

58131 Tietorakenteet ja algoritmit (kevät 2017)

Kurssikoe 1 (6.3.2017)

Tentissä saa olla mukana käsin kirjoitettu yksi A4-kokoinen ”lunttilappu”, jonka molemmilla puolilla saa olla tekstiä.

Vastaa kuhunkin tehtävään erilliselle konseptipaperille. Kirjoita jokaisen paperin yläkulmaan **kurssin nimi, kokeen päivämäärä, nimi, nimikirjoitus ja opiskelijanumero**. Vaikka jättäisit johonkin tehtävään vastaamatta, palauta silti vastauspaperi kyseiseen tehtävään.

Tehtävissä, joissa pyydetään algoritmia, voit käyttää luentojen (Cormenin) tyyppistä pseudokoodia tai muita ymmärrettäviä pseudokoodityylejä tai oikeaa ohjelmointikieltä, esim. Javaa. Jos käytät oikeaa ohjelmointikieltä, selitä erityisen hyvin, mitä ohjelmassasi tapahtuu, äläkä käytä mitään kielen erikoista piirrettä tai valmiita kirjastoja.

Vastaa kaikkien kysymysten kaikkiin kohtiin. Kokeen maksimipistemäärä on 22.

1. [4 pistettä]

(a) Pitääkö paikkansa, että $3n^2 + 5n + 1 = \mathcal{O}(n^3)$?

(b) Pitääkö paikkansa, että $3n^2 + 5n + 1 = \Omega(n^3)$?

Perustele vastauksesi täsmällisesti \mathcal{O} - ja Ω -merkinnän määritelmästä lähtien.

2. [6 pistettä] Syötteenä on annettu kaksi n -alkioista kokonaislukutaulukkoa, A ja B . Tehtävänä on laskea, kuinka moni taulukon B alkioista esiintyy myös taulukossa A . Jos sama kokonaisluku esiintyy useita kertoja taulukossa B , jokainen esiintymä lasketaan mukaan. Useampikertaiset esiintymät taulukossa A eivät vaikuta asiaan.

Esim. jos $n = 10$ ja $A = [8, 2, 11, 5, 3, 7, 3, 9, 5, 11]$ ja $B = [6, 3, 8, 1, 8, 5, 4, 10, 8, 12]$, niin vastaus on 5, koska taulukon B alkioita 3, 8, 8, 5 ja 8 esiintyvät myös taulukossa A .

Esitä ongelmalle algoritmi, jonka pahimman tapauksen aikavaativuus on $\mathcal{O}(n \log n)$. Perustele, että algoritmillasi on tämä aikavaativuus. Mikä on algoritmisi tilavaativuus?

Algoritmille ei tarvitse esittää yksityiskohtaista pseudokoodia, kunhan esität ratkaisuperiaatteen selkeästi niin, että aikavaativuuden voi todeta. Ratkaisussa saa käyttää apuna mitä tahansa kurssilla esitettyjä algoritmeja ja niiden tunnettuja aikavaativuuksia jne., mutta mainitse aina selvästi, mitä algoritmia tai tulosta käytät.

Käännä!

3. [6 pistettä] Sovelluksessa tarvitaan dynaamista joukkoa, jonka avaimet ovat merkkijonoja. Oletetaan, että se toteutetaan hajautustauluna käyttäen jakolaskumenetelmää ja ylivuotolistoja.
- (a) Mitä seikkoja pitää ottaa huomioon talletusalueen koon valinnassa?
 - (b) Selitä yksityiskohtaisesti, miten SEARCH-operaatio toimii, ts. miten annettu merkkijono etsitään rakenteesta.
 - (c) Entä jos ylivuotolistojen sijaan käytettäisiinkin *kaksoishajautusta*: Miten tämä kenties vaikuttaisi talletusalueen koon valintaan? Miten SEARCH nyt toimisi? Mitä etuja ja haittoja tällä olisi ylivuotolistoihin verrattuna?
4. [6 pistettä]
- (a) Anna pseudokoodina algoritmi, joka saa syötteenä osoittimen binääripuun juureen ja palauttaa puun lehtien lukumäärän.
 - (b) Määrittele, millainen on *binäärihakupu*. Mitä ehtoja binäärihakupuun pitää täyttää, että se olisi *AVL-puu*?
 - (c) Anna esimerkki tilanteesta, jossa avaimen lisääminen AVL-puuhun rikkoo AVL-ehdon ja tilanne korjataan suorittamalla kaksoiskierto.