T–106.4100 Algoritmien suunnittelu ja analyysi, syksy 2014

Tentti, 10.4.2015 **Ei apuvälineitä eikä laskimia**

Kirjoita jokaisen palauttamasi paperin yläreunaan selvästi "T-106.4100 Algoritmien suunnittelu ja analyysi, 10.4.2015", nimesi, opiskelijanumerosi ja koulutusohjelmasi sekä palauttamiesi paperien kokonaismäärä.

1. a) (3p) Mitkä seuraavista väittämistä pitävät paikkansa ja mitkä eivät? Perustele lyhyesti! (Lyhyt sanallinen perustelu riittää, tarkkoja matemaattisia todistuksia ei vaadita.)

   i. $2n^5 + 2n \in \Omega(1)$

   ii. $3n^6 + 7n \log n \in O(n^8)$

   iii. $2n + 16n \log \log n \in \Theta(n^3 \log n)$

   b) (3p) Verkon syvyyssuuntainen läpikäynti luokittelee verkon kaaret puukaariin, eteneviin kaariin, takautuviin kaariin ja poikittaiskaariin. Mitä nämä neljä tarkoittavat?

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

$$T(n) = \begin{cases} 1, & \text{kun } n = 1 \\ 2T(n/3) + 3n & \text{kun } n > 1 \end{cases}$$

   b) (3p) Ratkaise sopivalla muuttujanvaihdolla (n ottaa vain arvoja joilla T(n) pystytään ratkaisemaan).

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

3. a) (3p) Selitä, miten tasoitettu vaativuus (amortized complexity) ja keskimääräinen vaativuus (average case complexity) eroavat toisistaan.

   b) (3p) Esitä kaksi Fibonacci-keon operaatiota, joilla tasoitettu aikavaativuus on eri kuin pahimman tapauksen aikavaativuus. Anna näiden operaatioiden molemmat aikavaativuudet ja kerro, mihin nämä aikavaativuustulokset perustuvat (tarkkoja matemaattisia johtoja ei tarvitse esittää).

4. a) (2p) Selitä lyhyesti pyyhkäisyviiva-algoritmien (engl. sweeping) periaate.

   b) (2p) Selitä lyhyesti branch-and-bound -menetelmän periaate.

   c) (1p) Millaisten ongelmien ratkaisuihin branch-and-bound -menetelmää voi ja kannattaa käyttää?

   d) (1p) Mitä voidaan sanoa branch-and-bound -menetelmää käyttävän algoritmin aikavaativuudesta?

5. (6p.) Verkko $G = (V, E)$ on kaksijakoinen, jos verkon solmut voidaan jakaa kahteen joukkoon V_1 ja V_2 siten, että minkään kahden samaan joukkoon kuuluvan solmun välillä ei ole kaarta. (Esimerkiksi jos $u \in V_1$ ja $v \in V_1$, niin u:n ja v:n välillä ei ole kaarta verkossa G.) Kirjoita pseudokoodi algoritmille joka tutkii, onko annettu suuntaamaton verkko kaksijakoinen. Algoritmin aikavaativuuden tulee olla $O(|V| + |E|)$.

   Vinkki: Suuntaamaton verkko on kaksijakoinen jos ja vain jos verkon solmut voidaan värittää käyttämällä vain kahta eri väriä niin, että kaikilla kaarilla $(u, v) \in E$ solmulla u on eri väri kuin solmulla v.

Summakaavoja:

$$\begin{aligned}
\sum_{i=1}^{n} i &= \frac{n\,(n+1)}{2}\\
\sum_{i=1}^{n} i^2 &= \frac{n\,(n+1)\,(2\,n+1)}{6}\\
\sum_{i=0}^{n} 2^i &= 2^{n+1}-1\\
\sum_{i=0}^{n} a^i &= \frac{a^{n+1}-1}{a-1}\\
\sum_{i=0}^{n} i\,a^i &= \frac{n\,a^{n+2}-(n+1)\,a^{n+1}+a}{(a-1)^2}, \text{ jossa } a \neq 1
\end{aligned}$$

Logaritmi:

$x_1 > x_2 \Rightarrow \log x_1 > \log x_2$

$\log x_1 = \log x_2 \Rightarrow x_1 = x_2$

$\log 1 = 0$

$\log(x_1 x_2) = \log x_1 + \log x_2$

$\log x^b = b \log x$

$x_1^{\log x_2} = x_2^{\log x_1}$ $(\log x_2 \log x_1 = \log x_1 \log x_2)$