

## Ohjelmistojen mallintaminen, syksy 2009, laskuharjoitus 5

1. Seuraavassa on lueteltu käsitteitä, jotka ovat jossain suhteessa keskenään. Mitä suhdetyyppejä (normaali yhteys, kompositio, kooste, perintä, riippuvuus, ...) kussakin tapauksessa kannattaa käyttää (siis jokaiseen palloon oma suhde). Piirrä esim. pieni luokkakaavio jokaisesta tilanteesta.

- Ruokaileva filosofi käyttää haarukkaa
- Unixissa tiedosto on joko tavallinen tiedosto tai hakemisto
- Tiedosto muodostuu tavuista
- Monikulmion kulmat ovat järjestetty jono pisteitä
- Opiskelija hallitsee Java-kielen
- Ihmisellä on syntymävuosi
- Hiiri ja näppäimistö ovat I/O-laitteita (= tietokoneen syötelaite)
- Polku yhdistää kaksi kylää

2. Laajennetaan edellisellä viikolla aloitettua monopolin luokkamallia. Ota pohjaksi viime viikon ratkaisusi tai esimerkkivastaus.

Seuraavassa asioita, jotka pitäisi tulla mallissa esille.

Ruutuja on useampaa eri tyyppiä:

- aloitusruutu
- vankila
- sattuma ja yhteismaa
- asemat ja laitokset
- normaalit kadut (joihin liittyy nimi)

Monopolipelin täytyy tuntea sekä aloitusruudun että vankilan sijainti.

Jokaiseen ruutuun liittyy jokin toiminto.

Sattuma- ja yhteismaaruutuihin liittyy kortteja, joihin kuhunkin liittyy joku toiminto.

Toimintoja on useanlaisia. Ei ole vielä tarvetta tarkentaa toiminnon laatua.

Normaaleille kaduille voi rakentaa korkeintaan 4 taloa tai yhden hotellin. Kadun voi omistaa joku pelaajista. Pelaajilla on rahaa.

Tässä vaiheessa ei vielä liitetä peliin metodeja. Niiden aika tulee seuraavalla viikolla.

3. Täydennetään LAL-järjestelmän luokkakaavio niiltä osin, mikä rajattiin pois edellisistä laskareista.

Voit ottaa lähtökohdaksesi edellisen viikon ratkaisusi tai esimerkkivastauksen.

4. Ohjelmoinnin jatkokurssin loppupuolella tutustutaan sangen hyödylliseen luokkaan ArrayList. ArrayList-luokan olioden voi ajatella olevan automaattisesti itseään kasvattava vaihtelevanmittainen taulukoja, joihin voidaan tallettaa muita olioita. Lyhyesti ilmaistuna ArrayList-oliot ovat "oliosäiliöitä".

Etsi Java-API:sta (<http://java.sun.com/j2se/1.5.0/docs/api/>), ArrayList-luokka. Tutkitaan ArrayListin ominaisuuksia. Tällä kertaa ei kiinnitetä huomiota ArrayListin käyttöön vaan sen sijaintiin Java-API:n luokkahierarkiassa. Eli mikä on ArrayList:in yläluokka, mitä sisäluokkia (eli yläluokan muita aliluokkia) sillä on, mikä on ArrayListin yläluokan yläluokka, mitä serkkuluokkia ArrayListilla on, jne. . .

Tutki myös mitä "sopimuksia" eli rajapintoja ArrayList mainitsee toteuttavansa? Mitä noissa sopimuksissa mainitaan?

Piirrä tilanteesta luokkakaavio. Jos kaavio kasvaa liian isoksi, älä tunge kaikkea kuvaan. Metodinimiä ei kannattane luokkakaavioon ainakaan montaa merkitä.

ArrayList, sen ylliluokka, sisarluokat, setä- ja tätiluokat sekä serkkuluokat liittyvät läheisesti kurssin Tietorakenteet aihepiiriin. Ammattimaisessa Java-ohjelmoinnissa tämä erilaisten "oliosäiliöiden" luokkaperhe on erittäin hyödyllinen.

5. ArrayListin lieene yksi parhaista tavoista toteuttaa olion yhteys moneen olioon.

Kurssille voi osallistua monta opiskelijaa. Kurssilla on metodi lisääOsallistuja(), jonka avulla opiskelija liittyy kurssille. Kurssilla on myös metodi tulostaOsallistujat(), joka toimii niinkuin olettaa saattaa.

Täydennä luokan Kurssi toteutus:

```
import java.util.*;

public class Kurssi{
    private ArrayList<Opiskelija> osallistujat;

    public Kurssi(){ ... }

    public void lisääOsallistuja( Opiskelija o){ ... }

    public void tulostaOsallistujat(){ ... }
}
```

Kurssin opiskelija voi toteuttaa esim. seuraavasti:

```
public class Opiskelija{
    private String nimi;

    public Opiskelija(String n){ nimi=n };

    public void tulosta(){ System.out.print( nimi ); }
}
```

Tee pääohjelma millä testaat ohjelman toimintaa.

Ohjeita löytyy esim. seuraavasta:

<http://www.cs.helsinki.fi/u/wikla/Ohjelmointi/Sisalto/5/ObjListMap.html>