



test

Max Mustermann

Fachhochschule Trier

18. Juli 2015



Table of content

eins

Komplexität

Implementierung

eins



HOCHSCHULE **TRIER**
Trier University of Applied Sciences
Informatik - Computer Science

test

Komplexität

Ursprüngliche Implementierung

Ursprüngliche Implementierung

- ▶ Einzelschritte:

Ursprüngliche Implementierung

- ▶ Einzelschritte:
 - ▶ Initialisieren der Arrays je $O(m)$

Ursprüngliche Implementierung

- ▶ Einzelschritte:
 - ▶ Initialisieren der Arrays je $O(m)$
 - ▶ Abarbeiten der grauen Knoten $O(m)$

Ursprüngliche Implementierung

- ▶ Einzelschritte:
 - ▶ Initialisieren der Arrays je $O(m)$
 - ▶ Abarbeiten der grauen Knoten $O(m)$
 - ▶ Bestimmen des Minimums $O(m)$

Ursprüngliche Implementierung

- ▶ Einzelschritte:
 - ▶ Initialisieren der Arrays je $O(m)$
 - ▶ Abarbeiten der grauen Knoten $O(m)$
 - ▶ Bestimmen des Minimums $O(m)$
 - ▶ Aktualisieren der Nachfolger $O(\deg(v)) \rightarrow O(k)$

Ursprüngliche Implementierung

- ▶ Einzelschritte:
 - ▶ Initialisieren der Arrays je $O(m)$
 - ▶ Abarbeiten der grauen Knoten $O(m)$
 - ▶ Bestimmen des Minimums $O(m)$
 - ▶ Aktualisieren der Nachfolger $O(\deg(v)) \rightarrow O(k)$
- ▶ insgesamt Komplexität $O(m^2)$

Implementierung mit Heap

- ▶ Vorteile:

Implementierung mit Heap

- ▶ Vorteile:
 - ▶ Heapoperationen in $O(\log m)$

Implementierung mit Heap

- ▶ Vorteile:
 - ▶ Heapoperationen in $O(\log m)$
 - ▶ Bestimmen des Minimums in $O(\log m)$

Implementierung mit Heap

- ▶ Vorteile:
 - ▶ Heapoperationen in $O(\log m)$
 - ▶ Bestimmen des Minimums in $O(\log m)$
- ▶ insgesamt Komplexität $O(k * \log m)$

Eigenschaften

Implementierung

Eigenschaften

- ▶ Programmiersprache: Python

Eigenschaften

- ▶ Programmiersprache: Python
- ▶ Umsetzung mit Heap (Priority Queue)

Eigenschaften

- ▶ Programmiersprache: Python
- ▶ Umsetzung mit Heap (Priority Queue)
- ▶ keine Speicherung der Farbstufen wie bei Dijkstra

Eigenschaften

- ▶ Programmiersprache: Python
- ▶ Umsetzung mit Heap (Priority Queue)
- ▶ keine Speicherung der Farbstufen wie bei Dijkstra
 - ▶ kürzerer und übersichtlicherer Code



Implementierung

Kompletter Code

Implementierung

Eingabe

- ▶ leer

Algorithmus

- ▶ Initialisierung

Algorithmus

- ▶ Initialisierung
- ▶ Erweitern

Algorithmus

- ▶ Initialisierung
- ▶ Erweitern
- ▶ Aktualisieren

Rekursives Bestimmen des Pfades

- ▶ leer

Aufruf

- ▶ leer