TCA101 - 1er semestre : TP 3

Introduction à la récursivité

Type abstrait graphes centré arêtes

Exercice 3.1

1. La fonction factorielle peut être définie de la manière suivante :
   - $0! = 1$
   - $\forall n > 0, n! = 1 \times 2 \times \ldots \times (n-1) \times n$

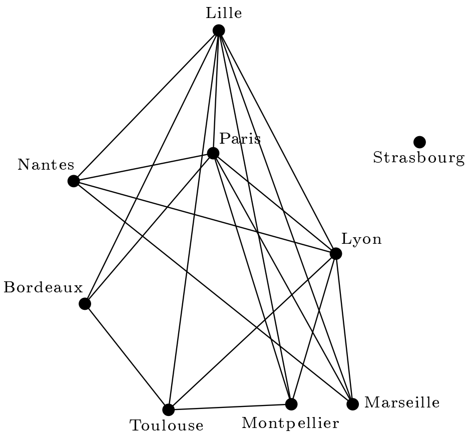
   Écrire une fonction **fact_iterative** qui utilise cette definition pour calculer n!
2. La fonction factorielle peut être définie de la manière suivante :
   - $0! = 1$
   - $\forall n > 0, n! = n \times (n-1)!$

   Écrire une fonction **fact_recursive** qui utilise cette definition pour calculer n!

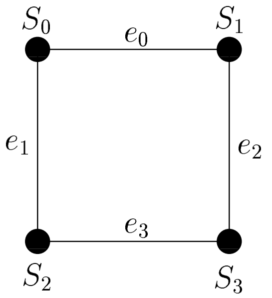
Dans toute la suite, on considère le type abstrait graphe appelé *type abstrait centré-arêtes.* Il contient les opérations suivantes :

- `nb_nodes(G)` : donne le nombre de sommets du graphe G,
- `nb_edges(G)` : donne le nombre d'arêtes du graphe G,
- `node(G, i)` : donne le sommet numéro i du graphe G, les sommets étant numérotés à partir de 0,
- `edge(G, i)` : donne l'arête numéro i du graphe G, les arêtes étant numérotées à partir de 0,
- `edge_end(e, i)` : pour $i \in \{0, 1\}$, donne le sommet première ou deuxième extrémité de l'arête e,
- `mark_node(s)` : marque le sommet s,
- `unmark_node(s)` : démarque le sommet s,
- `node_marked(s)` : détermine si le sommet s est marqué,
- `mark_edge(e)` : marque l'arête e,
- `unmark_egde(e)` : démarque l'arête e,
- `edge_marked(e)` : détermine si l'arête e est marquée.

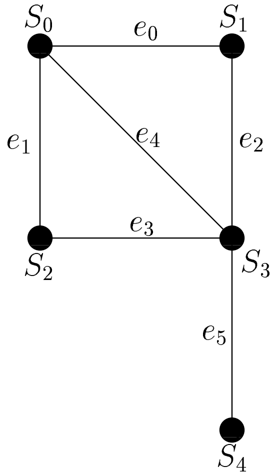
Pour tester les fonctions, on utilisera les graphes `g1`, `g2`, `g3`, `tgv` du fichier `exemples_graphes.py` et le graphe `weblabri` du fichier `weblabriuniversity.py`.



tgv



g1



g3

Exercice 3.2
Écrire une fonction `degre_moyen` qui a pour paramètre un graphe G et qui retourne le degré moyen du graphe G.

Exercice 3.3
Écrire une fonction `degre_sommet` qui a pour paramètre un graphe G et un sommet s et qui retourne le degré du sommet s.

Exercice 3.4
Écrire une fonction `degre` qui a pour paramètre un graphe G et un entier n, et qui retourne le premier sommet rencontré dont le degré est n. Si aucun sommet ne répond à la question la fonction ne retourne rien.

Exercice 3.5
Écrire une fonction `sommet_isole` qui a pour paramètre un graphe G et qui retourne 1 s'il existe un sommet isolé dans ce graphe, 0 sinon. On écrira deux versions de cette fonction, l'une n'utilisant pas les marques sur les sommets, l'autre les utilisant et on comparera les temps d'exécution (notament sur le graphe weblabri)

Exercice 3.6
Écrire une fonction `degré_max` qui a pour paramètre un graphe G et qui calcule le degré maximum des sommets de ce graphe.

Exercice 3.7
Écrire une fonction `sommet_de_plus_haut_degre` qui a pour paramètre un graphe G et qui retourne le sommet de plus haut degré du graphe G. Dans le cas où il existe plusieurs sommets répondant à la question, seul le premier sommet rencontré est retourné. Cette fonction ne parcourt qu'une seule fois les sommets du graphe.