

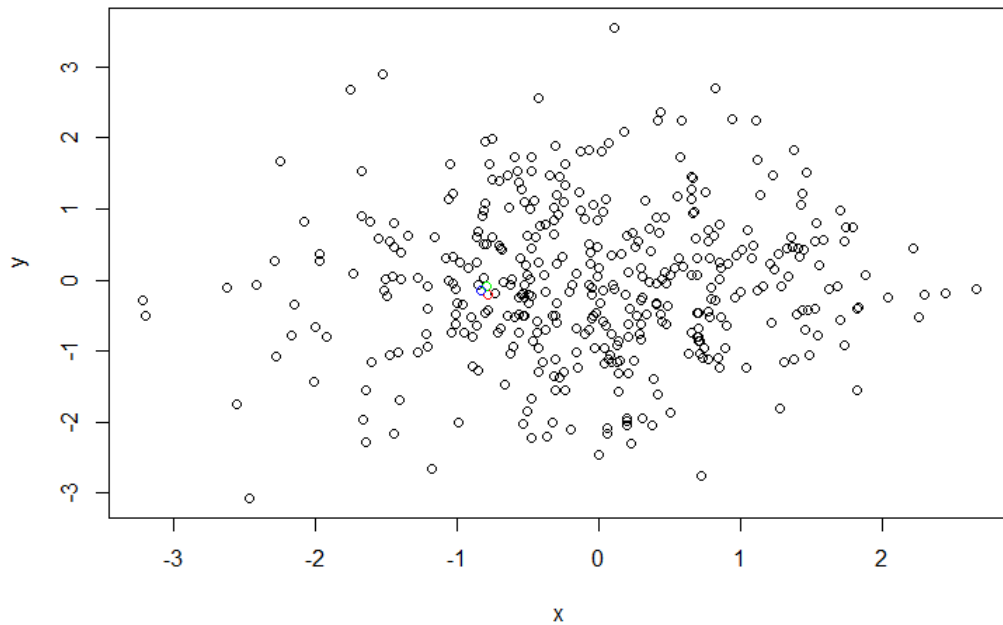
Tietorakenteet 2018 harjoitustyö

Työssä toteutetaan Javalla graafi tietorakenne, jonka avulla tarkastellaan tasossa olevien *pisteiden suhteita toisiinsa*. Työtä varten annetaan kaksi tiedostoa. T2018.java ja Tdata.txt. T2018.java sisältää esimerkin tason pisteiden lukemiseksi tekstitiedostosta sekä pisteiden kirjoittamisen tiedostoon. Tdata.txt sisältää esimerkkidatan graafin luomista varten. Graafi toteutetaan linkitettyinä rakenteena, jossa yksittäinen solmu voi olla esimerkiksi

```
public class Node {  
  
    private float x; //Pisteen x-koordinaatti.  
  
    private float y; //Pisteen y-koordinaatti.  
  
    private Node neighbors[]; //Linkit pisteen "naapureihin"  
  
    private boolean visited;  
  
}
```

Voit vapaasti valita muita muuttujia sekä metodeja jotka katsot tarpeellisiksi. Työssä toteutetaan seuraava toiminnallisuus.

1. Graafi luodaan lukemalla esimerkkitiedostosta pisteet. Jokaista pistettä varten luodaan oma solmu sekä linkki kyseistä solmua lähinnä olevaan solmuun. Etäisyysmittana käytetään Euklidista etäisyyttä. Tämän vaiheen tuloksena saadaan joukko aligraafeja, joissa on joko kaksi tai kolme yhdistettyä solmua. (yhdistetyt komponentit)
2. Lisätään graafin jokaiselle solmulle toiseksi lähinnä olevan pisteen sisältämä solmu
3. Kirjoitetaan tiedostoon BFS.txt solmujen koordinaatit leveyshaun antamassa järjestyksessä
4. Kirjoitetaan tiedostoon DFS.txt solmujen koordinaatit syvyysshaun antamassa järjestyksessä
5. Kirjoitetaan tiedostoon Degrees.txt solmujen lähtö- ja tuloasteet
6. Poistetaan annettu solmu graafista ja tulostetaan jäljelle jäänyt graafi tiedostoon DIM.txt leveyshaun mukaisessa järjestyksessä
7. Lisää solmuille lähimpiä naapureita, kunnes graafista tulee yhtenäinen, eli mistä tahansa solmusta voidaan käydä kaikissa graafin muissa solmuissa ja palata takaisin lähtösolmuun. Tulosta graafi syvyysshaun mukaisessa järjestyksessä tiedostoon COMP.txt
8. Muodostetaan graafille minimivirittävä puu ja tulostetaan se tiedostoon MSP.txt leveyshaun mukaisessa järjestyksessä
9. Ensimmäisessä toiminnallisuudessa tuloksena oli kahden tai kolmen solmun aligraafeja. Vertaile näitä ja mieti, millainen rooli kolmen solmun aligraafilla voisi olla
10. Muuta pisteiden etäisyyslaskentaa sopivasti ja muodosta graafi, jossa on mukana pistejoukon "reuna-alueen" pisteet sisältävät solmut. Tulosta graafi leveyshaun mukaisessa järjestyksessä tiedostoon OUTLIERS.txt. Kts kuva 1. Voit itse määritellä reuna-alueen.



Kuva 1. Pisteparven reuna-alueilla olevat pisteet voidaan usein tulkita poikkeaviksi arvoiksi "outliers".

Työn palautus

Työ palautetaan viimeistään 31.1.2019. Palautus tehdään sähköpostilla osoitteeseen [jyrki.rasku\(at\)uta.fi](mailto:jyrki.rasku@uta.fi) otsikolla Tira2018. Pakkaa lähdekoodisi yhteen tiedostoon käyttämällä 7z tai zip pakkausta. Palautetun ohjelman pitää kääntyä komennolla `javac T2018.java` ja olla ajettavissa komennolla `java T2018`. Liitä vastauksesi toiminnallisuuskohtaan 9 kommenttina lähdekooditiedostosi alkuun. Ilmoita palautuksesi yhteydessä myös se, mitkä toiminnallisuuksista 1-10 olet toteuttanut.

Arvostelusta

Työ, joka toteuttaa edellä mainitut kymmenen toiminnallisuutta on kymmenen pisteen arvoinen. Minimivaatimus on kohtien 1-3 täyttäminen.