

 markiere u als besucht;

$n \leftarrow n+1$; $n[u] \leftarrow n$; $d[u] \leftarrow d[v]+1$; $\text{pred}[u] \leftarrow v$;

 trage u in die Worklist ein

 andernfalls entferne v aus der Worklist