



Examen d'informatique

2<sup>ème</sup> semestre AU : 2019 – 2020

Date : 8 juillet 2020 ; Durée : 1h30 ; Nombre de page : 2 ; Section : PB1

*L'utilisation des calculatrices n'est pas autorisée pour cette épreuve.*

*Le langage de programmation sera obligatoirement Python.*

\*

\* \*

**Implémentation.** Dans ce sujet, nous adopterons la syntaxe du langage Python. On rappelle qu'en Python, il importe de bien respecter les indentations car elles permettent de définir des blocs.

**Exercice 1 :**

Soit le script Python suivant :

```
def power ( Var x : int , y : float , z = 1 ) :  
    local r  
    r = ( x ** y ) mod z  
    print ( r )  
print( 4 + power(3,2) )
```

Corriger les erreurs et réécrire le script python.

## Exercice 2 : Réseaux sociaux

Le but de ce problème est de regrouper des personnes dans un réseau social. Nous supposons que les personnes sont numérotées de 0 à  $n - 1$  où  $n$  est le nombre total d'individus. Nous représenterons chaque lien d'amitié entre deux individus  $i$  et  $j$  par un tuple contenant leurs deux numéros dans un ordre quelconque, c.-à-d. par le tuple  $(i,j)$  ou par le tuple  $(j,i)$  indifféremment.

Un réseau social  $R$  entre  $n$  individus sera représenté par une liste  $R$  où :

- $R[0] = n$  : un entier contient le nombre d'individus appartenant au réseau
- $R[1] = \text{liens}$  : une liste non-ordonnée des liens d'amitié déclarés entre les individus

La figure 1 donne l'exemple d'un réseau social ' $R$ ' où chaque lien d'amitié entre deux personnes est représenté par un trait entre elles.

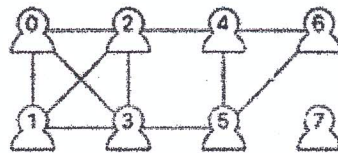
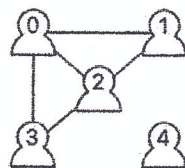


Figure 1 – Un réseau à 8 individus ayant 11 liens d'amitié déclarés

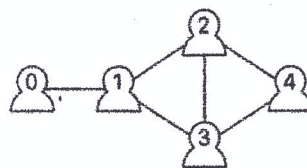
- $R = [ 8, [ (0,1), (1,3), (3,2), (2,0), (0,3), (2,1), (4,5), (6,4), (6,5), (2,4), (5,3) ] ]$

Travail demandé :

**Question 1.** Donner les instructions python permettant de créer les réseaux RA et RB pour chacun des deux réseaux sociaux ci-dessous :



Réseau A



Réseau B

**Question 2.** Ecrire une fonction **creerReseauVide( $n$ )** qui crée, initialise et renvoie la représentation sous forme de liste du réseau à  $n$  individus n'ayant aucun lien d'amitié déclaré.

**Question 3.** Ecrire une fonction **estUnLienEntre( $paire, i, j$ )** où  $paire$  est un tuple à deux éléments, et  $i$  et  $j$  sont deux entiers. La fonction renvoie True si les deux éléments contenus dans  $paire$  sont  $i$  et  $j$  dans un ordre quelconque ; et renvoie False sinon.

**Question 4.** Ecrire une fonction **sontAmis( $i, j, R$ )** qui renvoie True s'il existe un lien d'amitié entre les individus  $i$  et  $j$  dans le réseau  $R$  et renvoie False sinon.

**Question 5.** Ecrire une fonction **declareAmis( $i, j, R$ )** qui modifie le réseau  $R$  pour y ajouter le lien d'amitié entre les individus  $i$  et  $j$  si ce lien n'y figure pas déjà.

**Question 6.** Ecrire la fonction **amisDe( $i, R$ )** qui renvoie la liste des amis de  $i$  dans le réseau  $R$ .