ISE 313 – WEB-Design & Management Tools: Homework

Due on Thursday, March 26, 2015

Asst. Prof. Dr. Tolga Ovatman
R.A. Atakan Aral

Spring 2015

Contents

I. Scenario

You will develop a simple Java application that manages the public transportation network of Istanbul. It will support stations and lines between them. Transportation network is considered as a graph where vertices (nodes) represent stations and edges represent the lines.

II. Requirements

(a) Entities

There are sea (ferry and seabus), land (bus and metrobus) and rail lines (metro and train/tram). Stations can be in 4 types: sea, land, rail or interchange stations that support all 3 types. Intermediate stations are not considered. Only start and end stations are required.

(b) Features

Line features include the following:

Print Details Prints all the information about the line in a user friendly way.

Set Stations Updates the stations of the line.

Disable Disables the line temporarily.

Station features include the following:

Set Max Lines Sets the maximum number of lines each station can support. Same value is effective for all stations in the network.

Print Lines Prints the information about all the lines stopping at the station sorted alphabetically by line type.

Connections Prints the lines connecting the stations. If there is a direct connection, only one line is printed. If line transfer is necessary, all lines are printed. You may assume that there is a single connection between each station (no alternates).

(c) Test Code

Object-oriented principles and practices that are introduced during the lesson should be taken into consideration while writing your code. You should decide which classes, fields and methods need to be implemented. However, attached tester class should compile without errors and work as expected.

```java
package Homework;

public class Tester {
    public static void main(String[] args){
        Station.setMaxLines(3);

        Station S1 = new Station("Kadikoy");    //Interchange station by default
        Station S2 = new Station("Besiktas", "Sea");
        Station S3 = new Station("Taksim");
        Station S4 = new Station("Gebze", "Rail");
        Station S5 = new Station("Sariyer", "Land");
        Station S6 = new Station("ITU Ayazaga", "Interchange");
        Station S7 = new Station("Kabatas");
        Station S8 = new Station("Mecidiyekoy");
        Station S9 = new Station("Sisli");

        FerryLine L1 = new FerryLine("Line1", S1, S2);
        SeabusLine L2 = new SeabusLine("Line2", S7, S1);
        BusLine L3 = new BusLine("Line3", S8, S3);
        BusLine L4 = new BusLine("Line4", S7, S5);
        MetrobusLine L5 = new MetrobusLine("Line5", S1, S8);
        MetroLine L6 = new MetroLine("Line6", S3, S8);
        TrainLine L7 = new TrainLine("Line7", S1, S4);  //Error: max lines reached for S1

        L1.PrintDetails();  //Prints all the information about the line in a user friendly way
        S3.PrintLines();    //Prints the information about all the lines stopping at the station
                            //sorted alphabetically by line type.

        L6.setStations(S3, S6); //Updates the stations
        L6.setStations(S3, S5); //Error: Sariyer is a land station. No changes are made.
        L6.PrintDetails();

        BusLine L8 = new BusLine("Line8", S6, S2);  //Error: Besiktas is a sea station.
        L8.setStations(S9, S6);
        L8.PrintDetails();

        L3.disable();           //Disables a line temporarily
        S1.connectionTo(S5);    //Prints the lines connecting the stations (L2, L4 for this case)
        L2.disable();
        S2.connectionTo(S5);    //No connection found
    }
}
```

III. Report

A detailed report is critical for this homework. It should include the following sections.

(a) Algorithms

Explain in detail how your code detects maximum line limits, check station types, sort lines, disable lines and calculates connections.

(b) Design

Answer the following questions:

- What are the classes in your design? What is the purpose of each class?
- Which data structures did you used? Give reasons.
- Where and how did you use object oriented principles such as encapsulation, inheritance and polymorphism? Give reasons and code snippets.

IV. Grading

This homework constitutes the 10% of your grade. It will be graded over 10 points according to the following criteria:

- Code [6 points]
  - Requirements [4 points]
  - Object oriented principles [2 points]
- Report [4 points]

If anything in this document is unclear, please contact teaching assistant: aralat@itu.edu.tr