

Ohjelmoinnin perusteet, erilliskoe 3.6.2014

Kirjoita jokaiseen konseptiin kurssin nimi, kokeen päivämäärä, nimi ja opiskelijanumero. Vastaukset palautetaan tehtäväkohtaisiin pinoihin.

Voit lyhentää komennon `System.out.println` kirjoittamalla `sout` ja käyttää muitakin lyhenteitä, mikäli selität lyhenteen jokaisessa paperissa, jossa sitä käytät. Voit olettaa kaikissa tehtävissä, että ohjelman runko on seuraava:

```
import java.util.Scanner;

public class Ohjelma {
    public static void main(String[] args) {
        Scanner lukija = new Scanner(System.in);
        // int luku = Integer.parseInt( lukija.nextLine() );
        // String rivi = lukija.nextLine();
    }
}
```

1. (6p) Vastaa jokaiseen kohtaan lyhyesti ja ytimekkäästi, anna jokaisesta kohdasta konkreettinen esimerkki. Koko tehtävän vastauksen pituudeksi riittää noin kaksi sivua.

- (a) Mikä on muuttuja? Mitä tarkoitetaan muuttujan tyyppillä? Miten alkeis- ja viittaustyyppiset muuttujat eroavat toisistaan? Anna muutama esimerkki alkeis- ja viittaustyyppisistä muuttujista. (2p)
- (b) Mikä on metodi? Miten ja mihin metodeja käytetään? (2p)
- (c) Mitä ovat olio ja luokka? Mikä on olion ja luokan välinen suhde? (2p)

2. Ohjelmoinnin perusasiat (6p)

- (a) (3p) Tee ohjelma, joka tulostaa toistolausetta (esim. while tai for) käyttäen kaikki 4:lla jaolliset kokonaisluvut aloittaen luvusta 4 ja päätyen lukuun 800. Tulostuksen tulee tapahtua siten, että jokaiselle riville tulostetaan 4 lukua, tämän jälkeen tulostus alkaa seuraavalta riviltä. Luvut tulee tulostaa sarakkeittain siten, että kunkin sarakkeen lukujen viimeisen numeron tulee sijaita luvun koosta siippumatta samalla kohtaa.

Ohjelman tulostuksen tulee siis näyttää **täsmälleen** seuraavalta:

```
4   8   12  16
20  24  28  32
36  40  44  48
52  56  60  64
(paljon rivejä välissä)
772 776 780 784
788 792 796 800
```

- (b) (3p) Tee ohjelma, joka lukee käyttäjän syöttämiä kokonaislukuja kunnes käyttäjä syöttää luvun 0. Luettuaan luvut, ohjelma ilmoittaa negatiivisten ja positiivisten lukujen lukumäärän sekä sen mikä oli suurimman ja pienimmän syötetyn luvun väli.

Esimerkki:

```
Syötä lukuja, 0 lopettaa:
5
3
-7
```

```
4
-4
0
positiivisia: 3
negatiivisia: 2
suurimman ja pienimmän väli 12
```

Esimerkissä pienin luku on -7 ja suurin 5 eli lukujen väli on 12.

Lukua 0 ei lasketa pienimmäksi eikä suuremmaksi, eli seuraavassa esimerkki tilanteessa jossa syöttössä on ainoastaan negatiivisia lukuja:

```
Syötä lukuja, 0 lopettaa:
-5
-7
-2
-9
0
positiivisia: 0
negatiivisia: 4
suurimman ja pienimmän väli 7
```

Pienin syöte on nyt -9 ja suurin -2 eli niiden väli on 7.

3. Metodeja (6 p)

- (a) (3p) Tee metodi `public static void tulostaVali(int reuna1, int reuna2)` joka tulostaa jokaisen parametriensa määrittelemällä välillä olevan kokonaisluvun.

Jos ensimmäinen parametri on toista pienempi, tulostuvat välin luvut *nousevassa järjestyksessä*. Eli esim. jos kutsutaan `tulostaVali(3, 7)` tulostuu

```
3 4 5 6 7
```

Jos ensimmäinen parametri on toista suurempi, tulostuvat välin luvut *laskevassa järjestyksessä*. Eli jos kutsutaan `tulostaVali(10, 8)` tulostuu

```
10 9 8
```

- (b) (3p) Tee metodi `public static int montakoKertaaEsiintyy(int[] luvut, int luku)` joka palauttaa tiedon, montako kertaa paremetrinä annettu luku esiintyy parametrinä annetussa taulukossa. Mikäli lukua ei taulukosta löydy, palauttaa metodi 0.

Esim. Metodi taulukon `[1,2,3,4,5,5]` ja luvun 5 parametrina saadessaan palauttaa 2. Tee myös pääohjelma, joka demonstroi metodin käyttöä.

Huom: jos et osaa käyttää taulukkoja, voit tehdä metodista version `public static int montakoKertaaEsiintyy(ArrayList<Integer> luvut, int luku)` jossa se siis saa parametrikseen luvut sisältävän ArrayList:in.

4. (12p) Tehtävässä tehdään oluiden ja panimoiden tietoja käsittelevä sovellus.

- (a) (5p) Tee ensin luokka `Olut`. Oluella tulee olla oliomuuttujat jotka tallentavat oluen nimen, oluttyylin ja alkoholiprosentin sekä tiedon siitä onko olut tuotannossa. Luokan tulee toimia seuraavasti:

```
public static void main(String[] args) {
    Olut olut = new Olut("Karhu", "Lager", 4.7);
    System.out.println(olut);
    olut.poistaTuotannosta();
    System.out.println(olut);
}
```

Tulostuu:

```
Karhu (Lager) 4.7% tuotannossa: on
Karhu (Lager) 4.7% tuotannossa: ei
```

Tee siis luokalle

- konstruktori, jolla asetetaan oluen nimi, tyyli ja alkoholiprosentti. Jokainen olut on luontihetkellä tuotannossa.
- metodi `toString()`, joka palauttaa oluen tekstiesityksen (yo. esimerkin tapaan muotoiltuna)
- metodi `poistaTuotannosta()`, jonka kutsumisen jälkeen olut ei ole enää tuotannossa.

Edellisten lisäksi oluella tulee olla metodit

- `onTuotannossa()`, joka palauttaa `true` jos olut on tuotannossa ja muuten `false`
- `getTyyli()`, joka palauttaa oluen tyylin
- `getNimi()`, joka palauttaa oluen nimen

- (b) (7p) Ohjelmoi luokka `Panimo`.

Panimolla on nimi (joka asetetaan konstruktorissa) ja panimoon liittyy joukko `Olut`-olioita, jotka panimo tallettaa `ArrayList`-tyyppiseen oliomuuttujaan. Tee luokalle seuraavat metodit:

- `public void uusiOlut(String nimi, String tyyli, double prosentti)`, luo panimolle parametrien tiedot omaavan oluen.
- `public void tiedot()` tulostaa panimon tiedot allaolevan esimerkin mukaisesti muotoiltuna.
- `public void poistaTuotannosta(String oluenNimi)`, poistaa parametrina määritellyn nimisen oluen tuotannosta.

Seuraava pääohjelma havainnollistaa panimon käyttöä:

```
public static void main(String[] args) {
    Panimo panimo = new Panimo("Mikkeller");
    panimo.uusiOlut("American Dream", "Pils", 4.2);
    panimo.uusiOlut("Dark Horizon", "Stout", 15);
    panimo.uusiOlut("Tjekket Pils", "Pils", 4.7);

    panimo.tiedot();

    panimo.uusiOlut("Santa's Little Helper", "Porter", 9);
    panimo.poistaTuotannosta("Dark Horizon");

    System.out.println("");
    panimo.tiedot();
}
```

Ohjelman tulostus oikein toteutetuilla luokilla on:

Mikkeller

Oluita yhteensä 3, joista tuotannossa 3

Valmistetut tyylit: Pils Stout

Oluet:

American Dream (Pils) 4.2% tuotannossa: on

Dark Horizon (Stout) 15.0% tuotannossa: on

Tjekket Pils (Pils) 4.7% tuotannossa: on

Mikkeller

Oluita yhteensä 4, joista tuotannossa 3

Valmistetut tyylit: Pils Stout Porter

Oluet:

American Dream (Pils) 4.2% tuotannossa: on

Dark Horizon (Stout) 15.0% tuotannossa: ei

Tjekket Pils (Pils) 4.7% tuotannossa: on

Santa's Little Helper (Porter) 9.0% tuotannossa: on

ArrayListin käyttöohje

- Konstruktori `public ArrayList<T>()` luo uuden ArrayList-olion, jossa listan alkiot ovat tyyppiä T.
- `public boolean add(T x)` lisää listan loppuun olion x.
- `public int get(int i)` palauttaa listan alkion kohdasta i.