

Programmieraufgabe - Logik

Rahmenbedingungen

- Schreiben Sie produktionsreifen Code
- Nutzen Sie .NET oder Javascript
- Nutzen Sie keine externen Bibliotheken zur Problemlösung, für untergeordnete Themen wie Unit-Testing, Mocking, Logging o.ä dürfen sie gern auf Drittkomponenten zurückgreifen
- Übermitteln Sie den Code und alles was zur Kompilierung notwendig ist. Übermitteln Sie keine Binaries.
- Falls Sie es für notwendig erachten, übermitteln Sie bitte Doku zu Implementierung, Ausführung und Test

Ausgangssituation

In einem Streckennetz eines lokalen Bahnbetreibers sind alle Strecken Einwegstrecken, d.h. dass die Existenz einer Verbindung von Oberdorf nach Unterstadt nicht bedeutet, dass es auch eine Verbindung von Unterstadt nach Oberdorf gibt. Sollte es beide Verbindungen geben, so bedeutet dies nicht, dass beide Verbindungen die gleiche Entfernung und gleiche Kosten besitzen.

In der folgenden Aufgabe sollen Algorithmen zur Ermittlung folgender Sachverhalte erstellt werden:

- Ermittlung der Entfernung entlang einer Route
- Ermittlung der Anzahl an Routen zwischen zwei Orten
- Ermittlung der kürzesten Route zwischen zwei Orten

Eingabedaten

Als Eingabe wird ein gerichteter Graph genutzt. Die Knoten des Graphen repräsentieren die Orte, die Kanten repräsentieren Verbindungen zwischen den Orten. Jede Kante ist gewichtet, die Wichtung repräsentiert die Entfernung zwischen den Orten. Eine Route zwischen zwei Orten existiert stets nur einmal. Der Ausgangs- und Zielort einer Verbindung sind stets verschieden.

Als Test-Eingabe wird folgender Graph verwendet [Start][Ende][Länge]:

- AB5, BC4, CD8, DC8, DE6, AD5, CE2, EB3, AE7

Ausgabedaten

Wenn keine für die folgenden Anfragen gefunden wurde, soll "No route exists" ausgegeben werden.
Die folgenden Fragen sollen beantwortet werden:

1. Länge der Route A-B-C.
2. Länge der Route A-D.
3. Länge der Route A-D-C.
4. Länge der Route A-E-B-C-D.
5. Länge der Route A-E-D.
6. Anzahl der Routen, die bei C anfangen und auch bei C wieder enden und die nicht mehr als 3 Stopps haben.
7. Anzahl der Routen die bei A anfangen und bei C enden mit exakt 4 Stopps.
8. Länge der kürzesten Route von A nach C.
9. Länge der kürzesten Route von B nach B.
10. Anzahl von Routen von C nach C mit einer Länge < 30 .

Erwartete Ergebnisse

#1: 9
#2: 5
#3: 13
#4: 22
#5: NO SUCH ROUTE
#6: 2
#7: 3
#8: 9
#9: 9
#10: 7