

# TiRaLabra – Viikkoraportti numero 5

---

Kehitin koodia suorituskykytestauksille. Idea oli toistaa kumpaakin algoritmia satunnaisella pakettijärjestyksellä tarvittavalla määrällä toistoja. Siten, että kummallakin algoritmilla ajetaan aina sama järjestys, mitataan kesto, arvotaan uusi järjestys ja mitataan kesto kummallakin jne. Lisäksi suoritusjärjestys rotatoidaan, jotta voidaan eliminoida mahdolline HW/OS:ien vaikutus esim. cachetukset, muistin käyttö ym. Testiä varten on oma luokka Testi, joka saa syötteenä testattavat tavarat ja sakin. Testit ajetaan metodilla ajaTestit(), joka tulostaa mittaustulokset.

Testauksessa ilmeni kuitenkin ongelmia tuloksia tarkastellessa, kun tulosten hajonta oli erittäin suuri. Vaikka ajoin testit samallailman järjestysten vaihtamista 10 miljoonalla toistolla niin hajonta oli edelleen suurta. Tämän selvittämiseen meni aika paljon aikaa. Lopulta syy on ilmeisemmin se, että mittaus oli millisekunneissa, joka oli yksittäisenä ajona liian karkea mittari testin ajamiselle, tämä aiheutti suurta hajontaa. Joudun miettimään nyt testien rakentamisen uudelleen tai ottamalla käyttöön microsekuntitason.

Kuitenkin tulokset antoivat osviittaa, että DP algoritmi on kymmeniä kertoja tehokkaampi kuin naiivi algoritmi, kun tavaroiden lukumäärä kasvaa.

Tuli myös mieleen idea approksimointialgoritmista, joka voisi olla vielä tehokkaampi kun pakettien määrä on iso ja niitä mahtuu paljon säkkiin. Idea olisi lähestyä greedillä menetelmällä eli latoa arvokkaimmat arvo/paino –suhteella sisään.

En tehnyt vielä lisää yksikkötestejä, koska kehittämäni Testi –luokka vielä vaiheessa sen vertaa, että tulee muuttumaan. Tein kuitenkin Testi-luokan ajamisen yhteydessä toisenlaista testailua, kun rakensin sisään tarkistuksen että kummankin algoritmin tuottama ratkaisu tuottaa arvon suhteen saman lopputuloksen joka ajokerralla. Tämä on mielestäni parempi tapa testata ylipäättään, että algoritmit toimivat oikein, kuin yksikkötestauksen kautta niiden metodien testailu.

Jatkan testien suunnittelulla ja suorituskykytestauksen raportoinnilla. Kun tämä on kunnossa aloitan suljettujen tietorakenteiden aukikoodaamisen.