Lab 03 – Delegates & Events

Hinweis: Das Aufgabenblatt enthält neben den Aufgaben zum Modulpraktikum auch Zusatzfragen zum Verständnis und zur Repetition des Stoffes. Diese Fragen müssen nicht für die Fallstudie beantwortet werden und sind freiwillig. Sie können jedoch zur Prüfungsvorbereitung verwendet werden.

Aufgabe 1

Für den später zu implementierenden Algorithmus benötigen Sie in der `Cities`-Klasse eine Such-Funktion, die eine Stadt anhand des Namens findet und das entsprechende `City`-Objekt zurückgibt. Erweitern Sie dazu die Klasse `Cities` um einen zweiten Indexer, der als Argument einen Städtenamen akzeptiert:

```
public City this[string cityName]
```

Bei der Suche soll zwischen Gross- und Kleinschreibung **nicht** unterschieden werden. Wenn keine Stadt gefunden wurde, soll der Indexer eine `KeyNotFoundException` werfen. Wird eine Stadt mit dem Wert null übergeben, soll der Indexer eine `ArgumentNullException` werfen.

Hinweis: Verwenden Sie für die Formulierung der Suchabfrage im Indexer einen Predicate Delegate, welches Sie der `List.Find(…)`-Methode übergeben. Zum Vergleich der beiden strings sollen sie den Vergleichstyp InvariantCultureIgnoreCase verwenden.

Aufgabe 2

Sie finden auf dem Netzwerkshare die Klassen `Link` und `Links`. Fügen Sie beide Klassen dem Projekt RoutePlannerLib hinzu.

Erläuterungen:

- Die Klasse `Link`, repräsentiert die Verbindung zweier City-Objekte und deren Distanz. Ausserdem verfügt sie über das Property `TransportMode`, mit dem das Transportmittel für die Verbindung zwischen zwei Städten definiert werden kann. Die möglichen Transportmittel sind im Enum `TransportMode` im File `Link.cs` definiert.

- Die Klasse `Links` repräsentiert Liste von Verbindungen (einzelne `Link`'s) zwischen zwei Städten. Sie erlaubt das Einlesen von Städteverbindungen aus Dateien mittels der Methode `ReadLinks(..)`. Sie finden dazu auf dem Netzwerkshare die Datei `linksTestDataLab3.txt` mit einer Liste von Bahnverbindungen, sowie eine weitere City-Datei `citiesTestDataLab3.txt`. Die Klasse verfügt ausserdem über die noch leere Methode

  ```
  public List<Link> FindShortestRouteBetween(string fromCity,
            string toCity, TransportMode transportMode);
  ```

  die die kürzeste Route zwischen den beiden Städten mit den Namen `fromCity` und `toCity` mit dem angegebenen `transportMode` berechnet und in Form einer Liste von Verbindungen zurückgibt.

  Hinweis: Die Implementierung der Methode wird in einem späteren Lab erfolgen. In der jetzigen Version soll die Methode eine leere Liste ohne Links zurückgeben.

Aufgabe 3

Die nationale Tourismusbehörde will wissen, wie oft Routenanfragen zu einzelnen Städten gemacht werden. Sie will daher jedes Mal informiert werden, wenn eine Routenanfrage gemacht wird und wissen, welche Route angefragt wurde. Die Klasse muss daher allen interessierten Clients eine Möglichkeit bieten, sich zu registrieren und über Routenanfragen informiert zu werden. Erweitern Sie dazu die `RoutePlannerLib` um die benötigte Funktionalität:

a) Definieren Sie für die Klasse `Links` das folgende Delegate:

```
delegate void RouteRequestHandler(object sender, RouteRequestEventArgs e);
```

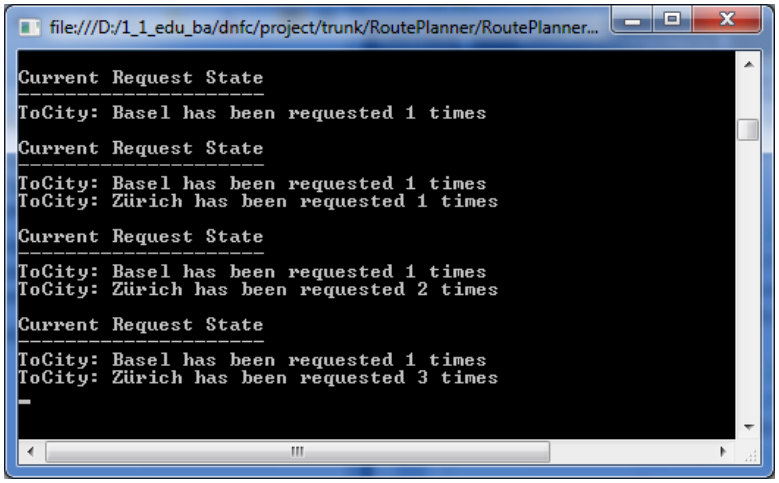
Die ebenfalls neu zu definierende Klasse `RouteRequestEventArgs` soll Informationen über die Anfrage (also `fromCity`, `toCity` und den Transportmodus) liefern und gemäss C#-Konvention von der Klasse `System.EventArgs` abgeleitet sein. Alle notwendigen Daten sollen über einen Custom-Constructor übergeben werden können.

b) Ergänzen Sie die Klasse `Links` um den Event `RouteRequested`.

c) Bauen Sie den Event-Aufruf an einer geeigneten Stelle in der `FindShortestRouteBetween`-Methode der Klasse `Links` ein und übergeben Sie die notwendigen Daten. Da die Methode im Moment noch nichts macht, können Sie den Event an beliebiger Stelle in der Methode auslösen.

Aufgabe 4

a) Implementieren Sie nun eine Klasse `RouteRequestWatcher`, welche eine Methode `LogRouteRequests` als Event-Handler für die Routenanfragen implementiert. Die Methode soll speichern, wie oft welche Zielstadt angefragt wurde. Die aktuellen Zählerstände sollen jeweils auf die Konsole geschrieben werden. Sie müssen nur die Methode `LogRouteRequests` implementieren. Die Methode wird von den Unittests der nächsten Aufgabe getestet.



Hinweis: Für die Verwaltung der Anzahl Anfragen pro Stadt bietet sich ein Dictionary an (z.B. `Dictionary<City, int>`), das die Liste der angefragten Städtenamen und Anzahl Anfragen verwaltet.

Die Anzahl der gemachten Anfragen soll sich für eine beliebige Stadt mit der Methode `int GetCityRequests(City city)` abfragen lassen.

Aufgabe 5

Testen Sie Ihre Implementierung mit den ausgelieferten Unit-Tests in den Files `*Test_Lab03.cs`, indem Sie diese Files dem Test-Projekt hinzufügen. Fügen Sie auch die neuen Testdaten Files ihrem Testprojekt-Verzeichnis hinzu (Denken Sie an die «Copy-if newer» Property nach dem Hinzufügen zum Projekt)

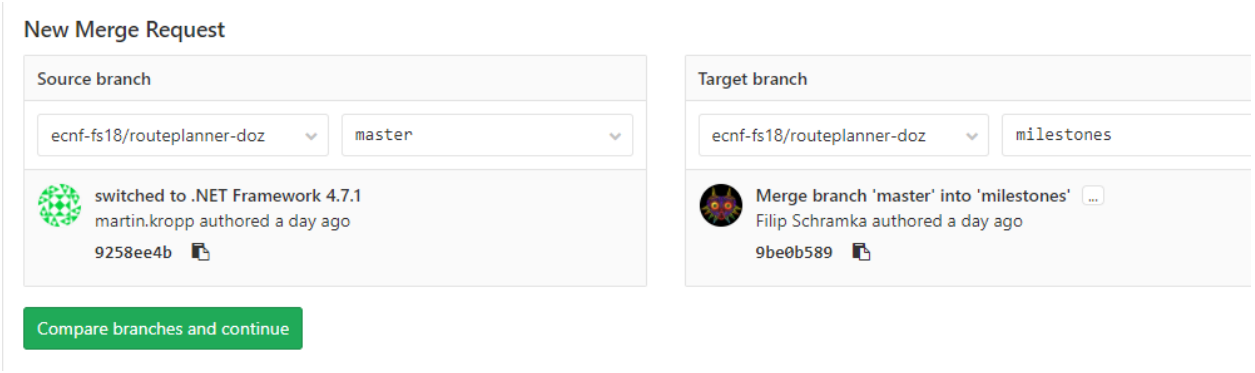
Aufgabe 6

Erstellen Sie einen Merge-Request in den Branch «milestones» und senden Sie ihn an Ihren Dozenten (Falls Sie den Branch noch nicht erstellt haben; siehe Details im Lab02).

Zur Beurteilung des Meilensteins wird die Code-Basis zum Zeitpunkt des Merge-Requests verwendet**. Erledigen Sie diesen Schritt also erst, wenn Sie mit dem Meilenstein wirklich fertig sind und alle Tests erfolgreich waren**.

Erstellen eines Merge-Requests in GitLab

1. Wählen Sie «Merge Request» im linken Navigation Menu.
2. Klicken Sie rechts oben «New Merge Request»
3. Wählen Sie ihren Entwicklungs-Branch und den Ziel-Branch „milestones“ aus.



4. Wählen Sie Continue und ergänzen Sie die Felder nach Bedarf. Geben Sie als «Assignee» ihren Dozierenden an und Klicken Sie den submit-Button.

Testfragen

- Was ist ein Delegate? Was ein Delegate Type?
- Was sind die Unterschiede zwischen Events und Delegates?
- Was bedeutet `Action`? Was bedeutet `Func`? Was bedeutet `Predicate`?
- Wozu werden diese Typen typischerweise eingesetzt?
- Wann hat ein Delegate/Event den Wert `null`?
- Was passiert, wenn man einen Event mit dem Wert `null` ausführt?
- Was geschieht, wenn Exceptions in einem Multicast Delegate auftreten?