

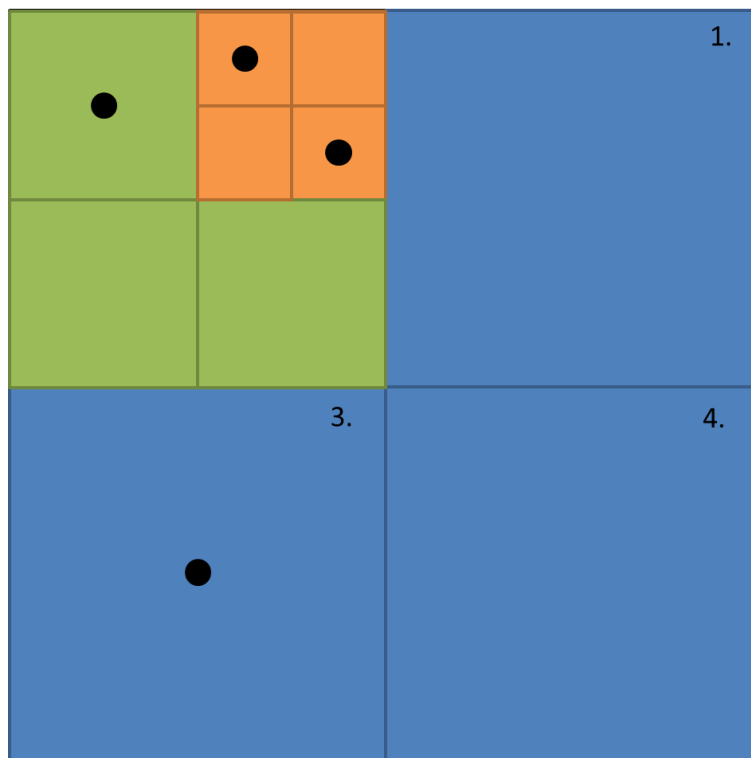
Tietorakenteet 2014 harjoitustyö

Harjoituksen aihe

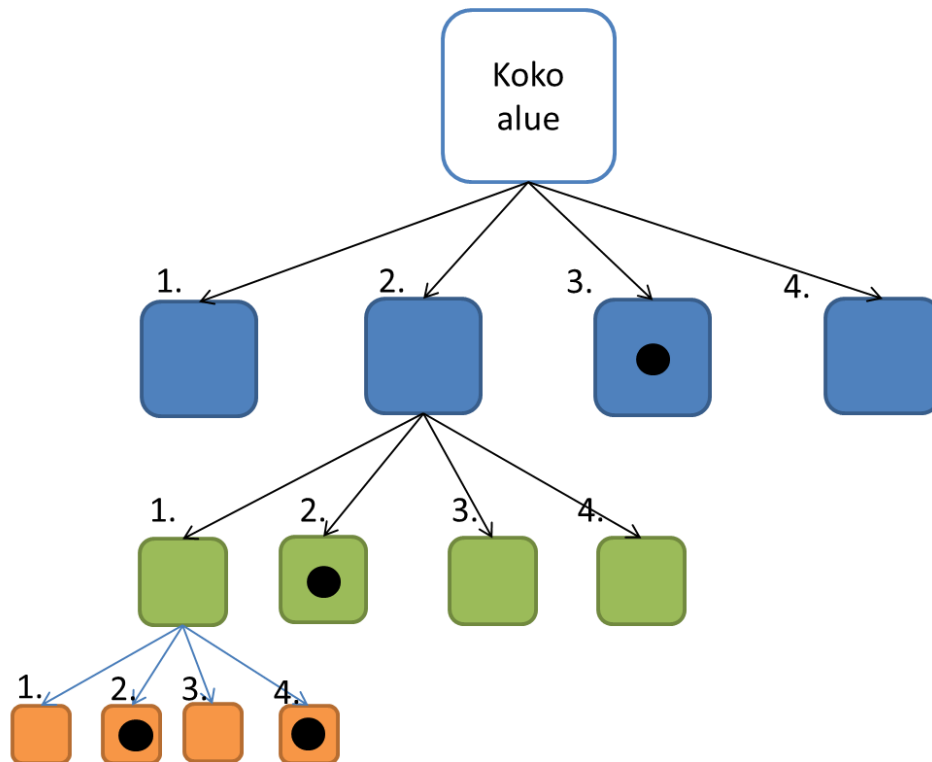
Harjoitustyönä toteutetaan Javalla quadtree-tietorakenne, joka mahdollistaa kaksiulotteisten *kokonaislukupisteiden* tallettamisen, hakemisen ja poistamisen. Quadtree-tietorakenne voidaan ajatella binaaripuun laajennoksena, jossa yksittäisen avainarvon sijasta talletetaan tietoa kaksiulotteisen avaimen avulla. Tätä tietorakennetta käytetään yleensä erilaisten geometrinen sovellusten yhteydessä. Esimerkiksi tietokoneanimaatiossa sitä käytetään kohteiden näkyvyyden sekä niiden välisten törmäysten tarkastelun nopeuttamiseen. Kolmiulotteisten kohteiden tapauksessa käytetään ns. octree- tietorakennetta, joka on quadtreen suoraviivainen yleistys kolmiulotteiseen tilaan. Käyttöidealtaan quadtree ja octree ovat samankaltaiset.

Quadtree

Quadtree on puurakenne, jossa jokaisella solmulla on *täsmälleen* neljä lapsisolmuja. Kaksiulotteisen tilan jakamisessa puun juurisolmu edustaa koko jaettavaa aluetta. Tarkastellaan puurakennetta alla olevien kuvien avulla. Kuvassa 1 koko alue on jaettu ensin neljään neljännekseen 1., 2., 3. ja 4. Toinen neljännes on edelleen jaettu neljään osaan. Lopuksi toisen neljänneksen ensimmäinen neljännes on jaettu neljään yhtäsuureen osaan. Mustat pisteet kuvaavat puuhun talletettuja pisteitä. Pisteiden sijainti näkymää vastaavassa quadtree tietorakenteessa on esitetty kuvassa 2.



Kuva 1. Näkymän tilajakoesitys



Kuva 2. Kuvan 1 näkymän quadtree-esitys

Toteutuksesta

Harjoituksessa käytettävät luokat ovat Tira.java, QuadTree.java ja Node.java. Valmiita luokkia ei tietorakenteen toteuttamisessa saa käyttää. Tulostukseen ja tiedostojen lukemiseen valmiiden luokkien käyttö on tietysti sallittu. Jaettavan alueen maksimikokona käytetään kakkosen potensseja, jolloin alueen jakaminen on helppoa. Jaettavan alueen maksimikoko annetaan QuadTree-luokan rakentimessa ja se on oletusarvoisesti 1024x1024. Työn pohjana käytetään Tira.java- sekä testdata.txt- tiedostoja.

Tira-luokka lukee tekstitiedostosta puuhun lisättävät, puusta haettavat ja siitä poistettavat pisteet. Tässä työssä yhteen solmuun talletetaan vain yksi piste. QuadTree-luokka pitää huolen pisteiden lisäämisestä (Insert(p)), hakemisesta (Find(p)) ja poistamisesta (Remove(p)). Jos poistettavaa pistettä ei löydy puusta, siitä ilmoitetaan käyttäjälle. Jos piste löytyy, tulostetaan komentorivin yhdelle riville sen alueen vasemman yläreunan ja oikean alareunan koordinaatit, jossa piste sijaitsee. Jokaisen poiston ja lisäyksen jälkeen tulostetaan puun sisältö komentoriville esijärjestyksessä. Sopimuksen mukaan näkymän vasen yläreuna on origo (0,0), x-koordinaatti kasvaa vasemmalta oikealle ja y-koordinaatti ylhäältä alas. Node-luokka sisältää tarvittavat osoittimet neljään lapseen, metodit osoittimien asettamista varten sekä tilan pisteen tallettamista varten. Lisäksi Node-luokka sisältää tiedon peittämästään alueesta ja osaa kertoa, kuuluuko lisättävä piste-ehdokas alueeseen vai ei. Materiaalia toteutuksen tueksi löytyy verkosta esimerkiksi <http://www.cs.umd.edu/~hjs/pubs/Samet85k.pdf>. Katso myös <http://www.youtube.com/watch?v=fuexOsLOfI0>.

Palautus

Työ palautetaan 31.1.2015 mennessä sähköpostilla osoitteeseen jyrki.rasku(at)uta.fi. Palautusviestin otsikko on Tira2014. Tira.java tiedostosta kommentoidaan pois lopun testiosuus siten, että ohjelman voi kääntää ja ajaa suoraan samanmuotoisella testitiedostolla, kun testdata.txt on. Ole hyvä ja pakkaa kaikki tiedostot yhteen nippuun 7z muotoon. Testdata.txt tiedostoa ei palauteta.

Arvostelu

Ajoissa palautetusta ja oikein toimivasta työstä saa kymmenen pistettä. Toimintavirheet Insert-, Find- ja Remove- metodeissa sekä luokkarakenteesta poikkeaminen vähentävät pisteitä.