

Määrittelydokumentti

Käytettävät algoritmit ja tietorakenteet

Brute force kauppamatkustajan ongelman ratkaisija-algoritmi. (Antaa tarkan vastauksen)

Held–Karp kauppamatkustajan ongelman ratkaisija-algoritmi. (Antaa tarkan vastauksen)

Ja jotakin lähes tarkan vastauksen antavaa algoritmia.

Algoritmivalinnat saattavat vielä muuttua, riippuen erityisesti siitä että päädyinkö tulkitsemaan etäisyydet kartalla autolle pisteiden välillä symmetrisesti($A \rightarrow B == B \rightarrow A$) vai epäsymmetrisesti($A \rightarrow B \neq B \rightarrow A$). [1]

Ratkaistava ongelma ja miksi ylläolevat algoritmit

Ongelmana selvittää nopein reitti tavarantoimittajalle jonka täytyy vierailla määrätyissä osoitteissa ja sitten palata takaisin lähtöpaikkaansa.

Valittu algoritmi jolla saa tarkan vastauksen jotta voidaan mitata kuinka paljon parempi tämä vastaus on kuin lähes tarkan vastauksen antava algoritmi. Vertaillaan toisaalta kuinka paljon nopeammin lähes tarkan vastauksen algoritmi toimii.

Ohjelman saamat syötteet ja näiden syötteiden käyttö

Käyttäjä syöttää ohjelmalle lähtöosoitteen sekä kaikki osoitteet joissa täytyy vierailla.

Tämän jälkeen ajoajat jokaisesta osoitteesta jokaiseen toiseen osoitteeseen selvitetään Google Maps Distance Matrix API:n avulla[2].

Kun nämä etäisyydet on saatu API:lta niin lasketaan valituilla algoritmeilla ajoajaltaan pienin reitti joka käy kaikissa osoitteissa palaten takaisin lähtöosoitteeseen.

Aikavaativuudet

Brute forcella $O(n!)$ [3]

Held-Karpilla $O(n^2 2^n)$ [3]

Lähteet

[1] https://en.wikipedia.org/wiki/Held–Karp_algorithm

[2] <https://developers.google.com/maps/documentation/distance-matrix/>

[3] https://en.wikipedia.org/wiki/Travelling_salesman_problem