Bellman Ford Algorithmus

Anders als beim Algorithmus von Dijkstra können die Gewichte der Kanten auch negativ sein. Zyklen negativer Länge, die vom Startknoten aus erreichbar sind, müssen jedoch ausgeschlossen werden können, da diese sonst beliebig oft durchlaufen und so immer kürzere Wege konstruiert werden könnten. Der Algorithmus vermag jedoch Zyklen negativer Länge zu erkennen.

== Beispiel ==
Betrachten Sie das Beispiel Bild:
easy-coding.de/Attachment/772/…0aa2fc3c177519f36290b1a7f

Suche den Weg von "A nach E"
easy-coding.de/Attachment/773/…0aa2fc3c177519f36290b1a7f

Ergebnis: Der kürzeste Weg ist [AB, BD, CE] mit dem Gewicht 14

== Algorithmus ==
G bezeichnet den gewichteten Graphen mit V als Knotenmenge und E als Kantenmenge. Gewicht ist die Gewichtsfunktion des Graphen und bestimmt die Distanz von zwei Knoten, die durch eine Kante verbunden werden. s ist der Startknoten, von dem ausgehend die kürzesten Wege zu allen anderen Knoten berechnet werden, und n ist die Anzahl der Knoten in V.

Wenn die Ausführung des Algorithmus' endet, kann der Ausgabe entnommen werden, ob G einen Zyklus negativer Länge besitzt. Falls dies nicht der Fall ist, enthält Distanz die Abstände aller Knoten zu s. Um von einem Knoten auf dem kürzesten Weg zum Startknoten s zu gelangen muss man also so lange den Knoten besuchen, der durch Vorgänger(v) gegeben ist, bis man s erreicht hat. Genauer gesagt wird durch Vorgänger ein Spannbaum definiert, der die von s aus ausgehenden minimalen Wege in Form eines In-Trees speichert.

== Weblinks ==

- links.math.rpi.edu/applets/appindex/graphtheory.html|Bellman Ford Java Applet