

R0027 – Ohjelmoinnin perusteet Kehittämisharjoitukset

Taulukoita ja toistolauseita

TEHTÄVÄT

1. Taulukon luominen ja läpikäyminen

Määrittele uusi luokka ja sen main-metodiin seuraavat taulukot:

- kokonaislukutaulukko (int), jossa on 5 alkioita; alusta alkiot arvoilla 5, 7, 32, 31, 8
- desimaalilukutaulukko (double), jossa on 3 alkioita; alusta alkiot arvoilla 12.4 -13.55 ja 67.44
- merkkijonotaulukko (String), jossa on 4 alkioita; alusta alkiot arvoilla Helsinki, Lissabon, New York, Shanghai

- a) Tulosta ruudulle lause ”Alkiossa numero 3 on : ” ja liitä perään merkkijonotaulukon kolmas alkio
- b) Tulosta ruudulle lause ”Alkiossa numero 5 on : ” ja liitä perään kokonaislukutaulukon viides alkio
- c) Tulosta ruudulle lause ”Desimaalitaulukossa on XX arvoa”, jossa liität XX:n perään ko. taulukon pituuden kertovan kentän nimen

- d) Kirjoita kokonaislukutaulukkoa varten for-lause, jolla tulostat taulukon sisällön ruudulle
- e) Kirjoita desimaalilukutaulukkoa varten for-lause, jolla tulostat taulukon viimeisestä alkioista ensimmäiseen (eli käänteisessä järjestyksessä)
- f) Kirjoita merkkijonotaulukkoa varten do-while – silmukka, jolla käyt läpi kaikki alkiot ja asetat niiden arvoksi tekstin ”Tyhjä”.

2. Arvosanat taulukossa

Tee ohjelma, joka kysyy viiden harjoitustehtävän arvosanat peräkkäin ja tallentaa ne int – tyyppiseen kokonaislukutaulukkoon (Vihje: tässä kannattaa käyttää for – tai while – silmukkaa).

Kun arvosanat on luettu, ohjelma tulostaa ne ruudulle järjestettynä pienimmästä suurimpaan, sekä laskee niiden alle taulukon lukujen keskiarvon. Käytä keskiarvon laskemisessa apuna taulukon pituuden kertovan kentän arvoa (taulukko.length).

Taulukon saat järjestettyä käyttämällä Javan valmista metodia: Arrays.sort(taulukon_nimi). Jotta saat ko. metodin käyttöösi, joudut tuomaan sen sisältämän paketin import-lauseella ohjelman alussa:

```
import java.util.Arrays;
```

3. Etsimistä taulukosta (peräkkäishaku)

Tee ohjelma, joka etsii taulukosta erilaisia tietoja. Alusta ohjelman alussa kaksi taulukkoa:

```
int[] lukuTaulu = {55, 23, 6456, 324, 21, 234, 72, 21}  
String[] nimiTaulu = {Petteri, Kauko, Matti, Minna, Maisa, Juuso, Mauno}
```

Kysy tämän jälkeen käyttäjältä ”Mitä haluat etsiä? ”. Lue käyttäjän antama vastaus talteen.

Käy tämän jälkeen for- silmukalla taulukkoa läpi, ja vertaa jokaisen alkion kohdalla onko taulukossa kyseisellä silmukan laskurin arvon mukaiseen indeksiin tallennettu alkio sama kuin käyttäjän syöttämä arvo. Mikäli sama arvo löytyy, tulosta ruudulle "Luku löytyi taulukon alkioista numero: XX", jossa liität XX:n paikalle taulukon indeksin arvon (eli silmukan laskurin arvon).

Tee sama hakualgoritmi molemmille erityyppisille taulukoille. Huomaa, että String-tilukossa alkioita ei voi vertailla yhtäsuuruusoperaattorin avulla, vaan joudut käyttämään jo viime harjoituksissa tutuksi tullutta metodia equals tai equalsIgnoreCase.

```
Esim.  
String teksti = "Mika";  
String toinenTeksti = "Minna";  
  
if ( teksti.equals(toinenTeksti) ) {  
    System.out.println("Samat tekstit!");  
}
```

4. Indeksien tehokäyttöä: Arvosanoja

Kirjoita ohjelma, joka pyytää käyttäjältä luokan opiskelijoiden (21 kpl) arvosanat ja tulostaa arvosanajakauman käyttäen tähtiä seuraavasti:

```
0: **  
1: ***  
2: **  
3: *****  
4: **  
5: *****
```

Yhteensä: 21 opiskelijaa.

Vinkki: Tässä taulukon indeksejä kannattaa käyttää arvosanojen osoittimina, ja laskea arvosanojen lukumääriä taulukon alkioihin. Eli esim. jos syötetty arvosana on 3, kasvatetaan taulukon arvoa paikassa 3 yhdellä.

5. Suurimman ja pienimmän etsiminen, taulukon alustaminen satunnaisluvuilla

Kirjoita ohjelma, joka alustaa 100 alkion mittaiseen kokonaislukutaulukon käyttäen satunnaislukuja väliltä 0-1000.

Tämän jälkeen tee for-silmukka, jolla käyt läpi taulukkoa ja etsit alkioista suurimman arvon. Tulosta lopuksi suurin arvo.

Tee tämän jälkeen toinen for – silmukka, jolla etsit taulukon pienimmän arvon. Tulosta se lopuksi.

*Vinkkejä: Satunnaisen desimaaliluvun väliltä 0-0.99 saat arvottua käyttäen javan Math.random() – metodia. Kokonaisluvuksi saat muutettua sen tyyppimuunnoksella, eli (int) Math.random(). Luvut väliltä 0-1000 saat puolestaan kertomalla satunnaisluvun haluamallasi asteikolla, eli: (int) Math.random()*1000.*

Löytääksesi pienimmän tai suurimman arvon, valitse taulukon 1. alkio ja käytä sitä vertailuarvona (esim. pienimpänä alkiona). Vertaa sitten jokaisella silmukan kierroksella luettua alkioita tähän vertailuarvoon. Jos taulukosta löytyy pienempi luku kuin vertailuarvo, ota se talteen vertailuarvoksi. Näin vertailuarvoon on taulukon läpikäynnin jälkeen päätynyt pienin arvo. Sama logiikka toimii myös suurimmalle arvolle.

6. Moniulotteisia taulukoita: laivanupotus

Tee kaksiulotteinen `char[][]` -tyyppinen taulukko, jonka koko on 5x5 ruutua. Sijoita taulukkoon 5 kpl '*' -merkkejä kuvaamaan laivoja.

Tee ohjelmaan silmukka, joka kyselee käyttäjältä mihin koordinaatteihin ammutaan, ja katsoo onko samassa kohtaa ruudukossa '*' -merkki. Ohjelma antaa käyttäjälle palautteen "Osuma", mikäli ammus osuu. Muuten annetaan palaute "Ohi meni! Syötä uudet koordinaatit."

Kun saat sovelluksen toimimaan, voit laajentaa ruudukkoa ja sijoittaa sinne isompia laivoja.