

## MAT.APP.270 Syksy 2021

### Kolmannen (ja viimeisen) ohjelmointiharjoitustyön tehtävänanto

Tehtävänä on analysoida graafissa tiettyjä todennäköisyyksiä. Voit valita kahdesta eri vaihtoehdosta.

Syöte on molemmissa tapauksissa graafi, joka esittää (jonkinlaista) sosiaalista verkostoa. Graafin koko vaihtelee, mutta voit luottaa siihen, ettei graafissa ole enempää kuin 3000 solmua missään tapauksessa. Graafi on kaikissa tapauksissa suunnattu ja se voi olla painotettu tai painottomaton.

Palautetun ohjelman tulisi analysoida graafin rakenne ja sen jälkeen palauttaa arvio siitä, kuka “voittaa” annetun pelin. Graafin solmut ovat “pelaajia”. Sanomme, että pelaajalla  $u$  on yhteys pelaajaan  $v$  jos  $(u, v)$  on kaari.

Pelejä on kaksi:

1. Ensimmäinen peli on nimeltään “**Kapularalli**”. Tässä pelissä ensin arvotaan satunnaisesti joku pelaaja, joka saa kapulan. Tämän jälkeen pelaaja lähettää kapulan jollekin sellaiselle pelaajalle, johon hänellä on yhteys. Tämä pelaaja valitaan arpomalla.

Jos graafi on painotettu, niin arpominen suoritetaan niin että pelaajan  $v$  todennäköisyys tulla valituksi on

$$p_v = \frac{w(u, v)}{\sum_t w(u, t)}$$

Kun joku pelaaja saa kapulan, tämä suorittaa samanlaisen arvonnän. Kuitenkin pienellä todennäköisyydellä  $d$ , joka annetaan pelin alussa, kapula annetaan uudelle pelaajalle.

Peli pyörii niin kauan kunnes musiikki lakkaa. Pelaaja, jolla kapula on, kun musiikki lakkaa, saa pisteen. Voit olettaa että musiikki pyörii niin kauan että kapula ehtii vaihtaa haltijaa tuhansia kertoja.

Peliä pelataan myös lukuisia kierroksia niin, että voidaan olettaa, että käytännössä kaikki pelaajat saavat pistetiä. Tehtävänä on laittaa pelaajat järjestykseen sen mukaan kuinka paljon heidän voi olettaa saavan pisteitä kun peliä on pelattu tuhansia kierroksia.

2. Tämä peli on nimeltään **“Juorukello”**. Satunnaiselle pelaajalle kerrotaan “juoru” joka koskee toista pelaajaa. Kun pelaaja kuulee juorun, hän kertoo juorun kaikille, joihin hänellä on yhteys. Kun pelaaja kuulee juorun pelaajalta  $u$ , hän lisää juorun tiedon “kuulin pelaajalta  $u$ , että...  $x$ ”. Esimerkiksi jos pelaaja 1 kertoo juorun pelaajalle 2 ja niin edelleen, niin pelaaja 4 sanoisi “kuulin pelaajalta 3, että hän kuuli pelaajalta 2, että hän kuuli pelaajalta 1 että... $x$ ”.

Jos graafi on painotettu, voidaan kaaren paino tulkita siten, että juorulta menee  $w(u, v)$  tuntia kulkeutua pelaajalta  $u$  pelaajalle  $v$ . Jos graafi on painottomaton, voit olettaa että  $w(u, v) = 1$ .

Kukin pelaaja kertoo eteenpäin vain ensimmäisenä kuulemansa juorun.

Kun juorun kohde eli  $x$  (joka on satunnaisesti valittu) kuulee juorun, saa jokainen hänen (ensiksi) kuulemassaan juorussa esiintynyt pelaaja yhden “juorukellopisteen”.

Tehtävänä on ennustaa pelaajien järjestys juorukellopelissä, eli laittaa pelaajat järjestykseen sen mukaan, kuinka monta juorukellopistettä he saavat jos peliä pelataan tuhansia kertoja.

Voit olettaa, että graafi on vahvasti kytketty.

Tee tehtävä seuraavalla tavalla:

1. Toteuta funktio joka ottaa parametrinaan graafin  $G$ , joka on joko painotettu tai painottamaton. Funktion nimi pitäisi olla joko *kapularalli* tai *juorukello*.  
Funktion pitäisi palauttaa lista, jossa on kaikki graafin solmut pistejärjestyksessä. Itse pisteytystä ei tarvitse palauttaa funktiossa.
2. Tee pdf-muotoinen selostus siitä, mitä algoritmia käytät ja miksi. Jos siirrät graafin tiedot johonkin toiseen tietorakenteeseen tai jos käytät jotain valmiita kirjastoja (Numpy tms) niin mainitse se selostuksessasi. Kaikenlaisten valmiiden kirjastojen käyttö on sallittua, mutta se pitää mainita selostuksessa.
3. Palauta mukana myös testiaineistona itse generoimiasia graafeja ainakin kaksi.
4. Ainoa tekniikka, joka *ei* ole sallittu, on sellainen jossa rakennetaan suoraan peliä pelauttava simulaatio, esim Monte Carlo-simuloinnin käyttö ei ole sallittua. Analysoivan algoritmin pitäisi tutkia graafin rakennetta, ei simuloida em. peliä.
5. Voit kuitenkin tehdä tällaisen simulaattorin testataksesi kuinka hyvin algoritmisi ennustaa voittajia. Moodle-sivulle tulee esimerkkisyötteitä ja mahdollisesti myös simulaattori myöhemmin.