

Emergency Roadside Assistance Service

Implement a service that helps Geico customers with a disabled vehicle, to get immediate roadside assistance, by connecting them directly with emergency assistance service trucks that are available and operating at nearby locations.

Functionality

- Update location of a service provider
- Return the 5 nearest service trucks ordered by ascending distance
- Reserve a service provider for a customer
- Release a service provider from a customer

Additional Considerations

- Please include a readme file with instructions on building and executing tests
- Implementation mechanisms like durable storage, a REST API, GUI are not strictly needed
- Make assumptions/compromises to scope the problem to your available time and please list them in readme
- Submit your work via a github URL. We request that you **make it private after it has been evaluated**.

Implement the following interface. Design of Customer, Assistant and Geolocation classes is entirely up to you.

```
public interface RoadsideAssistanceService {
```

```
    /**
     * This method is used to update the location of the roadside assistance service provider.
     *
     * @param assistant represents the roadside assistance service provider
     * @param assistantLocation represents the location of the roadside assistant
     */
    void updateAssistantLocation(Assistant assistant, Geolocation assistantLocation);

    /**
     * This method returns a collection of roadside assistants ordered by their distance from the input geo location.
     *
     * @param geolocation - geolocation from which to search for assistants
     * @param limit - the number of assistants to return
     * @return a sorted collection of assistants ordered ascending by distance from geoLocation
     */
    SortedSet<Assistant> findNearestAssistants(Geolocation geolocation, int limit);

    /**
     * This method reserves an assistant for a Geico customer that is stranded on the roadside due to a disabled vehicle.
     *
     * @param customer - Represents a Geico customer
     * @param customerLocation - Location of the customer
     * @return The Assistant that is on their way to help
     */
    Optional<Assistant> reserveAssistant(Customer customer, Geolocation customerLocation);

    /**
     * This method releases an assistant either after they have completed work, or the customer no longer needs help.
     *
     * @param customer - Represents a Geico customer
     * @param assistant - An assistant that was previously reserved by the customer
     */
    void releaseAssistant(Customer customer, Assistant assistant);
}
```