

Määrittely

Ratkaistava ongelma

Ohjelma laskee lyhyimmän reitin tähdeltä A tähdelle B, kun tähtien väliset yhteydet ovat pituudeltaan/painoltaan vakiot. Yhteydet muodostetaan etäisyyden

perusteella: tähdet, joiden välimatka on korkeintaan yhtä paljon, kuin annettu kynnysetäisyys, ovat yhteydessä toisiinsa.

Käytettävät algoritmit ja tietorakenteet

Oletusarvoisesti toteutetaan Breadth-First Search (BFS), sekä toivottavasti ajan ja energian salliessa myös A*. BFS:n toteuttamiseen tarvitaan jono (queue), lista (list), sekä assosiatiivinen lista (map). Nämä on toteutettu linkitettyllä listalla (linked list), dynaamisella taulukolla (array list), sekä hajautustaululla (hash map).

Oletettu aika- ja tilavaativuus

BFS:n aikavaativuus riippuu pitkälti yhteyksien määrästä - mitä enemmän verkossa on yhteyksiä, sitä huonommin algoritmi suoriutuu. Oletettava pahimman

tapauksen aikavaativuus on luokkaa $O(n^2)$.

Tilaa BFS vaatii tutkimista odottavien ja jo tutkittujen solmujen varastointiin, sekä kuljetun reitin muistamiseen: suuruusluokka on arviolta $O(n)$.

Syötteet

Ohjelma ottaa syötteenään JSON-formaatissa annetun listan tähdistä koordinaatteineen, lähtötähden nimen, sekä kohdetähden nimen. Esimerkiksi:

```
[ {"planets": [], "location": {"x": 0, "y": 0, "z": 0}, "name": "Sol"},  
{"planets": [], "location": {"x": -304, "y": 292, "z": -14}, "name": "Proxima Centauri"},  
{"planets": [], "location": {"x": -307, "y": 315, "z": -5}, "name": "Alpha Centauri"},  
{"planets": [], "location": {"x": 297, "y": 494, "z": 145}, "name": "Barnard's Star"}, ]
```

Koordinaattien yksikkö on 1/100 valovuotta, Auringon sijainnin toimiessa origona. Tähtien planeettalistoja ei käytetä (liittyvät toteuttamatta jääneeseen toiminnallisuuteen).

Tuloste

Mikäli annetulta lähtötähdeltä on reitti kohdetähdelle, ohjelma tulostaa reitin tähdet yksi per rivi, lähtötähdestä kohdetähteen. BFS-toteutuksella palautettu reitti on lyhyin mahdollinen, ja useamman samanpituisen reitin tapauksessa tulostetaan ensin löydetty reitti.

Lähteet

- http://en.wikipedia.org/wiki/Breadth-first_search
- http://en.wikipedia.org/wiki/Dijkstra%27s_algorithm
- http://en.wikipedia.org/wiki/A*_searchalgorithm
- [http://en.wikipedia.org/wiki/Queue\(*abstractdata_type*\)](http://en.wikipedia.org/wiki/Queue_(abstractdata_type))
- http://en.wikipedia.org/wiki/Linked_list
- http://en.wikipedia.org/wiki/Dynamic_array
- http://en.wikipedia.org/wiki/Hash_table
- http://en.wikipedia.org/wiki/Euclidean_distance
- <http://docs.oracle.com/javase/8/docs/api/index.html>