

# Projet Structures de données : avancé

## 1 Objectif général du projet

Un site web spécialisé dans le voyage à travers le monde à vélo désire développer un outil permettant aux cyclotouristes de calculer des bons itinéraires pour aller d'un pays à un autre à vélo. Il existe pas mal de sites proposant de calculer des itinéraires les plus courts entre différentes villes (ex : Google Maps) mais ceux-ci ne sont pas adaptés pour les longs voyages en vélo. En effet, les cyclotouristes ont souvent deux contraintes qui ne sont pas proposés par ces sites :

- Les cyclotouristes n'aiment pas traverser beaucoup de frontières. En effet, à chaque frontière, les cyclotouristes ont souvent besoin d'effectuer des démarches administratives (visas, ...) qui sont souvent lourdes et pénibles.
- Les cyclotouristes préfèrent choisir des itinéraires peu fréquentés. Ils préfèrent la solitude aux pays surpeuplés.

L'objectif général du projet est d'implémenter un programme java permettant de calculer des itinéraires entre deux pays. Votre programme sauvegardera les itinéraires calculés dans un fichier au format xml. Voici un exemple d'itinéraire:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<itineraire arrivee="India" depart="Belgium" nbPays="6" sommePopulation="2871572652">
  <pays cca3="BEL" nom="Belgium" population="11175653" />
  <pays cca3="DEU" nom="Germany" population="80523700" />
  <pays cca3="POL" nom="Poland" population="38533299" />
  <pays cca3="RUS" nom="Russia" population="143500000" />
  <pays cca3="CHN" nom="China" population="1361170000" />
  <pays cca3="IND" nom="India" population="1236670000" />
</itineraire>
```

Ceci représente un itinéraire de la Belgique vers l'Inde. Dans l'ordre, cet itinéraire passe par l'Allemagne, la Pologne, la Russie et la Chine. Le nombre de pays traversé est 6. La somme des populations de ces différents pays est 2871572652. Les itinéraires que vous calculerez devront respecter ce format.

## 2 Sur Moodle

Pour mener à bien votre projet, nous vous fournissons plusieurs fichiers :

- countries.xml contenant des informations à propos des pays du monde et leurs frontières. Chaque pays est identifié par un code appelé cca3. Ces informations peuvent être vues comme un graphe non dirigé. Les sommets de ce graphe sont les pays. Il y a un arc entre deux pays s'il y a une frontière entre ces deux pays.
- countries.xsd, un schéma xml permettant de valider countries.xml.
- une classe Main.java que vous ne pouvez pas modifier. Le code de cette classe est présenté ci-dessous. La classe à construire Graph devra contenir deux méthodes pour calculer des itinéraires. Ces deux méthodes seront détaillées dans la suite du document. Ces méthodes

prennent trois paramètres. Les deux premiers paramètres sont les codes cca3 des pays de départ et d'arrivée. Le troisième paramètre est le nom du fichier où il faudra sauvegarder le fichier xml de résultat.

```
import java.io.File;
import javax.xml.parsers.SAXParser;
import javax.xml.parsers.SAXParserFactory;

public class Main {
    public static void main(String[] args) {
        try {
            File inputFile = new File("countries.xml");
            SAXParserFactory factory = SAXParserFactory.newInstance();
            SAXParser saxParser = factory.newSAXParser();
            SAXHandler userhandler = new SAXHandler();
            saxParser.parse(inputFile, userhandler);
            Graph g = userhandler.getGraph();
            g.calculerItineraireMinimisantNombreDeFrontieres("BEL", "IND", "output.xml");
            g.calculerItineraireMinimisantPopulationTotale("BEL", "IND", "output2.xml");
        } catch (Exception e) {
            e.printStackTrace();
            System.exit(0);
        }
    }
}
```

### 3 Tâches à effectuer

#### 3.1 Tâches obligatoires

Nous vous demandons de rendre la classe Main fonctionnelle en réalisant les tâches suivantes :

- Implémenter un parseur SAX permettant la lecture du fichier XML et la construction du graphe en mémoire en utilisant les structures de données adéquates.
- Implémenter la méthode `calculerItineraireMinimisantNombreDeFrontieres` qui calcule l'itinéraire entre deux pays passant par le moins de frontières possibles (voir point 4). S'il est impossible d'aller d'un pays à un autre, votre programme lancera une exception.
- Implémenter la méthode `calculerItineraireMinimisantPopulationTotale` qui calcule l'itinéraire entre deux pays pour lequel la somme des populations des pays traversés est la plus petite (voir point 5). S'il est impossible d'aller d'un pays à un autre, votre programme lancera une exception.
- Sauvegarder ces itinéraires dans des fichiers XML.

De plus, nous vous demandons d'écrire la DTD `countries.dtd` qui permet de valider les mêmes documents que `countries.xsd` (tout en étant plus permissive évidemment).

#### 3.2 Tâches bonus

Si vous avez fini trop tôt et que vous vous embêtez durant les séances, vous pouvez faire une ou plusieurs tâches bonus. L'objectif est de faire le projet durant les séances, ne prenez pas du temps à la maison pour faire ces tâches bonus ; privilégiez les autres cours comme le projet Projet AE.

Voici les différentes tâches bonus :

1. Implémentez un parseur DOM pour créer le graphe.

2. Ajoutez une troisième façon de calculer un itinéraire en minimisant d'abord le nombre de pays traversés et ensuite la somme des populations.
3. Donnez la DTD permettant de valider les itinéraires calculés. Pour ajouter la dtd dans un fichier xml, voici une petite aide : <https://stackoverflow.com/questions/13553614/how-to-add-doctype-in-xml-document-using-dom-java>

## 4 Itinéraire passant par le moins de pays différents

La méthode `calculerItineraireMinimisantNombreDeFrontieres` calculera les itinéraires qui passent par le moins de pays possible.

Par exemple, pour aller de Belgique en Inde, l'itinéraire le plus court passe par 6 pays.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<itineraire arrivee="India" depart="Belgium" nbPays="6" sommePopulation="2871572652">
  <pays cca3="BEL" nom="Belgium" population="11175653" />
  <pays cca3="DEU" nom="Germany" population="80523700" />
  <pays cca3="POL" nom="Poland" population="38533299" />
  <pays cca3="RUS" nom="Russia" population="143500000" />
  <pays cca3="CHN" nom="China" population="1361170000" />
  <pays cca3="IND" nom="India" population="1236670000" />
</itineraire>
```

Il est peut-être possible de trouver d'autres itinéraires passant par 6 pays.

## 5 Itinéraire minimisant la population totale des pays traversés

La méthode `calculerItineraireMinimisantPopulationTotale` calculera l'itinéraire qui aura la plus petite somme des populations des pays traversés.

Pour aller de Belgique en Inde, il est possible de trouver un itinéraire dont la somme des populations des pays traversés vaut 1686913442. Cet itinéraire passe par 14 pays.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<itineraire arrivee="India" depart="Belgium" nbPays="14" sommePopulation="1686913442">
  <pays cca3="BEL" nom="Belgium" population="11175653" />
  <pays cca3="FRA" nom="France" population="65806000" />
  <pays cca3="CHE" nom="Switzerland" population="8085300" />
  <pays cca3="AUT" nom="Austria" population="8501502" />
  <pays cca3="SVN" nom="Slovenia" population="2061405" />
  <pays cca3="HRV" nom="Croatia" population="4290612" />
  <pays cca3="MNE" nom="Montenegro" population="620029" />
  <pays cca3="ALB" nom="Albania" population="2821977" />
  <pays cca3="MKD" nom="Macedonia" population="2058539" />
  <pays cca3="BGR" nom="Bulgaria" population="7282041" />
  <pays cca3="TUR" nom="Turkey" population="75627384" />
  <pays cca3="IRN" nom="Iran" population="77068000" />
  <pays cca3="PAK" nom="Pakistan" population="184845000" />
  <pays cca3="IND" nom="India" population="1236670000" />
</itineraire>
```

## 6 Organisation et livrables

Ce projet se déroule du 8 mars 2021 au 28 mars 2021 par groupe de deux étudiants. Les étudiants d'un même groupe doivent être dans la même série. Durant tout le projet, la présence aux cours est obligatoire.

Si une série comporte un nombre impair d'étudiant, nous accepterons un seul groupe de 3 étudiants. Ce groupe de 3 étudiants devra également faire la première et la troisième tâche bonus.

Le projet est à remettre via « Moodle » pour le dimanche 28 mars 2021 à 12h00 (midi). Nous ne demandons pas de rapport. Si certaines choses méritent des explications, vous pouvez les fournir en commentaire dans les différents fichiers.

**Le plagiat sera lourdement sanctionné (au minimum 0 à l'UE). Tous les projets seront automatiquement testés par un logiciel de détection de plagiat.**