

Sisällys

1 Johdanto.....	1
2 Testaus tavat	1
3 Testien kuvaukset	1
3.1 Yksikkö testien kuvakset.....	1
3.2 Ohjelman käyttö testit.....	2
4 Testien tulokset	3
4.1 Yksikkö testit.....	3
4.2 Ohjelman käyttötestin tulokset.....	3
Suoritus kyky mittauksia	5
5 Loppuyhteenveto.....	6

1 Johdanto

Tässä testausdokumentaatioissa on kuvattu Helsingin yliopiston tietojenkäsittelytieteen laitoksen Tietorakenteet ja algoritmit (alkukesä) harjoitustyön lopullisen työn testaamista. Dokumentissa kuvataan suoritettut testit ja niiden tulokset.

2 Testaus tavat

Ohjelman testaus suoritetaan yksikkötestauksen ja käytännön testauksen avulla. Yksikkötestaus suoritetaan NetBeansin ja JUnit-testiluokkien avulla. Käytännön testaus suoritetaan käyttämällä ohjelmaa kaikilla sen mahdollistamilla tavoilla. Ohjelman käyttötestit suoritetaan kahdella eri laitteistolla.

3 Testien kuvaukset

3.1 Yksikkö testien kuvakset.

Ohjelmassa on yksikkötestejä yhteensä 108 kappaletta. Dijkstra luokalle testejä on 21, Dijkstra8 luokalle testejä on 21, BellmanFord luokalle testejä on 21, Astar luokalle testejä on 21, Keko luokalle testejä on 15, EtsiReitti luokalle testejä on 8 ja Sijainti luokalle testejä on 1,

Luokissa Dijkstra, Dijkstra8, BellmanFord ja Astar testataan luokkien toimintaa mahdollisimman laajasti. Testit testaava luokkien toimintaa satunnaisilla syötteillä, missä reitti on sattumanvarainen. Testit testaavat myös kaikkiin ilman suuntiin kulkevat ennalta määrätty reitit. Jokaisen luokan metodin toiminta myös varmistetaan testeissä.

Keko luokassa testataan keon toimintaa laajasti. Luokkaa testataan seuraavilla testeillä: tyhjään kekoon lisätään kahdeksan alkioita. Tällöin nähdään, toimiiko keko oikein (pienillä syötteillä). Toisessa testissä ensin lisätään paljon alkioita ja poistetaan paljon alkioita. Testillä saadaan selville, toimiiko keko oikein myös suurilla syötteillä. Seuraavaksi testataan, onko keon huipulla oleva arvo oikea, jos lisätään kekoon satunnainen määrä satunnaisia alkioita. Yksi testi testaa tulostuuko keko oikein, jos kekoon yksinkertaisesti vain lisätään alkioita. Testeissä testataan myös tulostuuko keko oikein, jos keossa ei ole lainkaan alkioita.

EtsiReitti luokan testit testaavat luokan toimintaa laajasti. Testeissä testataan tulostuuko reitti oikein, kun käytetään luokkia: Dijkstra, Dijkstra8, BellmanFord ja Astar. Myös testataan haetaanko kuvan väripisteet oikein. Jokainen luokan metodin toiminta myös varmistetaan testeissä.

Luokka Sijainti on hyvin pieni luokka. Sen johdosta testejäkin on vain yksi. Se testaa meneekö parametreina annetut tiedot oikein muuttujiin.

Yksikkötesteissä ei testata luokkaa EtsiReittiUI, koska kyseinen luokka toteuttaa ohjelman käyttöliittymän. Ohjauksessa saamieni ohjeiden mukaan käyttöliittymää ei tarvitse yksikkö testata.

3.2 Ohjelman käyttö testit

Ohjelman toimintaa käytännössä testataan seuraavin tavoin. Ohjelma käynnistyy oikein, kuvan valinta toimii, eri ratkaisu tavat toimivat oikein, kuvien lisäys ja poisto toimii halutulla tavalla. Kaikki testit on lueteltu tarkemmin kohdassa 4.2 ohjelman käyttötestin tulokset.

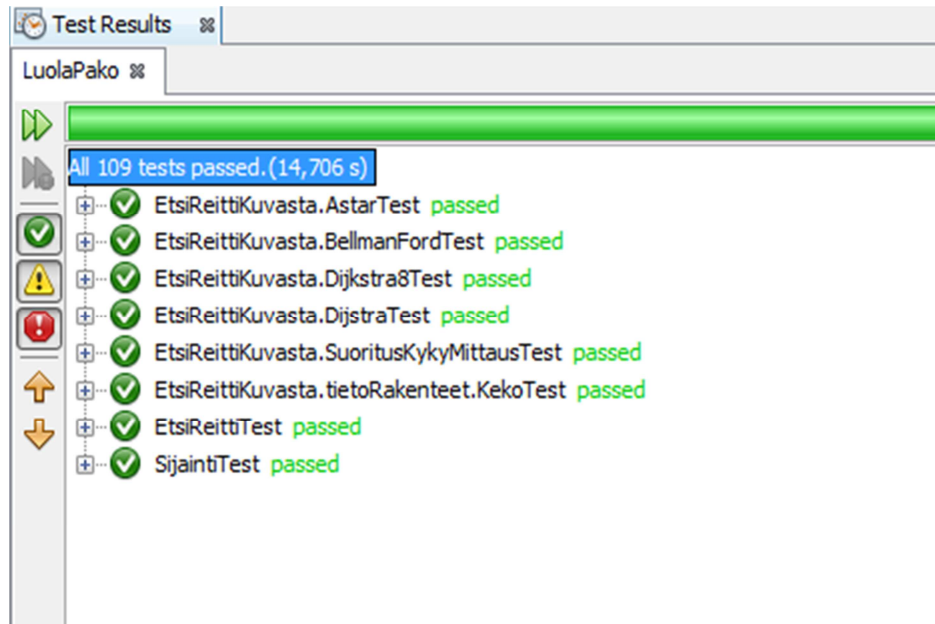
Testikokoonpano

Laitteisto1: Suoritin Intel Core i7 3612QM muistia 12 Gt, käyttöjärjestelmä Windows 7 Home Premium Edition 64-bit Service Pack 1, java versio 1.7.0_07.

4 Testien tulokset

4.1 Yksikkö testit

Yksikkö testit menivät kaikki läpi. Ohessa kuvakaappaus testi tuloksista.



4.2 Ohjelman käyttötestin tulokset

Testi kokoonpano 1 ja testi kuvana 450x450 täysin valkoinen kuva. Bellman Fordin testissä kuvan koko 100x100 pistettä täysin valkoinen.

Mitä testataan	Ohjelman oletettu toiminta	Toimiiko ohjelma
Käynnistyykö ohjelma oikein	Ohjelma aukeaa oletetulla tavalla	Toimii
Valitaan eri kuvia	Kuvat vaihtuvat	Toimii
Painetaan lisää uusi kuva painiketta	Ohjelma avaa lisäys valikon	Toimii
Lisätään uusi kuva	Kuva lisäytyy kuvalistaan	Toimii
Lisätään väärän tyyppinen kuva	Ohjelma ilmoittaa kuvan olevan väärää tyyppiä	Toimii
Painetaan poista kuva painiketta	Kuva poistuu kuvalistasta.	Toimii
Painetaan ratkaise painiketta ilman valintaa	Etäisyys on 0 aika jotain.	Toimii
Valitaan kuvasta alkupiste	Alkupisteen arvo muuttuu valitun kohdan koordinaateiksi.	Toimii
Valitaan kuvasta loppupiste	Loppupisteen arvo muuttuu valitun kohdan koordinaateiksi.	Toimii
Painetaan ratkaise painiketta, kun alku- ja loppupisteet on valittu. Valittu Dijkstra	Ratkaisu kuvaan piirtyy lyhin reitti	Toimii
Valittu Bellman Ford ja kuvan koko yli 100X100	Ohjelma ilmoittaa kuva liian isoksi ja ratkaisee kuvan käyttäen Dijkstra	Toimii
Valittu Bellman Ford ja kuvan koko alle 100X100	Ratkaisu kuvaan piirtyy lyhin reitti	Toimii

Valittu Dijkstra8	Ratkaisu kuvaan piirtyy lyhin reitti	Toimii
Valittu Astar	Ratkaisu kuvaan piirtyy lyhin reitti	Toimii
Valittu Dijkstra ja näytä kaikki pisteet	Ratkaisu kuvaan piirtyy lyhin reitti ja kaikki pisteet missä käyty.	Toimii
Valittu Astar ja näytä kaikki pisteet	Ratkaisu kuvaan piirtyy lyhin reitti ja kaikki pisteet missä käyty. Pitäisi olla vähemmän pisteitä kuin edellisessä.	Toimii
Valittu Dijkstra x=22 y=12 -> x=425 y=428	Piirtää ratkaisun oikealta ylhäältä vasempaan alakulmaan	Toimii
Valittu Dijkstra8 x=22 y=12 -> x=425 y=428	Piirtää ratkaisun oikealta ylhäältä vasempaan alakulmaan	Toimii
Valittu Astar x=22 y=12 -> x=425 y=428	Piirtää ratkaisun oikealta ylhäältä vasempaan alakulmaan	Toimii
Valittu Dijkstra x=186 y=17 -> x=184 y=432	Piirtää ratkaisun ylhäältä alas	Toimii
Valittu Dijkstra8 x=186 y=17 -> x=184 y=432	Piirtää ratkaisun ylhäältä alas	Toimii
Valittu Astar x=186 y=17 -> x=184 y=432	Piirtää ratkaisun ylhäältä alas	Toimii
Valittu Dijkstra x=416 y=23 -> x=28 y=409	Piirtää ratkaisun oikeasta yläkulmasta vasempaan alakulmaan	Toimii
Valittu Dijkstra8 x=416 y=23 -> x=28 y=409	Piirtää ratkaisun oikeasta yläkulmasta vasempaan alakulmaan	Toimii
Valittu Astar x=416 y=23 -> x=28 y=409	Piirtää ratkaisun oikeasta yläkulmasta vasempaan alakulmaan	Toimii
Valittu Dijkstra x=409 y=223 -> x=19 y=219	Piirtää ratkaisun oikealta vasempaan	Toimii
Valittu Dijkstra8 x=409 y=223 -> x=19 y=219	Piirtää ratkaisun oikealta vasempaan	Toimii
Valittu Astar x=409 y=223 -> x=19 y=219	Piirtää ratkaisun oikealta vasempaan	Toimii
Valittu Dijkstra x=420 y=427 -> x=13 y=25	Piirtää ratkaisun oikeasta alakulmasta vasempaan yläkulmaan	Toimii
Valittu Dijkstra8 x=420 y=427 -> x=13 y=25	Piirtää ratkaisun oikeasta alakulmasta vasempaan yläkulmaan	Toimii
Valittu Astar x=420 y=427 -> x=13 y=25	Piirtää ratkaisun oikeasta alakulmasta vasempaan yläkulmaan	Toimii
Valittu Dijkstra x=158 y=438 -> x=160 y=18	Piirtää ratkaisun alhaalta ylös	Toimii
Valittu Dijkstra8 x=158 y=438 -> x=160 y=18	Piirtää ratkaisun alhaalta ylös	Toimii
Valittu Astar x=158 y=438 -> x=160 y=18	Piirtää ratkaisun alhaalta ylös	Toimii

Valittu Dijkstra x=21 y=418 -> x=430 y=21	Piirtää ratkaisun vasemmasta alakulmasta oikeaan yläkulmaan	Toimii
Valittu Dijkstra8 x=21 y=418 -> x=430 y=21	Piirtää ratkaisun vasemmasta alakulmasta oikeaan yläkulmaan	Toimii
Valittu Astar x=21 y=418 -> x=430 y=21	Piirtää ratkaisun vasemmasta alakulmasta oikeaan yläkulmaan	Toimii
Valittu Dijkstra x=16 y=219 -> x=421 y=217	Piirtää ratkaisun vasemmalta oikealle	Toimii
Valittu Dijkstra8 x=16 y=219 -> x=421 y=217	Piirtää ratkaisun vasemmalta oikealle	Toimii
Valittu Astar x=16 y=219 -> x=421 y=217	Piirtää ratkaisun vasemmalta oikealle	Toimii
Valittu Dijkstra ja Näytetään kaikki käydyt pisteet	Näyttää kaikki käydyt pisteet	Toimii
Valittu Dijkstra8 ja Näytetään kaikki käydyt pisteet	Näyttää kaikki käydyt pisteet	Toimii
Valittu Astar ja Näytetään kaikki käydyt pisteet	Näyttää kaikki käydyt pisteet	Toimii
Avataan kuva joka viallinen	Ei lisää kuvaa	EI TOIMI

Suoritus kyky mittauksia

Mitattava algoritmi	Mitatut ajat ms	Keskiaika ms
Valittu Dijkstra x=179 y=430 -> x=179 y=27	344,94,93,93,78	140,4
Valittu Dijkstra8 x=179 y=430 -> x=179 y=27	203,94,109,93,93	118,4
Valittu Astar x=179 y=430 -> x=179 y=27	109,62,47,63,62,	68,6
Valittu Dijkstra x=34 y=413 -> x=414 y=36	94,93,78,78,78	84,2
Valittu Dijkstra8 x=34 y=413 -> x=414 y=36	125,109,124,109,109	115,2
Valittu Astar x=34 y=413 -> x=414 y=36	78,47,63,63,62	62,6
Valittu Dijkstra x=15 y=228 -> x=430 y=216	109,63,78,78,63	78,2
Valittu Dijkstra8 x=15 y=228 -> x=430 y=216	93,109,94,94,94	96,8
Valittu Astar x=15 y=228 -> x=430 y=216	47,63,47,47,62	53,2
Valittu Dijkstra x=24 y=34 -> x=417 y=427	78,78,78,94,78	81,2
Valittu Dijkstra8 x=24 y=34 -> x=417 y=427	109,125,109,125,124	118,4

Valittu Astar x=24 y=34 -> x=417 y=427	62,47,63,47,63	56,4
Valittu Dijkstra x=208 y=20 -> x=209 y=431	78,63,63,63,78	69
Valittu Dijkstra8 x=208 y=20 -> x=209 y=431	125,110,125,109,125	118,8
Valittu Astar x=208 y=20 -> x=209 y=431	63,47,47,47,62	53,2
Valittu Dijkstra x=409 y=36 -> x=23 y=431	78,94,78,78,78	81,2
Valittu Dijkstra8 x=409 y=36 -> x=23 y=431	125,125,109,125,125	121,8
Valittu Astar x=409 y=36 -> x=23 y=431	78,93,78,78,78	81
Valittu Dijkstra x=425 y=233 -> x=13 y=214	94,78,78,78,78	81,2
Valittu Dijkstra8 x=425 y=233 -> x=13 y=214	109,109,109,109,109	109
Valittu Astar x=425 y=233 -> x=13 y=214	47,47,47,47,46	46,8
Valittu Dijkstra x=409 y=424 -> x=19 y=24	78,94,78,78,78	81,2
Valittu Dijkstra8 x=409 y=424 -> x=19 y=24	125,125,110,125,109	118,8
Valittu Astar x=409 y=424 -> x=19 y=24	63,47,63,62,47	56,4

Suorituskyky mittauksia nähdään selvästi eri algoritmien suorituskyky. Eniten aikaa reitin löytämiseen käytti oletetusti Dijkstra8, joka käy kaikki ilmansuunnat läpi reittiä etsiessään. Dijkstra oli toiseksi nopein, mikä oli myös oletettu tulos. Astar, joka käyttää heuristiikka funktiota reitin löytämiseen, oli odotetusti nopein. Ainoa poikkeus oli, kun reittiä etsittiin oikeasta yläkulmasta vasempaan alakulmaan käyttäen Astar algoritmia. Kyseisten tulosten myötä heräsi epäily mahdollisesta virheestä ohjelmassa. Usean tunnin lisätestien jälkeen on todettava, että algoritmi Astar toimii oikein. Jostain syystä juuri kyseiset koordinaatit aiheuttavat algoritmin hidastumisen.

5 Loppuyhteenveto

Projektia tehdessäni opin käytännön kautta, että kattava testaus heti alusta alkaen on erittäin tärkeää. Alusta asti suoritettut testit myös säästävät turhalta työltä, sillä niiden avulla mahdolliset ohjelmointivirheet voi korjata heti virheen ilmentyessä. Mikäli testauksen aloittaa vasta myöhemmin, voi virheen korjaaminen olla huomattavasti

työläämpää. Joitain virheitä on erittäin vaikea, lähes mahdoton, huomata ilman kattavia testejä.