

Ilkka Lehtikoinen  
013787637

Aineopintojenharjoitustyö:  
Tietorakenteet ja algoritmit

## Verkon vahvasti yhtenäiset komponentit

### *Käyttöohje*

Ohjelma koostuu kuudesta luokasta: VYK, Kosaraju, Tarjan ja PathBased sekä apuluokista Pino, Solmu, HakuPuu ja PuuSolmu. VYK-luokka suorittaa suorituskkyttestausta eri algoritmeille, joilla etsitään verkosta vahvasti yhtenäisiä komponentteja ja toimii pääohjelmalla.

### **Ajatiedostonluominen**

Ohjelma on jaettu kolmeen pakkaukseen vyk, algoritmit ja tietorakenteet. Paketissa tietorakenteet sijaitsevat apuluokat Pino, Solmu, HakuPuu ja PuuSolmu, joita algoritmit paketissa sijaitsevat luokat Kosaraju, Tarjan ja PathBased käyttävät. Lopuksi on vielä pääpakkaus vyk, johon muut paketit myös kuuluvat.

Komentoriviltä ajettavan jar paketin voi luoda esimerkiksi netbeans-ohjelmalla **Build Project** -komennolla.

### **Ohjelman suoritus**

Ohjelma toimii kahdella tavalla. Ilman parametreja ohjelma generoi joukon erikokoisia suunnattuja verkkoja, joille se suorittaa kunkin algoritmin ja laskee suoritukseen käytetyn ajan, jonka se palauttaa mikrosekunteina. Ajan tulostus tapahtuu mikrosekunteina, koska se on pienin kokonaisluku, jonka ohjelman suoritus tuottaa yhden solmun kokoisella verkolla. Ohjelmalle voi myös antaa parametreina solmujen määrän ja tiheyden, jolloin ohjelma generoi annetuilla parametreilla suunnatun verkon ja laskee kaikilla algoritmeilla sen suoritukseen kuluva ajan.

Ilman parametreja ohjelman suoritus tapahtuu komennolla

**\$ java -Xss1024K -jar VYK.jar**

Tällöin ohjelma testaa erikokoisia verkkoja yhdestä solmusta 5000 solmuun. Parametria -Xss1024K tarvitaan, sillä muutoin ohjelman suoritukseen varattu muisti ylittyy rekursiopinon vuoksi yli 2500 solmun verkoilla.

Ohjelma tulostaa kullekin verkolle solmujen määrän ja verkon tiheyden, sekä algoritmin nimen ja sen perässä sen suorittamiseen käytetyn ajan mikrosekunneissa.

**Solmuja: 2000 Tiheys: 0.2**

**Kosaraju: 2050054 us**

**Tarjan: 2869102 us**

**PathBased: 4184021 us**

Esimerkki 1. ohjelman tulostuksesta.

Ohjelmalle voi myös antaa verkon solmujen määrän positiivisena kokonaislukuna (int) ja kaarien tiheyden liukulukuna nollan ja yhden välillä (double).

**\$ java -Xss1024K -jar VYK.jar 150 0.12**

Ohjelma tulostaa samat tiedot kuin edelläkin, mutta nyt vain tälle annetulle solmulle.

**Solmuja: 150 Tiheys: 0.12**

**Kosaraju: 97510 us**

**Tarjan: 66939 us**

**PathBased: 76356 us**

Esimerkki 2. ohjelman tulostuksesta.

Solmujen maksimi määrää ei ole rajoitettu, mutta ohjelmalla käytössä oleva muistimäärä rajoittaa sitä. Itse olen testannut 10 000 solmun verkoilla ja kasvattamalla käytössä olevaa muistia -Xss parametrilla sopivasti voidaan jonkin verran isompiakin verkkoja testata.

## ***Algoritmit***

Ideana on että algoritmeja voisi käyttää ilman pääluokkaa vyk, tällöin kutsuva metodi antaa verkon vierusmatriisimuodossa, jossa kaarta merkkää luku 1. Esimerkiksi:

**Kosaraju.kosaraju(verkko)**

Tällöin algoritmi palauttaa verkon merkkijonona (String), jossa vahvasti yhtenäiset komponentit ovat erotettu aaltosuluilla.

**{{0,1,4}{2}{3,7}{5,6}}**