Andreas Mühlmann

andi.muehlmann@arcor.de, 18.11.2021

Programmiersprache: Python

Dokumentation Vollgeladen

Aufgabe 2

Ansatz:

Meine Idee war es einen Baum aufzuspannen und diesen dann durchzugehen, indem man dieses Problem in viele Kleinere aufteilt. Jeder Knoten soll seine Kanten durchgehen und diese sollen wieder ihre Kanten durchgehen. Das ist rekursiv und wird Tiefensuche genannt, weil das Programm erst zum Ende jedes Weges geht. Der Weg mit der besten Bewertung wird gespeichert. Das funktionierte auch ganz gut, war aber zu langsam.

Dann fielen mir noch drei Optimierungen ein.

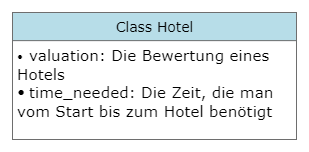
1. Man kann einen Weg abbrechen, wenn dieser eine niedrigere Bewertung hat als ein Vorheriger.
2. Man kann einen Weg abbrechen, wenn das Endziel, mit den restlichen Übernachtungen, gar nicht mehr zu erreichen ist.
3. Wenn man erst die Hotels mit den besten Bewertungen probiert, dann gibt es im Durchschnitt deutliche Zeitersparnisse, weil man dann oft viele Wege, die folgen würden abbrechen kann.

Diese Optimierungen haben gereicht um den Algorithmus zu genug beschleunigen.

|  |  |
| --- | --- |
| Ende | Übernachtungen |
| 800 | 2 |

*Werte aus hotels1 von 12; 4.3 – 590; 0.8, erfundenes Ende und erfundene Anzahl Übernachtungen*

Beschreibung:



Gemacht auf: <https://www.smartdraw.com/>

Funktion main:

* Die einzelnen Tests werden nacheinander durchgegangen.
* Die Hotels, die Anzahl der Hotels und die benötigte Zeit werden von der Website eingelesen (Funktion get\_hotels\_from\_website).
* Der Weg mit der besten niedrigsten Bewertung und die niedrigste Bewertung dieses Weges werden gefunden (Funktion path\_finder).
* Der Weg (Funktion write\_path\_to\_file) und die beste niedrigste Bewertung werden in die geöffnete Datei geschrieben.

Funktion path\_finder:

* Erst werden die zu erreichenden Hotels für die aktuell gefahrene Distanz gegeben (Funktion give\_reachable) (Wenn man an keinem Hotel ist, ist die Distanz null).
* Wenn die benötigte Zeit erreicht wird, wird die beste niedrigste Bewertung und der Weg abgespeichert, wenn die beste niedrigste Bewertung besser ist als alle Vorherigen.
* Wenn das Ziel nicht mehr zu erreichen ist, oder schon zu viele Übernachtungen benötigt sind, wird dieser Weg abgebrochen.
* Ein Weg wird schon dann abgebrochen, wenn es schon einen Weg gibt, der das Ziel erreicht hat und eine bessere niedrigste Bewertung hat, als der aktuelle Weg.
* path\_finder wird für die Kanten aufgerufen, die das Ziel theoretisch noch erreichen könnten.
* Am Schluss werden die beste niedrigste Bewertung und der Weg, der diese erreicht hat, zurückgegeben. Wenn das Ziel nicht zu erreichen war, wird ein leerer Weg zurückgegeben.

Funktion give\_reachable:

* Gibt „True“ zurück, wenn das Ende in einem Tag zu erreichen ist.
* Geht durch die Hotels und speichert die zu Erreichenden.
* Die Hotels werden dann noch nach ihrer Bewertung absteigend sortiert, damit die Bestbewerteten zuerst angeschaut werden.
* Wenn keine Hotels gespeichert werden, wird eine leere Liste zurückgegeben.

Funktion write\_path\_to\_file:

* Die einzelnen Hotels werden durchgegangen.
* Die beste niedrigste Bewertung und die Zeit, die man jeweils vom Start zum Hotel braucht, werden in die Datei geschrieben.

**Lösungen:**

---------------------------------------------------------------------------------------------------------------------------

Test 1

The path with the best lowest valuation:

Hotel at time 347min, valuation 2.7

Hotel at time 687min, valuation 4.4

Hotel at time 1007min, valuation 2.8

Hotel at time 1360min, valuation 2.8

It has the best lowest valuation of 2.7.

---------------------------------------------------------------------------------------------------------------------------

Test 2

The path with the best lowest valuation:

Hotel at time 341min, valuation 2.3

Hotel at time 700min, valuation 3.0

Hotel at time 1053min, valuation 4.8

Hotel at time 1367min, valuation 4.5

It has the best lowest valuation of 2.3.

---------------------------------------------------------------------------------------------------------------------------

Test 3

The path with the best lowest valuation:

Hotel at time 316min, valuation 4.8

Hotel at time 670min, valuation 4.2

Hotel at time 1027min, valuation 4.3

Hotel at time 1323min, valuation 4.2

It has the best lowest valuation of 4.2.

---------------------------------------------------------------------------------------------------------------------------

Test 4

The path with the best lowest valuation:

Hotel at time 340min, valuation 4.6

Hotel at time 676min, valuation 4.6

Hotel at time 1032min, valuation 4.9

Hotel at time 1377min, valuation 4.4

It has the best lowest valuation of 4.4.

---------------------------------------------------------------------------------------------------------------------------

Test 5

The path with the best lowest valuation:

Hotel at time 280min, valuation 5.0

Hotel at time 636min, valuation 5.0

Hotel at time 987min, valuation 5.0

Hotel at time 1322min, valuation 4.9

It has the best lowest valuation of 4.9.