

Mukana saa olla kynä, kumi ja viivain. Max. 60p

Lue tarkkaan, mitä tehtävässä kysytään/vaaditaan.

Vastaa kaikkiin tehtäviin. Muista perustelut, pelkkä vastaus on noin 1-2 pisteen arvoinen.

1. Lukujärjestelmämuunnoksia (yht. 8p)

- Muunna luku  $12.8_{10}$  binääriseksi (6 desimaalibitin tarkkuudella)? (2p)
- Esitä luku  $27_{10}$  heksadesimaalijärjestelmän lukuna (2p)
- Muunna luvut  $69_{10}$  ja  $-39_{10}$  binäärisiksi kahden komplementtia hyödyntäen ja laske tämän jälkeen niiden summa. Tarkasta vielä, että binäärilukujen yhteenlasku on tuottanut **oikean** lopputuloksen ( $30_{10}$ ). (4p)

2. Määrittele seuraavat termit esim. pienten esimerkkien avulla: (yht. 6p)

- Ylivuoto(2p)
- Luonnollinen kieli(2p)
- Vuokaavio (flow diagram)(2p)

3. Algoritmit ja ohjelmointi (yht. 8p)

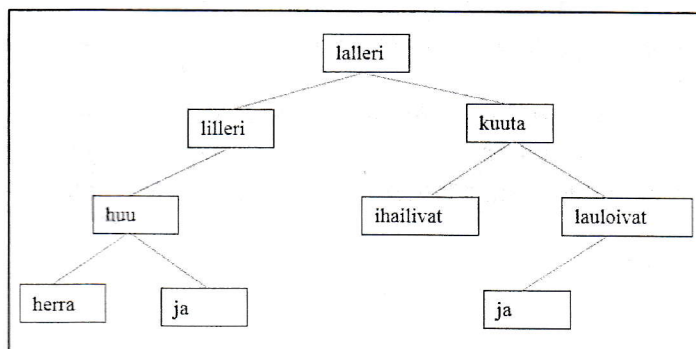
- Keksi jokin pieni algoritmi ja esitä se kahdella eri abstraktiotasolla (4p)
- Keksi algoritmi, joka on päättymätön eli jää niin sanottuun ikuisen silmukkaan. (4p)

4. Onko mahdollista piirtää alla olevan vaatimuksen mukainen puu? Jos mahdollista, piirrä. Jos ei, perustele miksei. (yht. 8p)

- 4-arinen (4-haarainen) puu, jossa on 5 lehteä(2p)
- Täydellinen binääripuu, jossa on 5 lehteä (2p)
- Järjestetty binääripuu, jonka korkeus on 1. (2p)
- 1-haarainen puu, jonka korkeus on 2 (2p)

5. Luettele alla olevan puun alkioit (yht.9p)

- Esijärjestyksessä (3p)
- Välijärjestyksessä (3p)
- Jälkijärjestyksessä (3p)



6. Laadi algoritmi, joka laskee alla olevan lukutaulukon (nimeltään "Lukutaulukko") alkioden keskiarvon. Taulukon indeksointi alkakoon nolasta. (5p)

|   |   |   |    |   |   |   |   |
|---|---|---|----|---|---|---|---|
| 2 | 5 | 6 | 2  | 5 | 9 | 1 | 0 |
| 6 | 8 | 3 | 3  | 7 | 1 | 1 | 4 |
| 3 | 5 | 8 | 82 | 2 | 1 | 2 | 3 |

7. Kerro välivaiheineen (esim. piirtämällä + sanallisesti) mitä alla oleva *Lajittele*-algoritmi tekee listalle  $L=(6,2,9,9,5,4,7,3,6,8,8,8)$  (5p)

```
procedure lajittele (L)
  ListaPuuksi (L, p)
  PuuListaksi (p, L)
```

```
procedure ListaPuuksi (L, p)
  p= tyhjä puu
  while (L != tyhjä)do
    (LisääPuuhun(L.arvo, p)
    L=L.seuraava)
```

```
procedure LisääPuuhun (alkio a, p)
  if (p=tyhjä) then
    (p.arvo=a
    p.vasen=tyhjä
    p.oikea=tyhjä)
  else if (a < p.arvo) then
    (LisääPuuhun(a, p.vasen))
  else (LisääPuuhun(a, p.oikea))
```

```
procedure PuuListaksi (p,L)
  if (p != tyhjä) then
    (PuuListaksi (p.vasen, L)
    Lisää juuren arvo listan L loppuun
    PuuListaksi (p.oikea, L))
```

8. Oikein vai väärin? (Oikea vastaus 1p, väärä vastaus -0,5p, vastaamatta jättäminen 0p). (Max 4p, min 0p)
- While-silmukka on tietorakenne
  - Laskettaessa kolme ykkös-bittiä yhteen, vastaus on  $11_2$ . ( $1+1+1=11_2$ )
  - Jos 32-bittisen liukulukuesityksen eksponentti on 00000010, niin se tarkoittaa, että alkuperäisessä kiinteän luvun esityksessä pilkkua oli siirretty kaksi askelta vasemmalle.
  - Loogisen operaation: **1 AND 0** lopputulos on 0 (eli FALSE)
9. Tiedot-taulukko sisältää 60:n potilaan tiedot (tiedot esitetty tietueena). Tietueen nimi on Potilas ja sen tarkempi tietorakenne on esitetty alla:

```
Potilas:RECORD(
  nimi:string
  laakari:string
  laakitys:string
  vastaanottoaika: date
  sairaus:string
)
```

Lisäksi on olemassa Raportti-taulukko, joka on 2x60 -merkkijonotaulukko (string-tyyppinen taulukko), joka on alussa tyhjä.

**Kysymys: Mitä seuraava algoritmi tekee?** Älä "suomenna" rivejä (esim. "rivillä 3 muuttuja i asetetaan ykköseksi") vaan vastaukseksi riittää mikä algoritmin perimmäinen idea on.  
(Sovitaan, että taulukoiden indeksointi lähtee 1:stä)? (7p)

```
k=0
j=1
i=1
while i<=60 do
  eka=Tiedot[i]
  toka= Tiedot[60-k]
  a=toka.nimi
  b= eka.laakitys
  Raportti[1,j]=a
  Raportti[2,j]=b
  i=i+1
  j=j+1
  k=k+1

print(Raportti)
```

