

Coloriage de carte

Objectif

L'objectif de ce projet est le coloriage d'une carte de sorte que 2 régions **adjacentes** (voisines) ne soient pas coloriées par la même couleur et que le nombre de couleurs utilisées soit minimisé.

Hypothèses de travail

- Une carte est constituée d'un ensemble de régions. Elle est stockée sous forme compressée dans une liste de lignes, chaque ligne décrivant une ligne de la carte sous la forme suivante :

$nom_1 \ nb_1 \ nom_2 \ nb_2 \ nom_3 \ nb_3 \dots$

où nom_i représente un nom de région et nb_i représente le nombre de points de la région nom_i sur la ligne considérée.

Par exemple, la ligne : ['G', 1, 'H', 4, 'A', 3, 'B', 1, 'T', 1,]
 décrit la ligne de la carte : ['G', 'H', 'H', 'H', 'H', 'A', 'A', 'A', 'B', 'T']

Exemple de carte complète sous sa forme initiale compressée :

```
carte = [ ["C", 4, "F", 3],
          ["C", 1, "B", 1, "D", 1, "B", 2, "E", 1, "F", 1],
          ["A", 1, "B", 4, "E", 1, "A", 1],
          ["A", 7] ]
```

- La relation d'adjacence (voisinage) entre régions tient compte d'une connectivité de 4. Ceci signifie que 2 régions R1 et R2 sont adjacentes (voisines) s'il existe un élément de R1 dont l'un des 4 voisins (au-dessus, au-dessous, à droite ou à gauche) appartient à R2.

Gondor	Harad	Harad	Harad	Harad	Armor	Armor	Armor	Barandur	Imladris
Gondor	Gondor	Harad	Harad	Harad	Harad	Armor	Armor	Barandur	Imladris
Gondor	Terre du milieu	Terre du milieu	Harad	Harad	Barandur	Barandur	Barandur	Barandur	Imladris
Gondor	Terre du milieu	Terre du milieu	Moria	Harad	Harad	Harad	Barandur	Mordor	Mordor
Gondor	Terre du milieu	Terre du milieu	Moria	Moria	Moria	Mordor	Mordor	Mordor	Mordor
Gondor	Gondor	Terre du milieu	Terre du milieu	Terre du milieu	Terre du milieu	Eriador	Eriador	Eriador	Lorien

Dans la carte ci-dessus, les régions *Armor* et *Harad*, *Harad* et *Barandur* sont adjacentes, alors que les régions *Gondor* et *Barandur*, *Terre du milieu* et *Mordor* ne le sont pas (ces exemples ne définissent pas la relation d'adjacente de façon exhaustive).

- Dans la suite, nous supposerons que la carte n'est constituée que de régions connexes.

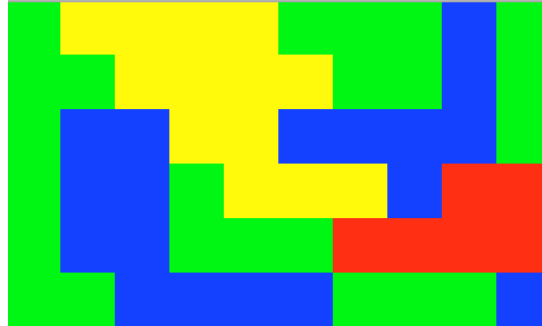
Questions

Le but du projet est d'écrire une fonction de coloriage qui permette d'associer à chaque région une couleur qui soit différente des couleurs de ses voisins et qui affiche cette carte coloriée.

Le théorème des quatre couleurs indique que, dans tous les cas, il est possible de n'utiliser que quatre couleurs au maximum, mais on ne vous demande pas d'arriver obligatoirement à ce résultat. Dans un premier temps, on se contentera de trouver un coloriage qui ne nécessite pas forcément autant de couleurs que de régions, le nombre de couleurs disponibles pour le coloriage n'étant pas limité.

Afin d'illustrer de façon visuelle le coloriage proposé, on demande de dessiner la carte coloriée à l'aide des fonctions du module *Turtle* de *Python*.

La carte précédente comprenant 10 régions, peut être représentée, par exemple, avec les 4 couleurs suivantes :



Démarche

Le travail s'organisera de la façon suivante :

- ✓ Imaginer et décrire un algorithme qui permet d'associer à chaque région une couleur. Procéder en plusieurs étapes, écrire un premier algorithme de haut niveau non détaillé et affiner progressivement cet algorithme. On peut s'inspirer du vocabulaire et de certains algorithmes décrits dans la théorie des graphes.
- ✓ A partir des opérations à mettre en œuvre pour réaliser ces algorithmes, en déduire les objets *Python* les mieux adaptés pour représenter les informations manipulées.
- ✓ Identifier les fonctions principales et secondaires utiles à la programmation de cet algorithme.
- ✓ Programmer ces fonctions. On pourra les tester sur les exemples de cartes fournis sur Arche.
- ✓ Compléter le programme en affichant la carte à l'aide des fonctions *turtle* (on peut utiliser le programme qui a permis d'afficher une image matricielle).
- ✓ Outre la qualité de l'algorithme mis en œuvre, on accordera également une grande importance à la structuration du programme en fonctions et en plusieurs fichiers stockant ces fonctions selon leurs rôles, à la présentation du programme, aux choix des noms de variables et aux commentaires associés.