

Correction du CC2 Informatique

J1 MI 1003, groupe B3, Université Bordeaux

Attention recto et verso!

Exercice 1 (5 points) Énoncer la formule des poignées de main.

La somme des degrés des sommets d'un graphe est égale au double du nombre d'arêtes, soit en termes mathématiques :

$$\sum_{s \text{ sommet de } G} \text{degré}(s) = 2 \times \text{nombre d'arêtes.}$$

Quelques conseils : Apprenez votre cours. (!) N'utilisez pas le signe \sum si vous ne le comprenez pas. Un graphe n'a pas de degré, seuls les sommets en ont. Ecrivez bien le mot 'arête', un seul r, un seul t, un accent circonflexe, les autres orthographes ont tendance à horripiler.

Exercice 2 (5 points)

```
def mystere(n):
    x=3
    for i in range(n):
        x=2*x
    return x
```

Que renvoie `mystere(0)` ? 3

Que renvoie `mystere(1)` ? 6

Que renvoie `mystere(2)` ? 12

Que renvoie `mystere(3)` ? 24

Que calcule `mystere(n)`, pour un entier naturel n quelconque ? 3×2^n

Quelques conseils : Attention au décalage d'indice. La boucle `for i in range(n)` : effectue précisément n itérations. (Elle commence en 0 et finit en $n - 1$.) Il n'y a pas d'erreur avec un `range(0)`, le contenu est tout simplement non lu. (Je l'ai dit plusieurs fois en cours.)

Exercice 3 (6+3 points)

Ecrire une fonction Python `grapheImpair(G)` qui renvoie `True` si tous les sommets du graphe G ont un degré impair, `False` sinon.

```
def grapheImpair(G):
    for s in listeSommets(G):
        if (degre(s)%2 == 0):
            return False
    return True
```

La structure a été vue en cours (et j'ai insisté).

Pourquoi un graphe avec uniquement des sommets de degré impair a-t-il un nombre pair de sommets ?

Modulo 2, on a :

$$0 \equiv 2 \times \text{nb d'arêtes} \stackrel{\text{formule des poignées de main}}{=} \sum_{s \text{ sommet}} \text{degré}(s) \stackrel{\text{par hypothèse}}{\equiv} \sum_{s \text{ sommet}} 1 = \text{nb de sommets.}$$

Tourner la page s.v.p.

Exercice 4 (4 points)

Un sommet **pseudo-isolé** est un sommet qui n'est relié à aucun autre sommet. (Mais il peut comporter des boucles.) Écrire une fonction Python `AvecPseudoIsoles(G)` qui renvoie `True` si le graphe G a au moins un sommet pseudo-isolé, `False` sinon.

Le plus simple était d'écrire deux fonctions, dont une qui teste si un sommet est pseudo-isolé. Quand est-ce qu'un sommet est pseudo-isolé? Quand tous ses voisins ne sont rien d'autre que lui-même...

```
def pseudoIsole(s):
    for v in listeVoisins(s):
        if v != s:
            return False
    return True

def avecPseudoIsole(s):
    for s in listeSommets(G):
        if pseudoIsole(s):
            return True
    return False
```

Annexe : voici un rappel des principales fonctions disponibles pour manipuler les graphes (à vous de voir celles qui sont utiles pour ce devoir).

L'argument G est un graphe	
<code>listeSommets(G)</code>	retourne la <i>liste</i> des <i>sommets</i> de G .
<code>nbSommets(G)</code>	retourne le <i>nombre</i> de <i>sommets</i> de G .
<code>sommetNom(G,etiquette)</code>	retourne le <i>sommet</i> de G désigné par son <i>nom</i> (étiquette). Exemple : <code>sommetNom (Europe, 'Italie')</code> .
<code>sommetNumero(G,i)</code>	retourne le <i>sommet</i> numéro i dans G ; la numérotation commence à 0.

L'argument s est un sommet	
<code>listeVoisins(s)</code>	retourne la <i>liste</i> des <i>voisins</i> de s .
<code>degre(s)</code>	retourne le <i>degré</i> de s .
<code>nomSommet(s)</code>	retourne le <i>nom</i> (étiquette) de s .
<code>colorierSommet(s,c)</code>	colorie s avec la couleur c . Exemples de couleurs : <code>'red'</code> , <code>'green'</code> , <code>'blue'</code> , <code>None</code> .
<code>couleurSommet(s)</code>	retourne la <i>couleur</i> de s .
<code>marquerSommet(s)</code>	
<code>demarquerSommet(s)</code>	marque ou démarque s .
<code>estMarqueSommet(s)</code>	retourne <code>True</code> si s est marqué, <code>False</code> sinon.