

Kirjoita jokaiseen paperiin oma nimesi, oppilasnumerosi, tutkinto-ohjelmasi, 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 + 1p = 11p)

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 algoritmia (fact_1 ja fact_2), jotka laskevat kertoma-funktion (*factorial*). 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 väittämät voi perustella yhtä hyvin vastaamalla kyllä tai ei, joten pisteet tulevat vain *perusteluista!*

```
1 int fact_1(int n)
2 {
3     if (n<2)
4         return 1;
5     else
6         return fact_1(n-1)*n;
7 }

8 int fact_2(int n)
9 {
10    int i, fact;
11    fact = 1;
12    for (i=1; i<=n; i++)
13        fact = fact*i;
14    return fact;
15 }
```

- Selitä algoritmin 1 (ts. fact_1) toiminta *esimerkin avulla*.
- Selitä algoritmin 2 (ts. fact_2) toiminta *esimerkin avulla*.
- Missä järjestyksessä ja kuinka monta kertolaskua fact_1 suorittaa? Anna esimerkki, kun algoritmia kutsutaan parametrilla n=3.
- Missä järjestyksessä ja kuinka monta kertolaskua fact_2 suorittaa? Anna esimerkki, kun algoritmia kutsutaan parametrilla n=3.
- Analysoi algoritmin 1 suoritus aika sen saaman syötteen koon n funktiona.
- Analysoi algoritmin 2 suoritus aika sen saaman syötteen koon n funktiona.
- Perustele pitääkö väite paikkansa vai ei: algoritmi 1 on tehokkaampi kuin algoritmi 2.
- Perustele pitääkö väite paikkansa vai ei: algoritmi 1 laskee saman funktion kuin algoritmi 2.
- Mikä olisi algoritmin 1 kertolaskujen suoritusjärjestys, jos riviä 6 muutettaisiin muotoon "return n*fact_1(n-1);"? Anna esimerkki.
- Algoritmi 2 käyttää for-silmukkaa. Voitaisiinko se korvata jollakin toisella silmukalla? Perustele joko miksi ei tai anna esimerkki miten (kirjoita algoritmi uusiksi).

Bonustehtävä:

- Pohdi ja vertaile algoritmien 1 ja 2 muistinkäyttöä.

2) Terminologiaa (4 x (1p + 1p) = 8p)

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

- a) *Abstrakti tietotyyppi (ADT, Abstract Data Type)*
- b) *Prioriteettijono (Priority Queue)*
- c) *Binäärikeko (Binary Heap)*
- d) *Kekoehto (Heap Property)*

3) Lomitusjärjestäminen (2p + 2p + 2p + 2p + 2p = 10p)

- a) *Selitä sanallisesti miten lomitus (merging) toimii. Havainnollista selitystäsi käyttämällä sopivia piirroksia.*
- b) *Soveltuuko lomitus taulukoille (arrays) paremmin kuin linkitetyille listoille (linked lists)? Perustele vastauksesi (vihje: kysymykseen ei ole yhtä oikeaa kyllä/ei-vastausta, pyri hakemaan vähintään kaksi erilaista näkökulmaa, pisteet tulevat vain perusteluista).*
- c) *Selitä sanallisesti miten lomitusjärjestäminen (mergesort) toimii. Havainnollista selitystäsi käyttämällä sopivia piirroksia.*
- d) *Analysoi lomitusjärjestämisen aikavaatimus pahimmassa tapauksessa em. selityksen perusteella.*
- e) *Tarkastellaan listaa, jossa on seuraavat alkiot: 3, 1, 6, 2, 5, 7, 4, 8, 9, 0. Esitä välivaiheittain miten järjestäminen etenee lomitusjärjestämisessä.*

4) Hakurakenteet (2p + 2p + 2p + 4p = 10p)

- a) *Määrittele käsite hakurakenne (search structure). Anna esimerkkejä hakurakenteista.*
- b) *Selitä tasapainotettujen hakupuiden (balanced search trees) peruseriaatteet.*
- c) *Selitä hajautusmenetelmien (hashing) peruseriaatteet.*
- d) *Analysoi hakurakenteiden keskeisten operaatioiden aikakompleksisuus tasapainotettujen hakupuiden ja hajautusmenetelmien osalta. Vertaile tasapainotettuja hakupuita ja hajautusmenetelmiä keskenään eri operaatioiden keskimääräisen sekä pahimman tapauksen aikavaatimusten perusteella.*

5) Ajankäyttö (0p)

Arvioi tenttiin vastaamiseen käyttämäsi aika noin 15 minuutin tarkkuudella.

6) Palaute (2p)

Anna palautetta kurssin Noppa-sivulta löytyvän lomakkeen kautta 14.12.2014 mennessä.