

Kirjoita jokaiseen paperiin oma nimesi, oppilasnumerosi, tutkinto-ohjelmasi, kurssikoodi ja kurssin nimi, päivämäärä, sali, palauttamiesi paperien lukumäärä sekä *allekirjoituksesi*.

1) Kymmenen kysymystä (10 x 1p)

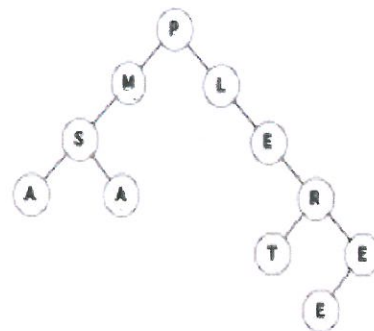
Tämä tehtävä on tentin pakollinen osa, josta on saatava vähintään 5p/10p, jotta loput tentistä tarkistetaan. Tämä tehtävä ei kuitenkaan yksistään riitä tentin läpäisyyn. Toisaalta viiteen pisteeseen ei edellytetä ”täysin oikeaa vastausta” vaan oleellista on, että pystyt osoittamaan ymmärtäneesi tehtävän koodin toiminnan. Käytä siis aikaa perustelujen miettimiseen ja esittämiseen. Viittaa perusteluissa ohjelmakoodin rivinumeroihin, jos mahdollista.

Alla on annettu kaksi binääripuun läpikäyntialgoritmia (`traverse1` ja `traverse2`). Lue ensin kaikki kysymyskohdat vastaamatta niihin ja sen jälkeen tutustu annettuihin koodinpätkiin erittäin huolella. Vastaa tämän jälkeen kaikkiin kysymyksiin ja käytä aikaa perustelujen pohtimiseen ja muotoilemiseen. Huomaa, että kaikissa kysymyksissä viitataan alla oleviin algoritmeihin ja, että vastaukset tulee perustella hyvin, tai siis *pisteet tulevat vain perusteluista!*

```
1 void traverse1(node t) {  
2   if (t != NULL) {  
3     visit(t);  
4     traverse1(t->left);  
5     traverse1(t->right);  
6   }  
7 }  
8  
9  
10
```

```
11 void traverse2(node root) {  
12   stack.push(root);  
13   while (stack.notEmpty()) {  
14     node next = stack.pop();  
15     visit(next);  
16     if (next->right != NULL)  
17       stack.push(next->right);  
18     if (next->left != NULL)  
19       stack.push(next->left);  
20   }  
21 }
```

- a) Selitä `traverse1`:n toiminta sanallisesti (ilman esimerkkiä). Huom! Pyri selittämään *miten* algoritmi ratkaisee ongelman. Älä selitä koodia rivi-riviltä.
- b) Selitä nyt `traverse2`:n toiminta sanallisesti. Miten se eroaa edellisestä?
- c) Anna esimerkki, kun `traverse1` vierailee kuvan 1 solmuissa lähtien juurisolmusta P. Vihje: Taulukoi mihin solmuihin muuttujat `t`, `t->left`, ja `t->right` viittaavat ohjelman suorituksen edetessä. Mikä on solmujen tulostusjärjestys, jos `visit(t)` tulostaa aina vieraillun solmun kirjaimen?
- d) Anna esimerkki, kun `traverse2` vierailee kuvan 1 solmuissa lähtien juurisolmusta P. Taulukoi tällä kertaa muuttujat `next`, `next->left` ja `next->right`. Mikä on solmujen tulostusjärjestys tässä tapauksessa?



Kuva1: Binääripuu

- e) *Analysoi* `traverse1`:n suoritus aika puussa olevien solmujen n funktiona.
- f) *Analysoi* `traverse2`:n suoritus aika puussa olevien solmujen n funktiona. Muista perustelut analyysissäsi (pelkkä lopputulos ei riitä pisteisiin)!
- g) *Peruste*le pitääkö väite paikkansa vai ei:
`traverse1` on tehokkaampi kuin `traverse2`.
- h) *Peruste*le pitääkö väite paikkansa vai ei:
`traverse1` suorittaa vähemmän vertailuja (*if*-lauseet) kuin `traverse2`.
- i) Mikä olisi `traverse1`:n tuottama puun läpikäyntijärjestys, jos rivit 3 ja 4 vaihtaisivat paikkaa? *Anna esimerkki*.
- j) Algoritmi `traverse2` käyttää *while*-silmukkaa. Voitaisiinko se korvata jollakin toisella silmukalla? Perustele joko miksi ei tai anna esimerkki miten (kirjoita algoritmi uusiksi).
Bonustehtävä:
- k) *Pohdi ja vertaile* algoritmien `traverse1` ja `traverse2` muistinkäyttöä.

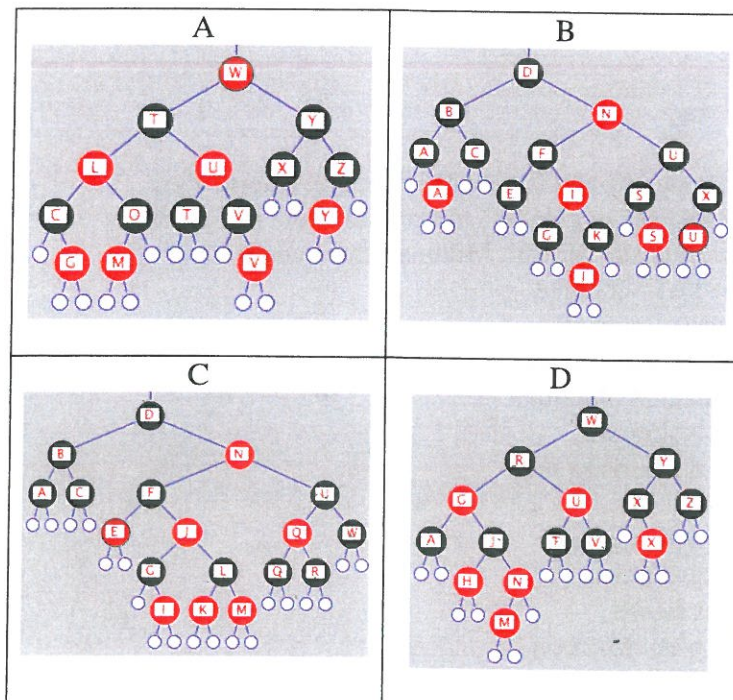
2) Terminologiaa (2p + 2p + 2p + 2p)

Määrittele seuraavat käsitteet (4 x 1p). Huom! *Anna* jokaisesta myös *esimerkki* (4 x 1p).

- Keko (heap)
- Hajautusfunktio (hash function)
- Neliöllinen kokeilu (quadratic probing)
- Valikointi-ongelma (selection-problem)

3) Punamusta puu (4p + 4p)

- Määrittele punamusta puu.
- Mitkä kuvien vaihtoehdoista ovat valideja punamustia puita ja mitä punamustan puun sääntöjä muut vaihtoehdot rikkovat?

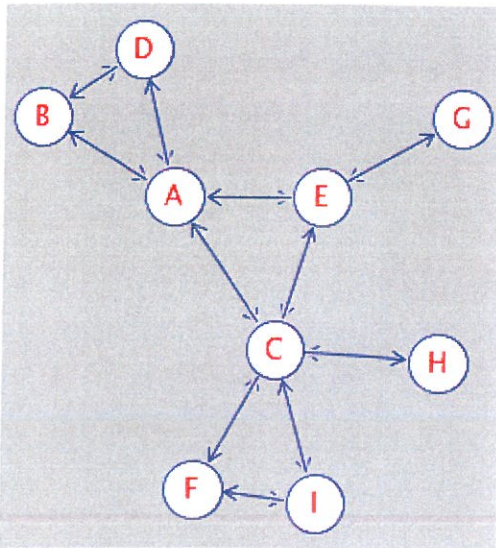


4) Järjestämismenetelmät (10p)

Kirjoite *essee* aiheesta *järjestämismenetelmät*. Lähde liikkeelle johdattelemalla lyhyesti lukijaa aiheeseen ja mm. määrittelemällä mitä tarkoitetaan järjestämisellä. Etene sen jälkeen kohti yksityiskohtia. Voit käsitellä aihetta laajasti vertailemalla erilaisia menetelmiä keskenään tai rajata aihettasi sopivasti esimerkiksi keskittymällä johonkin *tehokkaaseen* menetelmään ja kuvaamalla sen yksityiskohtaisesti ja esimerkkejä käyttäen. Esseeseen suosituspituus on 2 sivua.

5) Verkot (4p + 2p + 4p)

- Selitä sanallisesti* miten rekursiivinen syvyysuuntainen haku (DFS) toimii. Voit olettaa, että algoritmi saa syötteenä suuntaamattoman verkon, joka on kuvattu vieruslistaesityksenä. Vieruslistassa solmut ovat aakkosjärjestyksessä.
- Millainen olisi kuvan 2 verkon vieruslistaesitys em. tapauksessa?
- Miten rekursiivinen DFS etenee kuvan 2 verkossa, kun lähdetään liikkeelle solmusta A? Anna esimerkki siten, että siitä selviävät sekä rekursiokutsut (vierailujärjestys) että se missä järjestyksessä rekursiokutsut päättyvät.



Kuva 2: Suuntaamaton verkko.