

## OHI-1110 Laaja ohjelmointi 1. Tenti 10.12.2009.

Kirjoita jokaiseen vastauspaperiin opiskelijanumero ja nimi. Vastaa tehtävä 1 ja 2 paperiin 1 ja tehtävät 3 ja 4 paperiin 2. Tentissä ei saa olla mukana laskinta eikä muutakaan materiaalia. Vastatessasi kirjoita mahdollisimman selkeällä käsialalla.

### Tehtävä 1 (paperiin 1)

Seuraavassa on kolme ohjelmaa, jotka kaikki testaavat, onko parametri palindromi (eli sama etu- ja takaperin luettuna, esimerkiksi "otto", "anna"). *Korjaa* ohjelmien mahdolliset virheet, mutta *älä* tehosta algoritmeja, vaikka siihen olisi tilaisuus. Jos ohjelma on kunnossa, vastaa OK tms.

a. (2p)

```
bool pali (string kohde)
{
    for (unsigned int i = 0; i < kohde.length(); ++i) {
        if (kohde[i] != kohde[kohde.length() - i]) {
            return false;
        }
    }
}
```

b. (2p)

```
bool apupali (string& kohde, unsigned int paikka, unsigned int loppu)
{
    if (kohde[i] != kohde[loppu - 1]) {
        return false;
    }
    return apupali (kohde, paikka + 1, loppu - 1);
}

bool pali (string kohde)
{
    return apupali(kohde, 0, kohde.length());
}
```

c. (2p)

```
bool pali (string kohde)
{
    bool lippu = true;
    for (unsigned int i = kohde.length() ; i > kohde.length() / 2; --i) {
        if (kohde[i - 1] != kohde[kohde.length() - i]) {
            lippu = not lippu;
            break;
        }
    }
    return lippu;
}
```

## Tehtävä 2 (paperiin 1)

- a. Oletetaan, että

```
int luvut [ KOKO ]; // KOKO > 0
```

ja että luvut-taulukkoon on luettu jotkin arvot.

Tutustu seuraavaan ohjelmaan:

```
int luku1 = luvut[0];
int luku2 = luku1;
for (int i = 1; i < KOKO; ++i) {
    if (luvut[i] < luku1) {
        luku1 = luvut[i];
    }
    if (luvut[i] > luku2) {
        luku2 = luvut[i];
    }
}
cout << luku1 << ' ' << luku2 << endl;
```

Muuttujien nimien pitäisi viitata niiden käyttötarkoitukseen. Miten nimeäisit muuttujat luku1 ja luku2 uudelleen? (1p)

- b. Jos a-kohdan ohjelman jälkeen on lause

```
if ((luku1 <= luvut[KOKO - 1]) and (luvut[KOKO - 1] <= luku2)) {
    cout << "Tosi" << endl;
} else {
    cout << "Epätosi" << endl;
}
```

tulostaako tämä lause aina toden, aina epätoden vai voiko sen tulostus vaihdella? (1p)

- c. Jos a-kohdan for-lauseessa ++i vaihdetaan muotoon i += 2, tulostaako b-kohdan lause aina toden, aina epätoden vai voiko sen tulostus vaihdella? (1p)

- d. Jos b- tai c-kohdassa tulostus vaihtelee, mikä vaikuttaa vaihteluun? (1p)

- e. Kuinka monta suuruusvertailua (<, >) taulukosta luvut peräisin olevien arvojen välillä a-kohdan ohjelma yhteensä laskee? (1p)

- f. Mitä kohtia a-kohdan ohjelmassa pitää muuttaa, jotta se laskee samat arvot kuin ennenkin, mutta käyt taulukon luvut läpi lopusta alkuun? Anna koodin muuttuneet osat. (1p)

- g. Miten vaikuttaa a-kohdan tulokseen, jos silmukan sisällä olevat ehtolauseet kirjoitettaisiin alla olevaan muotoon? (1p)

```
if (luvut[i] < luku1) {
    luku1 = luvut[i];
} else if (luvut[i] > luku2) {
    luku2 = luvut[i];
};
```

## Tehtävä 3 (paperiin 2)

- a. Mitä ovat Tuttu ja kaveri seuraavissa määrittelyissä? (2p)

```
struct Tuttu { int a; double b; };  
Tuttu kaveri;
```

- b. Edellisen lisäksi on määritelty

```
Tuttu toverit[ KOKO ]; // KOKO > 0
```

Mitä vikaa seuraavissa kahdessa ohjelmarivissä on? (2p)

```
toverit.a = 42;  
cout << kaveri << endl;
```

- c. Tarvitaan ohjelma, joka tallettaa tietoja suunnistuskilpailuista. Jokaisesta juoksijasta pitää kilpailijan numeron avulla löytää seuraavat tiedot: nimi, suoritus aika ja tarkistuksesta tullut tieto, onko kilpailija leimannut radan rastit oikein. Minkälaisia C++-kielen rakenteita käyttäisit tietojen tallentamiseen? (4p)
- d. Kirjoita aliohjelma, joka etsii ja tulostaa tekemästäsi tietorakenteesta kilpailijan nimen perusteella, mikä hänen suoritus aikansa on. Aika tulostetaan vain siinä tapauksessa, että rastit on leimattu oikein, muuten tulostetaan "hylätty". (3p)

## Tehtävä 4 (paperiin 2)

Pascalin kolmiossa on jokaisella rivillä alaspäin mennessä yksi luku enemmän kuin edellisellä. Jokainen luku on sen yläpuolella olevien kahden luvun summa (olematon luku katsotaan nol-laksi). Alla viisi ensimmäistä riviä:

```
1  
1 1  
1 2 1  
1 3 3 1  
1 4 6 4 1
```

Kunkin luvun saa laskettua kaavasta  $\frac{n!}{(n-k)!k!}$ , missä n on rivi (alkaen nol-lasta) ja k luvun numero vasemmalta (alkaen nol-lasta).

Tehtävänäsi on tehdä ohjelma, joka kysyy käyttäjältä, montako riviä tulokseen halutaan, ja tulostaa sitten annetun määrän rivejä Pascalin kolmiosta. Älä huolehdi rivien sisennyksestä (kolmion reuna voi siis olla vasemmalta suora).

- a. Millaisiin funktioihin jakaisit ohjelman? Miksi juuri näin? (4p)
- b. Toteuta kukin suunnittelemasi funktio ja itse pääohjelma. (2p)



