

Toteutusdokumentti

Tomi Heiskanen

8. joulukuuta 2014

Tietorakenteet

Tietorakenteet on toteutettu omina java-luokkina. Kaikissa puissa on solmuksi nimetty alkio ja se on toteutettu omana luokkanaan ja siinä on tarvittavat ominaisuudet jokaiselle puulle.

O-analyysit

Binäärihakupuu

Lisääminen: Koska metodissa olevan loopin operaatiot ovat vakioaikaisia. ja looppi suoritetaan korkeintaan n -kertaa, sillä lisäykset voivat pahimmassa tapauksessa tapahtua suuruusjärjestyksessä, saadaan lisäämisen aikavaativuudeksi $O(n)$.

Poistaminen: Metodissa olevat operaatiot ovat vakioaikaisia. Solmun etsiminen suoritetaan toisella metodilla ja se on rekursiivinen. Rekursio suoritetaan pahimmillaan n - kertaa, sillä lisäämiset voivat tapahtua suuruusjärjestyksessä tai jos puuhun lisätään ja poistetaan sopivasti siten, että puu vastaa listaa. Poistamiselle saadaan siten aikavaativuudeksi $O(n)$.

Etsiminen: Etsiminen suoritetaan rekursiivisella metodilla ja rekursio suoritetaan pahimmassa tapauksessa n -kertaa, kuten edellä todettiin. Etsimisen aikavaativuudeksi saadaan $O(n)$.

Pienimmän ja suurimman alkion etsiminen: Pienin ja suurin alkio löytyvät aina lehdestä tai juuresta ja siten metodissa oleva looppi suoritetaan pahimmassa tapauksessa n -kertaa. Aikavaativuudeksi saadaan tällöin $O(n)$.

AVL-puu

Lisääminen: Lisääminen tapahtuu ensin kuten binäärihakupuulle ja sen aikavaatimus on $O(h)$, missä h on korkeus ja koska AVL-puu on tasapainoinen, niin lisäämisen aikavaatimus on $O(\log n)$. Koska lisäämisen jälkeen suoritettavat kierrot ovat vakioaikaisia ja niitä suoritetaan korkeintaan kaksi kertaa ja pahimmassa tapauksessa kuljetaan lisätystä solmusta juureen, niin tämän aikavaativuudeksi saadaan $O(\log n)$. Lisääminen koostuu siis kahdesta aikavaativuudeltaan $O(\log n)$ osasta ja siten lisäämisen aikavaativuus on $O(\log n)$.

Poistaminen: Poistaminen tapahtuu ensin kuten binäärihakupuulle ja sen aikavaatimus on $O(h)$, missä h on korkeus ja koska AVL-puu on tasapainoinen, niin poistamisen aikavaatimus on $O(\log n)$. Koska poistamisen jälkeen suoritettavat kierrot ovat vakioaikaisia ja niitä suoritetaan korkeintaan kaksi kertaa ja pahimmassa tapauksessa kuljetaan poistettusta solmusta juureen, niin tämän aikavaativuudeksi saadaan $O(\log n)$. Poistaminen koostuu siis kahdesta aikavaativuudeltaan $O(\log n)$ osasta ja siten poistamisen aikavaativuus on $O(\log n)$.

Etsiminen: Etsiminen suoritetaan rekursiivisella metodilla ja rekursio suoritetaan korkeintaan $\log n$ kertaa. Etsimisen aikavaativuudeksi saadaan $O(\log n)$.

Pienimmän ja suurimman alkion etsiminen: Pienin ja suurin alkio löytyy aina lehdestä ja metodissa oleva looppo suoritetaan siis $\log n$ -kertaa. Aikavaativuudeksi saadaan tällöin $O(\log n)$.

AVL-puu

Lisääminen: Lisääminen tapahtuu ensin kuten binäärihakupuulle ja sen aikavaatimus on $O(h)$, missä h on korkeus ja koska AVL-puu on tasapainoinen, niin lisäämisen aikavaatimus on $O(\log n)$. Koska lisäämisen jälkeen suoritettavat kierrot ovat vakioaikaisia ja niitä suoritetaan korkeintaan kaksi kertaa ja pahimmassa tapauksessa kuljetaan lisäystä solmusta juureen, niin tämän aikavaativuudeksi saadaan $O(\log n)$. Lisääminen koostuu siis kahdesta aikavaativuudeltaan $O(\log n)$ osasta ja siten lisäämisen aikavaativuus on $O(\log n)$.

Poistaminen: Poistaminen tapahtuu ensin kuten binäärihakupuulle ja sen aikavaatimus on $O(h)$, missä h on korkeus ja koska AVL-puu on tasapainoinen, niin poistamisen aikavaatimus on $O(\log n)$. Koska poistamisen jälkeen suoritettavat kierrot ovat vakioaikaisia ja niitä suoritetaan korkeintaan kaksi kertaa ja pahimmassa tapauksessa kuljetaan poistettusta solmusta juureen, niin tämän aikavaativuudeksi saadaan $O(\log n)$. Poistaminen koostuu siis kahdesta aikavaativuudeltaan $O(\log n)$ osasta ja siten poistamisen aikavaativuus on $O(\log n)$.

Etsiminen: Etsiminen suoritetaan rekursiivisella metodilla ja rekursio suoritetaan korkeintaan $\log n$ kertaa. Etsimisen aikavaativuudeksi saadaan $O(\log n)$.

Pienimmän ja suurimman alkion etsiminen: Pienin ja suurin alkio löytyy aina lehdestä ja metodissa oleva looppి suoritetaan siis $\log n$ -kertaa. Aikavaativuudeski saadaan tällöin $O(\log n)$.

Punamustapuu

Lisääminen: Lisääminen tapahtuu ensin kuten binäärihakupuulle ja sen aikavaatimus on $O(h)$, missä h on korkeus ja koska punamustapuu on tasapainoinen, niin lisäämisen aikavaatimus on $O(\log n)$. Koska lisäämisen jälkeen suoritettavat kierrot ovat vakioaikaisia ja niitä suoritetaan korkeintaan kaksi kertaa ja pahimmassa tapauksessa kuljetaan lisätystä solmusta juureen, niin tämän aikavaativuudeksi saadaan $O(\log n)$. Lisääminen koostuu siis kahdesta aikavaativuudeltaan $O(\log n)$ osasta ja siten lisäämisen aikavaativuus on $O(\log n)$.

Poistaminen: Poistaminen tapahtuu ensin kuten binäärihakupuulle ja sen aikavaatimus on $O(h)$, missä h on korkeus ja koska AVL-puu on tasapainoinen, niin poistamisen aikavaatimus on $O(\log n)$. Koska poistamisen jälkeen suoritettavat kierrot ovat vakioaikaisia ja niitä suoritetaan korkeintaan kaksi kertaa ja pahimmassa tapauksessa kuljetaan poistettua solmusta juureen, niin tämän aikavaativuudeksi saadaan $O(\log n)$. Poistaminen koostuu siis kahdesta aikavaativuudeltaan $O(\log n)$ osasta ja siten poistamisen aikavaativuus on $O(\log n)$.

Etsiminen: Etsiminen suoritetaan rekursiivisella metodilla ja rekursio suoritetaan korkeintaan $\log n$ kertaa. Etsimisen aikavaativuudeksi saadaan $O(\log n)$.

Pienimmän ja suurimman alkion etsiminen: Pienin ja suurin alkio löytyy aina lehdestä ja metodissa oleva looppิ suoritetaan siis $\log n$ -kertaa. Aikavaativuudeski saadaan tällöin $O(\log n)$.

Suorituskyky

Toiminto	Binäärihaku- puu	AVL-puu	Punamustapuu	B-puu
Lisääminen	ms	ms	ms	ms
Poistaminen	ms	ms	ms	ms
Etsiminen	ms	ms	ms	ms
Pienimmän ja suurimman et- siminen	ms	ms	ms	ms