

Kirjoita jokaiseen paperiin oma nimi, oppilasnumero, tutkinto-ohjelma, kurssikoodi ja kurssin nimi, päivämäärä, sali, palauttamiesi paperien lukumäärä sekä *allekirjoituksesi*. Numeroi palauttamasi paperit juoksevilla numeroinnilla. Tentissä ei saa käyttää mitään ylimääräisiä apuvälineitä.

1) Kymmenen kysymystä (10 x 1p)

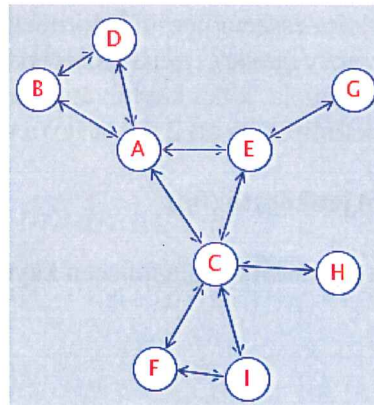
Tämä tehtävä on *tentin pakollinen osa*, josta on saatava vähintään 5p/10p, jotta loput tentistä tarkistetaan. Tämä tehtävä ei kuitenkaan yksistään riitä tentin läpäisyyn. Toisaalta viiteen pisteeseen ei edellytetä ”täysin oikeaa vastausta” vaan oleellista on, että pystyt osoittamaan ymmärtäneesi tehtävän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

Alla on annettu kaksi verkon läpikäyntialgoritmia (DFS1 ja DFS2). Lue ensin kaikki kysymyskohdat vastaamatta niihin ja sen jälkeen tutustu annettuihin koodinpätkiin erittäin huolella. Vastaa tämän jälkeen kaikkiin kysymyksiin. Huomaa, että kaikissa kysymyksissä viitataan alla oleviin algoritmeihin ja, että vastaukset tulee perustella hyvin, eli *pisteet tulevat vain perusteluista*

```
1 Algorithm DFS1(G, u)
2   visited[u] ← true
3   for each v ∈ Adj[u] do
4     if visited[v] == false then
5       DFS1(G, v)
6   finished[u] ← true
7
8
9
10
```

```
11 Algorithm DFS2(G, u)
12   Stack.push(u)
13   visited[u] ← true
14   while ( Stack not empty )
15     u = Stack.pop()
16     for each v ∈ Adj[u] do
17       if visited[v] == false then
18         visited[v] ← true
19         Stack.push(v)
20   finished[u] ← true
```

- Koodiriveillä 3 ja 16: $\text{each } v \in \text{Adj}[u]$ tarkoittaa kaikkia niitä solmuja v , jotka ovat solmun u naapureita. *Miten toteuttaisit tietorakenteen Adj?* Miksi?
- Anna esimerkki em. tietorakenteesta kuvan 1 verkon tapauksessa. Esitä tietorakenne siten, että siinä näkyy kaikkien solmujen vierussolmut ja niiden järjestys.
- Selitä* DFS1:n toiminta sanallisesti (ilman esimerkkiä). Huom! Pyri selittämään *miten* algoritmi toimii yleisesti. Älä selitä koodia rivi-riviltä.
- Selitä* nyt DFS2:n toiminta sanallisesti. Miten se eroaa edellisestä?
- Anna esimerkki, kun DFS1 vierailee kuvan 1 verkon solmuissa lähtien solmusta A. Ilmoita missä järjestyksessä solmu merkitään sekä *visited* että *finished* -lipuilla algoritmin suorituksen edetessä.
- Anna esimerkki, kun DFS2 vierailee kuvan 1 verkon solmuissa lähtien solmusta A. Ilmoita pinon *Stack* sisältö aina kun algoritmissa tullaan riville 14. Alleviivaa lisäksi solmu, joka poistetaan pinosta rivillä 15.
- Perustele* pitääkö väite paikkansa vai ei: DFS1 tuottaa saman läpikäyntijärjestyksen kuin DFS2.
- Perustele* pitääkö väite paikkansa vai ei: DFS1 on



Kuva 1: esimerkkiverkko.

- tehokkaampi kuin DFS2.
- Miten soveltaisit DFS1:stä etsiäksesi verkon yhtenäiset komponentit (*connected components*)?
 - Algoritmi DFS2 käyttää *pinoa* (stack) apurakenteenaan. Mikä olisi solmujen läpikäyntijärjestys, jos se korvattaisiin *jonolla*?
Bonustehtävä (+ 1p):
 - Pohdi ja vertaile* algoritmien DFS1 ja DFS2 muistinkäyttöä.

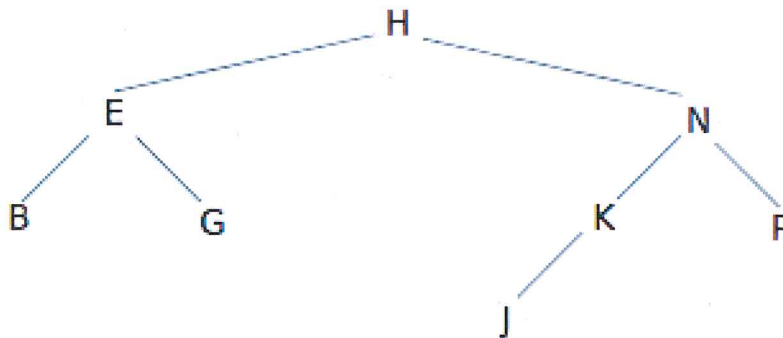
2) Terminologiaa (2p + 2p + 2p + 2p)

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

- Stabiili järjestämismenetelmä (*Stable sorting method*)
- Binäärihaku (Binary search)
- Hajautus (Hashing)
- Abstrakti tietotyyppi (ADT, Abstract Data Structure)

3) Binäärinen hakupuu (2p + 1p + 3p + 2p)

- Määrittele binäärinen hakupuu (BST, Binary Search Tree)
- Lisää alkio F kuvan 2 binääriseen hakupuuhun. Piirrä lopputulos.
- Selitä miten alkio voidaan poistaa binäärisestä hakupuusta (3 eri tapausta).
- Piirrä lopputulos, kun alkio H on poistettu kuvan 2 puusta.



Kuva 2: Binäärinen hakupuu.

4) Prioriteettijonot (8p)

Kirjoita essee aiheesta Prioriteettijonot. Vihjeitä: *Määrittele* käsite *prioriteettijono* (*priority queue*), *anna esimerkki jostain sen tehokkaasta toteutuksesta*, mainitse nimeltä algoritmeja, jotka käyttävät *prioriteettijonoa* apurakenteenaan, jne. Esseen maksimipituus on 2 sivua (joka toiselle riville).

5) Ajankäyttö (0p)

Arvioi tenttiin vastaamiseen käyttämäsi aika noin 15 minuutin tarkkuudella.