

**Devoir maison**

## Des problèmes qui ne sont pas tristes !

### 1 Sujet

On s'intéresse à la modélisation et la résolution de problèmes grâce à la théorie des graphes. Les trois problèmes énoncés ci-après peuvent être résolus, de manière plus ou moins similaire, en les modélisant par un graphe.

**Sudoku** Une grille de Sudoku est composée de  $9 \times 9 = 81$  cases qu'on remplit par des chiffres (de 1 à 9). Il est interdit d'avoir plusieurs fois le même chiffre :

- sur une même ligne,
- sur une même colonne,
- dans chacun des 9 sous-grilles de  $3 \times 3 = 9$  cases.

Étant donnée une grille avec déjà quelques chiffres dans certaines cases, l'objectif du jeu est de deviner comment remplir les cases vides. Voir sur Internet pour des exemples et une explication plus complète. Attention il se peut qu'une grille donnée soit impossible à résoudre.

**Problème :** Comment remplir une grille donnée qui admet une solution ?

**Cartes géographiques** Les cartes géographiques comme celles qu'on peut trouver dans un atlas, ou celles qui sont accrochées dans les classes, ou les livres de géographie, représentent généralement une partie du monde et, afin de mieux délimiter les pays, on colorie ceux-ci de différentes couleurs.

**Problème :** Comment colorier les pays de sorte que deux pays frontaliers ne soient pas de la même couleur ?

Pour aller plus loin, on pourrait se poser la question du nombre minimum de couleurs nécessaires.

**Allocation de fréquences dans les réseaux GSM** Dans un réseau d'antennes, on souhaite éviter que deux antennes voisines se voient affectée la même fréquence pour ne pas avoir d'interférences.

**Problème :** Comment affecter des fréquences d'un réseau d'antennes donnée (c'est-à-dire le voisinage de chaque antenne) ?

Pour aller plus loin, on pourrait se poser la question du nombre minimum de fréquences nécessaires.

### 2 Travail à faire

Pour ce devoir vous devez :

- identifier le point commun entre ces problèmes et proposer une modélisation sous forme de graphes pour chacun d'entre eux,
- proposer des algorithmes pour résoudre le problème, en prouver la terminaison et la correction,
- implémenter ces algorithmes, les valider en les exécutant sur des instances des problèmes et en interprétant les solutions obtenues,
- trouver une application, différente de celles citées ci-dessous, qui utilise la même modélisation.

#### 2.1 Modélisation

Après étude des trois applications, vous devez aboutir à la conclusion que **le problème à résoudre est en fait le même**, une fois la modélisation trouvée.

Pour chacune des 3 applications citées, vous indiquerez :

- ce qui est modélisé par les sommets,
- ce qui est modélisé par les arcs ou les arêtes (suivant le type de graphe que vous jugez nécessaire),

- les éventuelles informations complémentaires que vous jugerez nécessaires à ajouter au graphe,
- vous expliquerez ce qu'on cherche à réaliser dans le graphe, et en quoi cela permet de répondre au problème de l'application.

Vous terminerez par énoncer clairement le problème posé auquel on aboutit, en dehors des applications spécifiques citées ici.

**Rendu intermédiaire numéro 1 entre le 22 et le 26 novembre 2021.** Rendre sous forme rédigée la partie modélisation du DM.

## 2.2 Algorithmes

Le problème de graphes auquel vous devriez aboutir est un problème réputé difficile, sauf dans des cas particuliers. Ainsi, on trouvera des algorithmes permettant d'avoir une solution approchée, mettant en œuvre des *heuristiques*. Néanmoins, il se peut que votre proposition d'algorithme fasse qu'une solution ne soit pas trouvée. Dans ce cas il faudra le détecter et arrêter la recherche.

- Expliquer ce qu'on entend par « problème difficile ».
- Qu'est-ce qu'une heuristique ?
- Proposer un premier algorithme naïf, qui permettra de résoudre le problème lorsqu'on ne met pas de contraintes sur le nombre d'éléments permettant de résoudre le problème (typiquement le nombre de fréquences pour le problème des antennes, ou le nombre de couleurs pour le problème des cartes).
- Proposer un second algorithme qui mettra en œuvre une heuristique dont l'objectif sera de tenter d'envisager les sommets dans un ordre plus habile.
- Discuter des limites de cette heuristique.

**Rendu intermédiaire numéro 2 entre le 6 et le 10 décembre 2021.** Rendre sous forme rédigée la partie algorithmes du DM.

## 2.3 Applications

Vous implémenterez les deux algorithmes en Python dans un module nommé `algorithms.py`. Vous rendrez trois *main* nommés `sudoku.py`, `gsm.py` et `map.py` qui, étant donnés des fichiers modélisant une grille de sudoku, une disposition d'antennes ou une carte fournis en argument, produira une solution.

Chacun des 3 scripts se lancera de la même manière :

```
$ python3 script.py -i input.txt -o output.txt
```

où les noms de fichier sont remplacés par les noms adéquats. Le format des fichiers est donné plus bas.

**Rendu final pour le 17 décembre 2021.** Rendre les fichiers Python et un fichier `README.md` qui explicitera le fonctionnement des programmes, comment les exécuter (avec les commandes précises), et commentera les résultats produits.

## 2.4 Extension

Trouvez-vous des problèmes dont la modélisation aboutit à une structure de graphe particulière et pour laquelle on est capable d'avoir une résolution exacte du problème ?

## 3 Consignes de rendu

Le travail devra être réalisé en binôme. La modalité de rendue sera fixée par votre enseignant·e de TD/TP, ainsi que les dates exactes des rendus intermédiaires. La date de rendu final est fixée au **vendredi 17 décembre 2021** pour tous les groupes.

## 4 Formats d'entrée et de sortie

### 4.1 Sudoku

Le fichier d'entrée sera donné sous la forme d'un fichier texte de 3 colonnes, les deux premières donnant le numéro de la case (l'abscisse et l'ordonnée), le troisième donnant le chiffre contenu dans la case.

Par exemple :

```
1 9 1
1 7 2
4 5 6
```

Le fichier de sortie aura le même format mais contiendra 81 lignes, ou bien sera vide pour le cas où une solution ne peut être trouvée.

### 4.2 Cartes géographiques

Le fichier d'entrée sera donné sous la forme d'un fichier texte de 2 colonnes, donnant les couples de pays ayant une frontière.

Par exemple :

```
France Belgique
France Allemagne
Belgique Allemagne
Belgique Pays-Bas
```

Le fichier de sortie donnera une couleur pour chaque pays. Pour simplifier, on pourra supposer les couleurs codées par des entiers.

Par exemple :

```
France 0
Allemagne 1
Belgique 2
Pays-Bas 0
```

### 4.3 Antennes GSM

Le fichier d'entrée sera donné sous la forme d'un fichier texte de 2 colonnes, donnant les couples d'antennes voisines.

Par exemple :

```
1 2
1 3
4 5
4 6
5 6
```

Le fichier de sortie donnera une fréquence pour chaque antenne. Pour simplifier, on pourra supposer les fréquences codées par des entiers à partir de 0.

Par exemple :

```
1 0
2 1
3 1
4 0
5 1
6 2
```