T-106.4100 Algoritmien suunnittelu ja analyysi, syksy 2005

Tentti, 6.2.2006

Kirjoita jokaisen palauttamasi paperin yläreunaan selvästi "T-106.4100 Algoritmien suunnittelu ja analyysi, 6.2.2006", nimesi, opiskelijanumerosi ja koulutusohjelmasi sekä palauttamiesi paperien kokonaismäärä.

1. a) (3p) Ratkaise seuraava rekursioyhtälö, kun n on kahden potenssi. Anna täsmällinen ratkaisu (suuruusluokka ei riitä).

$$T(n) = \begin{cases} 1, & \text{kun } n = 1 \\ 4T(n/2) + 2n & \text{kun } n > 1 \end{cases}$$

   b) (3p) Arvaa hyvä ratkaisu seuraavalle palautuskaavalle ja todista ratkaisusi oikeaksi induktiolla, kun n on kahden potenssi (c_1 ja c_2 ovat vakioita).

$$T(n) \leq \begin{cases} c_1, & \text{kun } n = 1 \\ T(n/2) + c_2 & \text{kun } n > 1 \end{cases}$$

2. a) (2 p) Miksi Union-Find-rakenteissa pyritään siihen, että yhtä joukkoa kuvaava puurakenne on mahdollisimman matala?

   b) (1 p) Miten tähän tavoitteeseen pyritään Find-operaation suorittamisen yhteydessä (mainitse vain yksi tapa)? Esitä kuvan avulla esimerkki tällaisesta Find-operaatiosta.

   c) (3 p) Selitä, miten minimin poisto tapahtuu toisaalta binomikeosta ja toisaalta Fibonacci-keosta. Mitkä ovat operaation pahimman tapauksen aikavaativuudet näissä rakenteissa? Perustele aikavaativuustulokset lyhyesti.

3. a) (2p) Selitä lyhyesti dynaamisen ohjelmoinnin periaate.

   b) (4p) Selitä lyhyesti, miten karsi ja etsi -menetelmään perustuva algoritmi, joka etsii taulukosta i:nneksi pienintä avainta, toimii. Mikä on algoritmin aikavaativuus (ei tarvitse perustella)? Miten algoritmin aikavaativuus muuttuisi, jos sarana-alkioksi (pivot) otettaisiin suoraan käsiteltävän alueen viimeinen alkio (perustele)?

4. (6p) On yleisesti tunnettua, että n-alkioisen taulukon järjestäminen pikajärjestämisalgoritmilla (quicksort) vaatii keskimäärin ajan $O(n \log n)$. Miten tämä tulos voidaan johtaa? Esitä periaatteet jostain tavasta johtaa tämä tulos. Vastauksessasi ei tarvitse olla yksityiskohtaista matemaattista johtoa, mutta periaatteet eri vaiheista on esitettävä. Anna myös joitain kaavoja esimerkiksi analyysin siitä osasta, josta lähdetään liikkeelle. Sopiva vastauksen pituus on 1–2 sivua.

5. (6p) Tämä tehtävä liittyy verkon maksimaalisen virtauksen laskemiseen.

   Olkoon annettu suunnattu verkko $G = (V, E)$ siten, että jokaiseen kaareen $(u, v) \in E$ on liitetty sen kapasiteetti $c(u, v)$ ja kaarta pitkin tällä hetkellä menevä virtaus $f(u, v)$. Olkoon edelleen jo laskettu verkon tämänhetkistä virtausta vastaava jäännösverkko (engl. residual network) $G(f)$. Kirjoita pseudokoodi algoritmille, joka laskee jäännösverkon avulla yhden lyhimmän täydennyspolun (engl. augmenting path) solmusta s solmuun t ja täydennyspolkua vastaavan täydennyksen. Täydennyspolun ei tarvitse olla paras mahdollinen, mutta sen on oltava mahdollisimman lyhyt.

   Algoritmi saa siis syötteenä jäännösverkon sekä solmut s ja t ja sen tulee tulostaa täydennyspolkuun kuuluvat kaaret sekä polkua vastaava täydennys.