

Tietorakenteiden harjoitustyö
Tuomo Aalto (013862293)

Toteutusdokumentti

Sovelluksessa on 4 luokkaa ja yksi rajapinta:

- | | |
|--------------------------------|---|
| - DijkstraStarVertailuApp.java | pääohjelma |
| - Grid.java | Luokka, joka kuvaa läpikäytävää verkkoa |
| - Node.java | Luokka, joka kuvaa verkon solmuja |
| - NodeHeapImpl.java | Luokka, joka kuvaa minimikekoa, johon tallennetaan Node -tyyppisiä olioita. |
| - NodeHeapI.java | Minimikeon rajapinta. |

Luokkien tarkempi dokumentaatio Javadoc:issa.

Suorituskyky:

Taulukossa esitetty 20 ajokerran tulokset 1000x1000 satunnaisen verkon läpikäynnistä kummallakin algoritmilla sekä A*:n kohdalla kummallakin heuristiikalla. (Jokaisella rivillä siis läpikäyty sama verkko)

A* (Euklidininen)		A* (Manhattan)		Dijkstra	
Reitin pituus	Aika (ms)	Reitin pituus	Aika (ms)	Reitin pituus	Aika (ms)
354,22540	255	369,96551	97	354,22540	8179
1 033,27330	5788	1 057,47431	177	1 039,61645	33857
827,24978	2315	852,93102	1646	827,24978	22666
224,49747	35	234,43860	17	224,49747	1157
833,26198	1436	870,88434	620	838,23254	15095
310,52186	114	328,11984	21	313,10765	1064
596,45794	1060	631,61227	164	600,55844	9106
320,17872	131	331,87720	17	321,00714	5459
324,04372	97	348,58788	6	335,65895	691
557,68838	759	592,01429	191	562,61732	13400
796,49957	2786	839,63665	579	800,15642	17226
856,16147	831	904,51176	686	857,81833	8623
280,86501	85	290,90664	13	282,86501	3451
620,41631	815	659,02648	236	628,90159	6125
372,93102	189	399,07316	22	375,51681	6221
336,03658	136	357,87720	35	340,52186	6955
740,70981	1178	749,05296	41	740,95245	12238
267,27417	39	284,64675	4	270,78889	1287
1 058,22958	7921	1 129,18290	1201	1 066,71486	30957

478,70058	405	503,57063	43	479,52900	6552
-----------	-----	-----------	----	-----------	------

Tulosten valossa näyttäisi siltä, että A* on Dijkstraa selkeästi nopeampi, heuristiikasta riippuen n. 20 (Euklidinen) – 130 (Manhattan) kertaisesti. Ongelmia tulee kuitenkin reitin pituuden kanssa; Dijkstran ja A*:n Euklidisen heuristiikan reittien pituudet ovat hyvin lähellä toisiaan, mutta Manhattan heuristiikalla reittien pituudet ovat selvästi pidempiä. On tietenkin mahdollista että algoritmin toteutuksessa on virhe, joka aiheuttaa tämän eron, mutta testauksen vähäisyyden takia tästä ei ole taetta.

Puutteet:

Testitapaukset ovat valitettavan puutteelliset. Lisäksi sovellusta olisi voinut refaktoroida vielä huomattavasti pidemmälle, jolloin pääohjelman laajuus olisi varmasti pienentynyt.

Lähteet:

Cormen

http://en.wikipedia.org/wiki/A*

http://en.wikipedia.org/wiki/Dijkstra%27s_algorithm

<http://memoization.com/2008/11/30/a-star-algorithm-in-java/>

http://www.improvedoutcomes.com/docs/WebSiteDocs/Clustering/Clustering_

[Parameters/Manhattan_Distance_Metric.htm](http://www.improvedoutcomes.com/docs/WebSiteDocs/Clustering/Clustering_Parameters/Manhattan_Distance_Metric.htm)

<http://theory.stanford.edu/~amitp/GameProgramming/Heuristics.html>

<http://www.policyalmanac.org/games/aStarTutorial.htm>