

Testauksen toteutus

Automaattiset testit on luotu jokaiselle algoritmille hyödyntäen Junit testiluokkia. Näitten testien tarkoitus on tarkistaa algoritmin toimivuus erilaisilla syötteillä. Algoritmien ajan testaus tapahtuu itse ohjelmassa, sillä koko ohjelman tarkoitus on vertailla algoritmien aikoja.

Jokaiselle algoritmille ajetaan samat testit, sillä näiden perusasioiden kuuluisi toimia algoritmista huolimatta. Ensimmäinen testi tarkistaa, että taulukon kloonaukset toimii. Hyvin oleellinen ohjelman toiminnan kannalta, mutta itse algoritmin toimivuuden kannalta ei niinkään. Syötetesteissä tarkistetaan aluksi, että itse testissä ei ole vikaa ja katsotaan, että testi tunnistaa epäjärjestyksessä olevan taulukon. Tämän jälkeen testi ajaa algoritmin ja tarkistaa onko taulukko järjestetty oikein.

Aluksi kokeillaan toimivuutta yksinkertaisella [3, 4, 2, 1] taulukolla. Tästä nähdään, että järjestäminen ylipäättensä toimii. Seuraavaksi kokeillaan satunnaisesti luotua miljoona alkion sisältävää taulukkoa. Tämän testin tarkoituksena on katsoa, että algoritmit pystyy käsittelemään erilaisia variaatioita perustestistä. Kolmas syötetesti kokeilee, ettei algoritmi mene sekaisin päinvastaisessa järjestyksessä olevaa [10,9,8,7,6,5,4,3,2,1] taulukkoa järjestäessä. Neljäs syötetesti kokeilee, että ohjelma ei kaadu tai jumitu järjestäessä taulukkoa, joka sisältää monta pelkästään samaa arvoa omaava kokonaisluku. Kokeiluarvona oli 1 ja taulukon pituus 100 lukua. Lopuksi vielä kokeillaan, ettei ohjelma kaadu jos taulukossa on vain yksi kokonaisluku. Kaikki testit tuottivat positiivisen tuloksen.

Itse aikatestausta voi kokeilla ohjelmalla. Esimerkiksi ohjelmasta huomaa, että Quicksort on yleensä muita huomattavasti nopeampi. Esimerkiksi 10000 pituisella syötteellä muilla meni reilu 4 millisekuntia, mutta Quicksortilla vajaa yksi. Heapsort ja Mergesort ovat melko tasaväkisiä satunnaisella syötteellä.