

Tietorakenteet ja algoritmit(syksy 2012)

Harjoitus 5 (5.10.2012)

- (a) Alunperin tyhjäan binäärihakupuuhun lisätään materiaalin insert-operaatiota käyttäen avaimia seuraavassa järjestyksessä: 2, 0, 3, 7, 9, 1, 5, 6 ja 8. Näytä miten puu kasvaa lisäysten seurauksena. Piirrä siis kuva tilanteesta jokaisen (tai ainakin melkein jokaisen) lisäyksen jälkeen.

(b) Simuloi delete-operaation toimintaa tuloksena olevalle puulle parametrina solmu, jossa avain 3. Poista tämän jälkeen avain 9. Tämän jälkeen avain 2. Ja lopuksi avain 5. Piirrä jokaisen operaation jälkeen tuloksena oleva puu.
2. Esitä algoritmi, joka saa syötteen binääripuun juurisolmun ja laskee puun solmujen lukumäärän. Mikä on algoritmisi aika- ja tilavaativuus?
- (a) Piirrä mahdollisimman vähän solmuja sisältävä AVL-puu, jonka korkeus on neljä. Piirrä mahdollisimman monta solmua sisältävä AVL-puu, jonka korkeus on neljä.

(b) Oletetaan, että AVL-puun kunkin solmun *height*-kentälle varataan kahdeksan bittia (eli suurin luku on 255). Kuinka monta solmua puussa pitäisi olla, että juuren *height* vuotaisi yli? (Suuruusluokka-arvio riittää.)
- (a) Näytä miten AVL-insert toimii kun tyhjäan puuhun lisätään järjestyksessä seuraavat avaimet 41, 38, 31, 12, 19, 8, 27 ja 49. Näytä jokaisen lisäyksen yhteydessä tehtävät kierto-operaatiot. Tämän jälkeen näytä miten AVL-delete toimii kun se poistaa tuloksena olevasta puusta järjestyksessä avaimet 12, 49, 31 ja 8.

(b) Sama kuin edellä, mutta avaimet lisätään alussa tyhjäan puuhun päinvastaisessa järjestyksessä eli 49, 27, 8, 19, 12, 31, 38 ja 41. Tämän jälkeen poistetaan siitä järjestyksessä avaimet 12, 8, 49 ja 41
- (a) Binäärihakupuun operaatioiden *min* ja *max* (pienimmän ja suurimman alkion etsiminen) aikavaativuus on $\mathcal{O}(h)$, jossa h on puun korkeus. Miten puun pseudokooditoteutusta tulisi muuttaa, että operaatioiden *min* ja *max* aikavaativuus olisi vain $\mathcal{O}(1)$ ja muiden operaatioiden aikavaativuus säilyy ennallaan.

(b) Binäärihakupuun operaatioiden *succ* ja *pred* (seuraavan ja edellisen alkion etsiminen) aikavaativuus on $\mathcal{O}(h)$, jossa h on puun korkeus. Miten puun pseudokooditoteutusta tulisi muuttaa, että operaatioiden *succ* ja *pred* aikavaativuus olisi vain $\mathcal{O}(1)$ ja muiden operaatioiden aikavaativuus säilyy ennallaan.

(c) Mitä hyötyä ja haittaa edellisistä muutoksista on?
6. Osoita, että mikä tahansa binäärihakupuun voidaan muuntaa miksi tahansa toiseksi samat avaimet sisältäväksi binäärihakupuuksi suorittamalla sopiva jono kiertoja. Arvioi tarvittavien kiertojen lukumäärää. (Tässä puiden siis ei tarvitse olla tasapainoisia.)

Vihje: osoita ensin, että mikä tahansa puu voidaan muuntaa oikealle suuntautuvaksi ketjuksi (eli sellaiseksi, että millään solmulla ei ole vasenta lasta).