

### Viikkoraportti 3

Sain ratkaistua viimeviikkoisen ongelman, jossa HashMap-tietorakenteeseen ArrayList-taulukoita tallennettaessa mukaan tuli vain taulukon ensimmäinen alkio. Tallennettaessa LZW-sanakirjaan uusia sanoja/merkkijonoja samalla pakkauksen tulosteeseen pitäisi tulla yhtä yksikköä lyhyempi sana/merkkijono. Yksinkertainen virhe johtui siitä, että ohjelma poisti sanakirjaan tallennuksen jälkeen ArrayLististä viimeisen merkin, jolloin se hävisi myös HashMap-sanakirjasta. Ikävä kyllä korjaus täytyi tehdä koodia monimutkaistamalla ja joku elegantimpi ratkaisu on varmaan mahdollinen. Muutenkin koodi tällä viikolla sotkeentui kaikenlaisen testailun vuoksi, koska tämän jälkeen eteen tuli vaikeammin paikallistettava virhe pakkauksen purkuvaiheesta. Esimerkiksi n. 200 kilon kokoisesta tekstitiedostosta tuli kymmenisen virheilmoitusta keskeltä tekstiä paukkauksen purkamisen aikana. Aluksi näytti, että virheet johtuivat rivinvaihdoista, mutta syynä oli tietysti merkeistä riippumaton ongelma, joka esiintyi kun tekstissä sama merkki toistuu peräkkäin. Huomasin, että internetissä löytämien esimerkkien perusteella rakentamani purkualgoritmi ei toimikaan aivan 100-prosenttisesti oikein vaan jo pelkän "aa"-merkkijonon pakkauksen purku sai aikaan virheen. Purkuvaiheessa vastaan tulee a-kirjain ja vielä tuntematon symboli, jonka algoritmin pitäisi osata päätellä. "aa"-esimerkissä tämä onnistuu helposti, mutta keskellä tekstiä useampien kirjainten esiinnyttyä, en keksinyt itse ratkaisua. Koska tähän pieneen ongelmaan ja jo sen paikallistamiseen kului tuntikaupalla aikaa, päätin palata vasta seuraavalla LZW-algoritmia selittävien tekstilähteiden pariin tai keksiä ratkaisu itse. Tämän vuoksi myöskään testit eivät vielä 100-prosenttisesti toimi.

Positiivista kuitenkin tällä viikolla oli, että tekstitiedostojen luku, pakkaus ja purku näyttivät toimivan HashMap-rakenteella nopeasti. Jouduin kuitenkin tekemään ehkä arveluttavan ratkaisun ja laittamaan pakkausvaiheessa sanat eli tavumuotoon muutetut merkit ArrayList-taulukoihin HashMapin avainarvoiksi, jotta niihin pääsisi get-metodilla käsiksi ja sanakirjan koodit puolestaan HashMapissa arvoiksi. Saa nähdä, miten tästä edetä, kun kehittää omat tietorakenteet tilalle.

Tällä viikolla oli tarkoitus myös ehtiä tehdä menetelmät, joilla muuntaa ja tallentaa suurempibittiset kokonaisluvut tallennuskelpoisiksi 8-bittisiksi tavuiksi ja joilla kääntää erikokoisia bittijonoja 8-bittiseen muotoon. Ehdin jo aloittaa tekemisen, mutta en viitsinyt julkaista vielä mitään, jos lisäksi vaaditaan dokumenttien ja testien täydellinen kattavuus.