

## LUE NÄMÄ OHJEET HUOLELLISESTI!

Voit toimia kahdella eri tavalla. Merkitse vastauspapereihin kumman tavan valitsit. Epäselvät tapaukset tulkitaan vaihtoehdon 2 mukaan. Molemmissa vaihtoehdoissa aikaa on neljä tuntia (ennakkotiedoista poiketen).

1. Jos haluat suorittaa vain 2. välikokeen, tee silloin vain tehtävät 1–3.
2. Lopputentti: vastaa kaikkiin kysymyksiin (1–6). Valitse tämä vaihtoehto, jos olit poissa 1. välikokeesta, tai haluat “paikata” 1. välikokeen tulostasi. **Huom:** tehtävät 4–6 siis korvaavat 1. välikokeen tuloksen myös siinä tapauksessa että aiempi tuloksesi olisi ollut parempi!

1. [Yhteensä 15p] Vastaa *lyhyesti* seuraaviin kysymyksiin:
  - (a) Mitä etuja ja haittoja hajautuksella on verrattuna tasapainoisiin etsintäpuihin?
  - (b) Luettele prioriteettijonon tarjoamat operaatiot. Kerro myös niiden aikavaativuudet, kun prioriteettijono on toteutettu binäärikekona (eli kasana).
  - (c) Mitä tarkoitetaan verkon vahvasti yhtenäisillä komponenteilla ja komponenttiverkolla?
2. [Yhteensä 12p] Kirjoita kekojärjestämisalgoritmin pseudokoodi (8p). (Saat käyttää apuna tunnettuja kekooperaatioita, kirjoittamatta niiden pseudokooodeja.) Mikä on algoritmin aikavaativuus (2p) ja *aputilavaativuus* (2p)?
3. [Yhteensä 13p] Tarkastellaan rautatieverkkoa, jonka **solmuina** ovat rautatieasemat, **kaarina** ovat asemien väliset rataosuudet (oletetaan, että radat haarautuvat vain asemien kohdalla) ja **kaaripainona** on suurin junan paino, jonka kyseinen rataosuus kestää rikkoutumatta.
  - (a) Tarkastellaan kysymystä “Mitä reittiä pitkin  $t$  tonnia painava juna voi kulkea asemalta  $x$  asemalle  $y$ ?” Selitä *lyhyesti*, millainen verkkoalgoritmi (pseudokoodia ei tarvitse kirjoittaa) vastaa tähän kysymykseen (4p), ja ilmoita algoritmisi asymptoottinen aikavaativuus rautatieverkon esityksen koon suhteen (2p).
  - (b) Kuten edellisessä kohdassa, mutta kysymyksenä onkin nyt “Enintään kuinka painava juna voi kulkea asemalta  $x$  asemalle  $y$ ?” (5p+2p)
4. [Yhteensä 15p] Ovatko seuraavat väitteet totta? Vastaa “kyllä” tai “ei”, ja tapauksessa “ei” *perustele* yhdellä lauseella miksi ei.
  - (a) Väite: Punamustassa hakupuussa jokainen lehtisolmu on samalla syvyydellä. Totta vai ei? Jos ei, niin millä syvyydellä ne ovat?
  - (b) Väite: Jos hakupuun solmujen avaimet tulostetaan esijärjestyksessä, niin avaimet tulostuvat suuruusjärjestyksessä. Totta vai ei? Jos ei, niin onko olemassa jokin muu järjestys jolle väite pätee?
  - (c) Erään binääripuun solmujen avaimet ovat leveyssuuntaisessa järjestyksessä lueteltuna 3, 2, 5, 1. Saman puun avaimet jälkijärjestyksessä ovat 1, 2, 5, 3. Väite: voidaan päätellä, että puu on hakupuu. Totta vai ei? Jos ei, niin anna vastaesimerkki.
5. [Yhteensä 12p] Halutaan toteuttaa sellainen linkitetty tietorakenne *jono*, jolla on tavallisten jono-operaatioiden lisäksi vielä seuraava uusi operaatio: **invert**( $Q$ ) kääntää jonon  $Q$  päinvastaiseksi, eli sen ensimmäisestä alkioista tuleekin viimeinen, toisesta alkioista toiseksi viimeinen, ja niin edelleen. Selitä sanallisesti tällaisen kääntyvän jonon toteutus rakenne. Vanhojen operaatioiden enqueue ja dequeue asymptoottinen aikavaativuus ei saa kasvaa tämän uuden operaation invert lisäämisen seurauksena (8p). Täysiin pisteisiin vaaditaan lisäksi, että myös tämän uuden operaation asymptoottinen suoritus aika on sama kuin niillä (4p).
6. [Yhteensä 13p] On annettu jokin binääripuu, jossa on  $n$  solmua.
  - (a) Kirjoita algoritmi (pseudokoodi) joka käy puun läpi leveyssuuntaisessa järjestyksessä (5p). Algoritmisi tulee käyttää aputietorakenteena jonoa. Mikä on algoritmin aikavaativuus? (1p) Entä *aputilavaativuus*, jos puu on tasapainoinen? (1p) Aputilavaativuus, kun puun korkeus on  $\Theta(n)$ ? (1p)
  - (b) Jos algoritmissa muutetaan jono-operaatiot pino-operaatioiksi (tekemättä algoritmiin mitään muita muutoksia), niin millainen läpikäyntijärjestys puulle silloin saadaan? (3p) Mikä on muutetun algoritmin aputilavaativuus, kun puu on tasapainoinen? (1p) Aputilavaativuus, kun puun korkeus on  $\Theta(n)$ ? (1p)