CS320 Programming Languages
Assignment 1

Objectives and Skills:

In this Assignment you will use Java regex API for data extraction to build a user application.

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.***

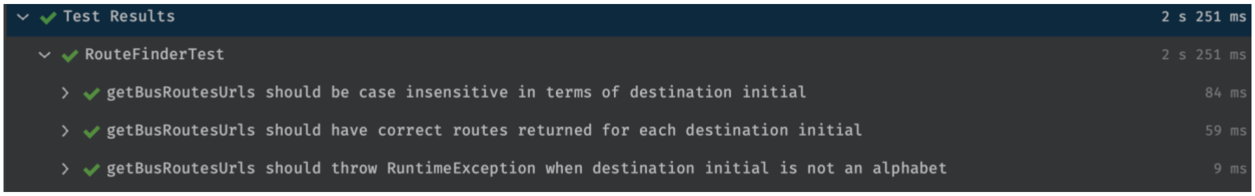
What to Submit:

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 methods, or classes 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. Below are some samples of the test cases that I will use, but not all. 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.



```
✓ Test Results                                                                                   2 s 251 ms
  ✓ RouteFinderTest                                                                              2 s 251 ms
    > ✓ getBusRoutesUrls should be case insensitive in terms of destination initial                     84 ms
    > ✓ getBusRoutesUrls should have correct routes returned for each destination initial               59 ms
    > ✓ getBusRoutesUrls should throw RuntimeException when destination initial is not an alphabet       9 ms
```

Here is a sample run for the application to show how it works

```
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: Bellevue

😍 Bus Trips Lengths in Minutes are:
{532/535 - Bellevue Transit Center= [68, 68, 68, 67, 63, 67, 68, 64, 64, 57, 60, 54, 62, 54, 62, 62, 63, 63,
60, 54, 54, 55, 55, 48, 54, 54, 53, 52, 54, 57, 57, 44, 59, 58, 62, 56, 63, 57, 55, 49, 54, 49, 49],
532/535 - Lynnwood and Everett= [80, 77, 80, 70, 67, 80, 80, 68, 58, 58, 66, 57, 79, 69, 79, 58, 67, 57, 58,
53, 53, 53, 62, 53, 54, 53, 68, 54, 55, 54, 71, 51, 54, 53, 60, 53, 57, 54, 53, 52, 52, 51, 51]}

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

Sample run Description:

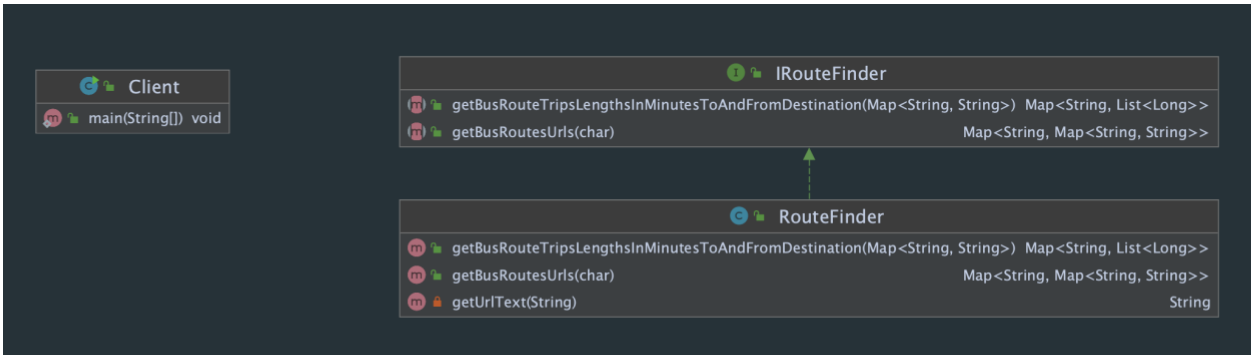
The Java application that provides the user with the Bus route trips length for a particular bus route number in a given schedule at Community Transit. You are supposed to use Java regex API for extracting data.

Here are the requirements for the application:

1. The Application allows user to enter an initial for a destination (e.g., B)
2. The Application retrieves the busses schedules from Community Transit website at https://www.communitytransit.org/busservice/schedules/
3. The Application list the possible stops and route numbers for the chosen initial (e.g., Bellevue, Bothell, Brier)
4. The Application allows user to enter a target destination out of the listed ones.
5. The Application can print the bus Trips Lengths in Minutes for the destination chosen by the user 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.



Client
main(String[]) void
IRouteFinder
getBusRouteTripsLengthsInMinutesToAndFromDestination(Map<String, String>) Map<String, List<Long>>
getBusRoutesUrls(char) Map<String, Map<String, String>>
RouteFinder
getBusRouteTripsLengthsInMinutesToAndFromDestination(Map<String, String>) Map<String, List<Long>>
getBusRoutesUrls(char) Map<String, Map<String, String>>
getUrlText(String) String

Figure 1: Bus Route Application recommended design

The IRouteInFinder Interface has two main methods:

1. getBusRoutesUrls
2. getBusRouteTripsLengthsInMinutesToAndFromDestination

Each method is responsible for part of the implementation steps to return the desire final output

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

```
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 list of trip lengths in minutes, grouped by
bus route and destination To/From
     * @param destinationBusesMap: key/value map of the routes with key is
bus route ID and
     *  value is the route page URL
     *  (e.g. of a map element <111,
https://www.communitytransit.org/busservice/schedules/route/111>>)
     * @return key/value map of the trips lengths in minutes with key is
the route ID - destination (e.g. To Bellevue)
     *         and value is the trips lengths in minutes
     *         (e.g. of a map element <111 - To Brier, [60, 50, 40, ...]>)
     */
    Map<String, List<Long>>
getBusRouteTripsLengthsInMinutesToAndFromDestination(final Map<String,
String> destinationBusesMap);

}
```

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