

Määrittelydokumentti

Tässä tietorakenteiden harjoitustyössä toteutetaan erilaisten hakupuiden tehokkuutta vertaileva ohjelma.

Mahdollisia vertailukohteita ovat vasemmalle nojaavat punamusta-puut, kaksi-kolme-hakupuurakenne, treap ja threaded-puut.

Punamusta-puut ovat itsestään tasapainottuvia rakenteita, joidenka tasapaino säilyy kolmen invariantina avulla: jokaisella punaiseksi värjättyllä solmulla on kaksi musta lasta (null = musta)m jokainen polku juuresta lehteen sisältää saman määrän mustia solmuja, ja vain vasemmat lapsisolmut voivat olla punaisia.

Treapit ovat myös keskimäärin tasapainoisia. Treapit ovat puita, jotka on ensin järjestetty perinteiseen binäärihakupuun tapaan. Tasapaino syntyy solmuille tallennettavista toisista ominaisuuksista, satunnaisesti arvotuista 'prioriteeteista', joihinka perustuen rakenne pidetään myös maksimikekona.

Kaksikolme-puut eivät ole "virallinen" puurakenne, vaan oma versioni hakupuusta. Kaksikolme-puu on mallinettu B-puun mukaan, mutta siitä puuttuu B-puissa käytetty tasapaino-ominaisuus. Kaksikolme puun solmuihin voidaan tallentaa joko yksi tai kaksi arvo. Yksiarvoisen solmun lapset jakautuvat kuten normaalissa binäärihakupuussa, mutta kaksiarvoisella solmulla voi olla kolmas lapsi. Lapset tässä tilanteessa jakautuvat siten, että vasemmalla on molempia solmuja pienemmät arvot, oikealla suuremmat, ja keskimmäiseen lapsipuuhun tallennetaan vanhemman kahden arvon väliset arvot.

Threaded-puut eivät myöskään ole automaattisesti tasapainottuvia. Niissä on solmut saavat lapsisolmuiksi null-viitteet vain kaikista pienimmän ja kaikista suurimman solmun tapauksissa. Muuten tyhjiin lapsiviitteisiin tallennetaan kyseisen solmun seuraaja ja/tai edeltä.

Ohjelmassa on mahdollista vertaillaan perusoperaatioiden (insert, delete, search) lisäksi myös puiden alkioiden tulostusta suuruusjärjestyksessä.

Harjoitustyön tarkoituksena on lähinnä harjoitella eri puiden rakentamista. Aikavaativuuksista Treap ja Punamustapuut toimivat vaativuudella $O(\log n)$, Threaded ja Kaksikolmepuut $O(n)$.

Käyttäjän on mahdollista valita syötteen koko ja alkioiden lisäämisjärjestys. Käyttäjä voi katsoa kunkin puun operaatioiden suoritusajat mikrosekunneissa. Lisäksi on mahdollista katsoa tietyn syötteen nopeiten ja hitaiten suoriutuvat puut.

Tilavaativuus tulee siis olemaan luokkaa $O(n)$, jossa n = puiden alkioiden määrä x puiden määrä + apumuuttujat (korkeintaan $O(n)$; poistoa yms. varten, puihin tallennetuista alkioista pidetään kirjaa listaesityksenä). Aikavaativuus on maksimissaan $O(n)$, kun vertaillaan alkoioiden tulostamista, ja minimissä $O(\log n)$ kun vertailussa ei tutkita alkioiden tulostamista ja kyseessä on toinen automaattisesti tasapainottuvista puista.

Lähteet:

http://en.wikipedia.org/wiki/Red%E2%80%93black_tree

<https://gist.github.com/rkapsi/741080>

http://en.wikipedia.org/wiki/Left-leaning_red%E2%80%93black_tree

<http://www.mew.org/~kazu/proj/red-black-tree/>

<http://en.wikipedia.org/wiki/Treap>

<http://en.wikipedia.org/wiki/B-tree>

http://en.wikipedia.org/wiki/Threaded_binary_tree