Department of Computing

COMP1127 – Introduction to Computing

Youba

Youba Ltd. is a startup company that aims to revolutionize the local public transportation by introducing a brand new way for travellers to connect with drivers. Users only need to have a cellphone to utilize the service. Given your knowledge of ADTs, Youba Ltd. has contracted you to implement the platform for their service.

1. Develop an ADT to represent a **Driver**:
    a. Implement a constructor having the following interface:

```
driver_make : firstName, lastName, carMakeAndModel -> Driver
```

The Driver ADT should be structured as follows:

```
('Driver', [firstName, lastName, carMakeAndModel, numberOfTripsCompleted])
```

Note: The number of trips completed is to be initialized to zero.

    b. Accessors:

```
i.   driver_getFirstName                : Driver -> string
ii.  driver_getLastName                 : Driver -> string
iii. driver_getCarMakeAndModel          : Driver -> string
iv.  driver_getNumberOfTripsCompleted   : Driver -> int
```

    c. Mutator:

```
i.   driver_increaseTripsCompleted      : Driver -> void
```

The number of trips is to be increased by 1

    d. Predicate:

```
i.   driver_isNewDriver                 : Driver -> Boolean
```

A driver is new if they have not yet completed any trips

2. To ensure fairness, the first driver that is available at a location is to be given preference to trips from that location. To enforce this, develop an ADT called **AvailabilityQueue**.

    a. Implement a constructor having the following interface:

    ```
    availabilityQueue_make : LocationName -> AvailabilityQueue
    ```

    The AvailabilityQueue ADT should be structured as follows:

    ```
    ('AvailablityQueue', LocationName, [])
    ```

    b. Accessors:
        i. `availabilityQueue_getLocationName : AvailabilityQueue -> string`
        ii. `availabilityQueue_front : AvailabilityQueue -> Driver`

    c. Mutator:
        i. `availabilityQueue_enqueue : AvailabilityQueue, Driver-> void`
        ii. `availabilityQueue_dequeue : AvailabilityQueue -> void`

    d. Predicate:
        i. `availabilityQueue_isEmpty : AvailabilityQueue -> Boolean`

3. Instantiate 4 **AvailabilityQueues** to represent "UWI", "Papine", "Liguanea" and "Half-Way-Tree" and store them in their respective variables named:

    a. availabilityQueue_UWI
    b. availabilityQueue_Papine
    c. availabilityQueue_Liguanea
    d. availabilityQueue_HalfWayTree

4. To improve customer experience, Youba gives 10% discounts to travellers if their last attempt to use the service failed due to unavailability of drivers in their area. The program uses the global variable called `knownPassengers` to track such failed attempts. Create a function having the following interface:

    ```
    calculateDiscount : PassengerTelephoneNumber -> float
    ```

    `knownPassengers` is a dictionary that is indexed by keys representing telephone numbers (in integer format, not strings). The value associated with the key signifies the number of failed attempts to use the service. In the example below, passenger having telephone number 4444444 has had 1 failed attempt. She is therefore entitled to a 10 % discount. Therefore, the function should return 0.10.

    ```
    Example:
    knownPassengers                = {4444444: 1}
    ```

5. Create a function to calculate fare having the following interface:

```
calculateFare : StartLocation, EndLocation, PassengerTelephoneNumber -> float
```

   The fare should be returned, less any discount that is applicable.

6. Create a function to move the taxi from one location's availability queue to the destination's availability queue. The function should have the following interface:

```
moveTaxi : startLocation, endLocation -> void
```

   Note that the first available driver is to be selected for the trip. The driver's completed trips is also to be incremented.

7. Create a function to request a taxi having the following interface:

```
requestTaxi : PassengerTelephoneNumber, PassengerLocation, PassengerDestination -> void
```

   If the start and end locations are the same then an appropriate error message is to be displayed and no further action is to be taken.

   Provided that the locations are correct, the fare should be displayed and the passenger asked to enter "Y" to confirm the trip or "N" to cancel.

   After the user confirms the trip, you should determine if there is a driver available. If so, execute the trip. The driver should now be reflected in the destination location and their trip count should have increased. Reset any failed trip attempts the passenger had to zero. If not, record the failed attempt for this passenger by adding 1 and display an appropriate error message.

8. Write a main driver function having the following interface:

```
youba : void -> void
```

   The passenger is asked to enter "Y" to request a taxi or "N" to end use of the service for that period. The facility accepts a number of taxi requests until the passenger chooses to end use of the service.

   The function makes use of `requestTaxi`, `moveTaxi` and all other defined functions to ensure that the Youba service runs efficiently and each passenger is informed of the calculated fare.

   When the service is ended for the day, the system displays a listing of drivers and the number of completed trips for the day, as well as a listing of the various locations and the driver at each location who is due to get the next trip.

You are expected to use the Python 3.6 (or higher) environment.

N.B. throughout this assignment, no abstractions are to be violated. If you do so, marks will be deducted.

- This assignment is worth 15% of your coursework mark
- You are to work in pairs
- Ensure both team member's ID numbers are included in the submitted code
- Submit well documented, original code using the container on OurVLE
- Name your file by concatenating the last four (4) digits of each member's ID# and prefix the concatenated digits with the characters"A1127". For example: A1127_1234_5678_9012.py
- The deadline for submission is midnight Sunday, December 2, 2018
- Late assignments are not encouraged. However, these are accepted and graded then 10% deducted for each day of late submission

Have fun!

APPENDIX

This code must be incorporated into your own. Insert your own ID numbers

```
"""
Group Information:

Member 1: IDNUMBER
Member 2: IDNUMBER

"""

# Global Variables
availabilityQueue_UWI           = None
availabilityQueue_Papine        = None
availabilityQueue_Liguanea      = None
availabilityQueue_HalfWayTree   = None

knownPassengers                 = {4444444: 1}
```