58131 Tietorakenteet

Erilliskoe 25.1.2008 kello 16–20, Exactum A111 kuulustelija: Jyrki Kivinen

Kirjoita oma nimesi ja kurssin nimi jokaiseen palauttamaasi paperiin. Kokeen maksimi on 60 pistettä.

1. [*20 pistettä*]

   (a) Esitä algoritmi, joka poistaa kahteen suuntaan linkitetystä järjestämättömästä tunnussolmullisesta listasta duplikaatit eli saman avaimen toistuvat esiintymät. Jos sama avain esiintyy useita kertoja, vain listassa ensimmäisenä oleva jätetään listaan ja muut poistetaan.

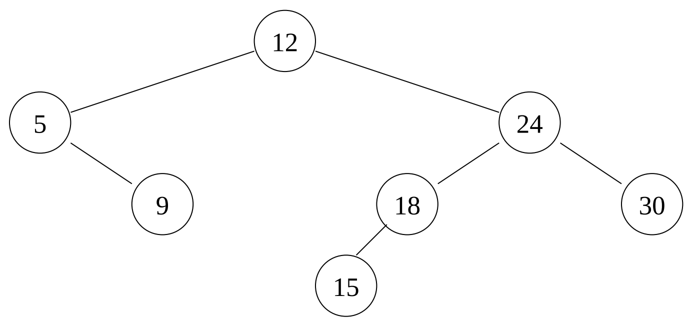
   Esitä ratkaisusi pseudokoodina (tai halutessasi Java-kielellä) kaikkine yksityiskohtineen, olettamatta ennalta annetuksi mitään kurssilla esitettyjä algoritmeja. Algoritmisi ei tarvitse olla erityisen tehokas.

   (b) Mikä on tehtävän (a) ratkaisusi aikavaativuuden suuruusluokka? Miten voisit ratkaista ongelman nopeammin käyttämällä apuna kurssilla esitettyjä tietorakenteita?

   Vastaukseksi riittää lyhyt sanallinen selitys nopeamman algoritmin periaatteista ja aikavaativuudesta. Pseudokoodeja tms. ei tarvitse esittää.

2. [*15 pistettä*]

   (a) Väritä allaolevan binäärihakupuun solmut niin, että punamustaehdot täyttyvät. (Kuvassa on siis tavalliseen tapaan jätetty NIL-solmut merkitsemättä.)



   (b) Anna jokin sellainen avain, että jos se lisätään paikalleen ylläolevassa puussa tekemättä tasapainotuksia tai muita puun rakenteen muutoksia, niin puuta ei enää voi värittää punamustaehtojen mukaisesti. Perustele, että avaimella on tämä ominaisuus.

   (c) Lisää (b)-kohdassa antamasi avain puuhun ja suorita sen jälkeen puussa yksi tai useampia kiertoja niin, että puu voidaan jälleen värittää punamustaehtojen mukaisesti.

   Kohdassa (c) sinun ei tarvitse simuloida mitään erityistä punamustan puun tasapainotusalgoritmia. Riittää, että esität jonkin sellaisen kierron (tai jonon kiertoja), että puu tulee tasapainoon, ja osoitat, että syntynyt puu todella voidaan värittää. Kohdan (b) avain kannattanee valita siten, että kohta (c) ei tule turhan vaikeaksi.

Käännä!

3. [*10 pistettä*] Oletetaan minimiprioriteetijono toteutetuksi kurssilla esitettynä kekorakenteena. Esitä yksityiskohtaisena pseudokoodina algoritmi, joka tulostaa kaikki rakenteessa olevat annettua arvoa x pienemmät avaimet. Rakenteen sisältöä ei saa muuttaa. Arvioi algoritmisi aikavaativuutta. Täysien pisteiden saamiseksi algoritmin tulee toimia ajassa $O(k)$, missä k on tulostettavien avainten lukumäärä.

4. [*15 pistettä*] On annettu n valuutan $c_1, \ldots, c_n$ väliset vaihtokurssit $r_{i,j}$, missä yhdellä yksiköllä valuuttaa c_i voi ostaa $r_{i,j}$ yksikköä valuuttaa c_j.

   Jos esim. $r_{2,3} = 2$, $r_{3,7} = 3$ ja $r_{7,2} = 0{,}2$, niin yhdellä yksiköllä valuuttaa c_2 voi ostaa 2 yksikköä valuuttaa c_3, näillä edelleen $2 \cdot 3 = 6$ yksikköä valuuttaa c_7 ja näillä edelleen $6 \cdot 0{,}2 = 1{,}2$ yksikköä valuuttaa c_2. Tässä siis valuuttaa c_2 voi monistaa pelkkiä valuuttakauppoja tekemällä.

   Esitä tehokas algoritmi, jolla voidaan havaita tällaiset tilanteet, joissa valuuttaa voidaan monistaa. Voit käyttää hyväksesi mitä tahansa kurssilla esitettyjä algoritmeja. *Vihje:* esittämällä tilanne sopivasti verkkona ongelma ratkeaa soveltamalla sopivaa algoritmia lähes sellaisenaan.