CS320 Programming Languages
Assignment 1
Due February 5th, 2021 at 11:59 PM

Objectives and Skills:

In this Assignment you will use Java regex API for data extraction.

This assignment will provide the opportunity to:

1. Learn how to use Java regex API for extracting data.
2. Write a structured OO application using Java.
3. Practice different Data Structures.

Tasks:

- Use ***Java regex API*** to extract the bus route stops schedule for a particular bus route number in a given schedule at **Community Transit.**
- Without changing anything in the given Interface IRouteFinder.java, ***follow the design*** *requirements to achieve* ***the application Requirements*** (Check below Design requirements section)
- Create the ***RouteFinder. java*** class that ***implements*** the ***IRouteFinder Interface***
- Create the Client.java class that has only ***a main method***. The Client class is to be used to test your application behavior and ***it should not include any Business logic.***

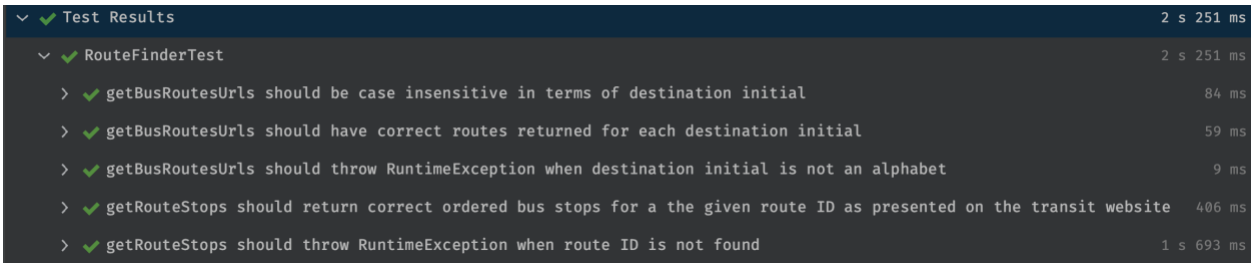
What to Submit:

1 zip file that contains at the following. java files

a. The IRouteFinder.java that was given to you
b. The RouteFinder.java that implements the IRouteFinder interface
c. Any helper method that supports your implementation

How will it be evaluated?

I will use automatic test cases to grade and evaluate your application. In order to get full grade your application has to pass all my test cases. Here are some of the test cases that I will use. You can use them as a guideline to test your own application. ***Please note if you didn't follow the Design Requirements by using the provided interface that means you didn't meet the requirements***. Also, please use appropriate code documentation for your project.



Here is a sample run for the application

Please enter a letter that your destinations start with **b**

Destination: Brier
Bus Number: 111
++++++++++++++++++++++++++++++++++
Destination: Bellevue
Bus Number: 532/535
++++++++++++++++++++++++++++++++++
Destination: Bothell
Bus Number: 105
Bus Number: 106
Bus Number: 120
Bus Number: 435
Bus Number: 532/535
Bus Number: Swift Green Line
++++++++++++++++++++++++++++++++++
Please enter your destination: **Brier**
Please enter a route ID: **111**

Destination: To Mountlake Terrace
Stop number: 1 is Brier Rd & 228th SW
Stop number: 2 is 228th SW & 48th W
Stop number: 3 is Mountlake Terrace Transit Center
Destination: To Brier
Stop number: 3 is Mountlake Terrace Transit Center Bay 2
Stop number: 2 is 228th SW & 48th W
Stop number: 1 is 228th SW & 29th W

Do you want to check different destination? Please type Y to continue or press any other key to exit

Sample run Description:

The Java application that provides the user with the Bus route stops schedule for a particular bus route number in a given schedule at Community Transit. You are supposed to use Java regex API for extracting data.

Here are the requirements for the application:

1. The program allows user to enter the initial for destination (e.g., B)
2. The program retrieves the busses schedules from Community Transit website at https://www.communitytransit.org/busservice/schedules/
3. The program lists the possible stops and route numbers for the chosen initial (e.g., Bellevue, Bothell, Brier)
4. The program allows user to enter the destination and the route number
5. The program prints the bus stops schedule for this particular route number in the available destinations for the ***weekdays*** schedule only and not the ***weekends***.

Design requirements:

Your application should follow the design in Figure 1 for the implementation. The IRouteFinder Interface has been given to you. You need to create the RouteFinder class that implements IRouteFinder Interface and the Client.java class for the application main method.

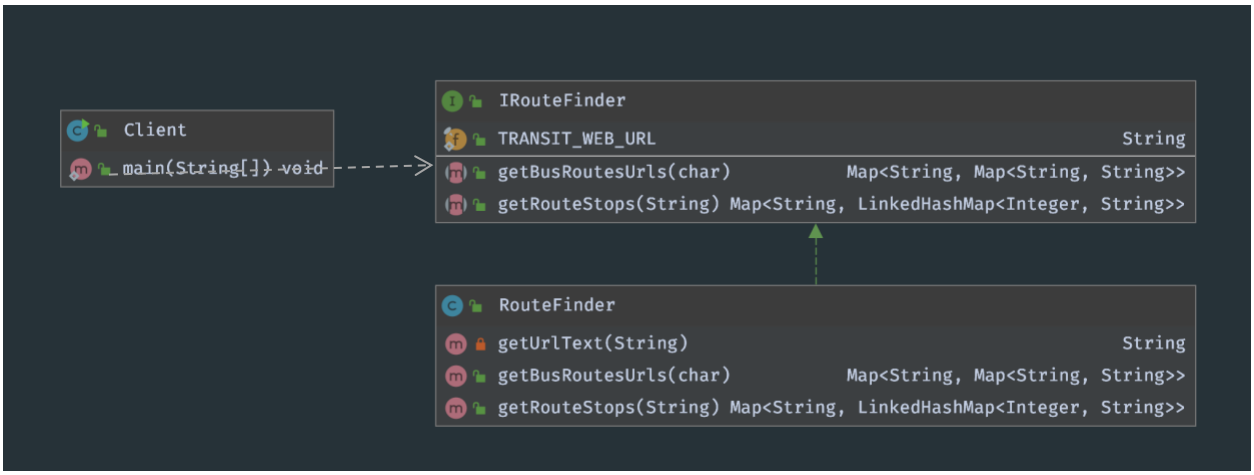


Figure 1: Bus Route Application Class Diagram

The IRouteInFinder Interface has two main methods:

1. getBusRoutesUrls
2. getRouteStops

Each method is responsible for part of the implementation steps in order to return the Bus route stops for a given RouteId for a particular destination

The description, the set of the parameters, and the return for each of the interface methods are as follows:

```java
import java.util.*;


public interface IRouteFinder {
   String TRANSIT_WEB_URL = "https://www.communitytransit.org/busservice/schedules/";

   /**
    * The function returns the route URLs for a specific destination initial using the URL text
    * @param destInitial This represents a destination (e.g. b/B is initial for Bellevue, Bothell, ...)
    * @return key/value map of the routes with key is destination and
    *       value is an inner map with a pair of route ID and the route page URL
    *       (e.g. of a map element <Brier, <111,
https://www.communitytransit.org/busservice/schedules/route/111>>)
    */
   Map<String, Map<String, String>> getBusRoutesUrls(final char destInitial);

   /**
    * The function returns route stops, grouped by destination To/From, for a certain route ID url
    * @param url: the URL of the route that you want to get its bus stops
    * @return map of the stops grouped by destination with key is the destination (e.g. To Bellevue)
    *  and value is the list of stops in the same order that it was parsed on
    * (e.g. of a map element <To Mountlake Terrace, <<1, Brier Rd &amp;amp; 228th Pl SW>, <2, 228th St SW
&amp;amp; 48th Ave W>, ...>>)
    */
   Map<String, LinkedHashMap<Integer, String>> getRouteStops(final String url);

}
```

You can download the interface from the Assignment view on Canvas.