Détermination de cycles. Corrigé

1)

```
def cycle(G) :
    n = len(G) ; M = {}
    # la valeur 0 est codée ici par le fait que x n'est pas dans le dictionnaire
    def traite(x) :
        if M[x] == 1 :  flag = True
        if not (x in M) :
            M[x] = 1
            for y in G[x] :  traite(y)
            M[x] = 2
    flag = False
    for s in range(n) :  traite(s)
    return flag
```

2)

```
def acyclique(G) :
    n = len(G) ; entrants = {} ; nombre = 0
    for x in G : entrants[s] = 0
# étape 1
    for x in range(n) :
        for y in G[x] :
            entrant[y] += 1
    pile = []
    for x in in range(n) :
        if entrant[x] :  pile.append(x)
# étape 2
    while pile :
        x = pile.pop() ; nombre += 1
        for y in G[x] :
            entrant[y] -= 1
            if entrant[y] == 0 :  pile.append(y)
# étape 3
```

```
15.         return (nombre == n)
```

Les sommets présents dans la pile à un moment donné n'ont aucun prédécesseur dans le graphe en cours d'étude. Il en est donc de même des sommets extraits de la pile, car les degrés entrants ne peuvent que décroître. De ce fait, aucun sommet ne peut être ajouté à la pile deux fois (), car tout sommet ajouté ligne 14 est un sommet qui était de degré entrant non nul à la ligne 11 de la même boucle, et donc n'appartenait pas à la pile. On en déduit que chaque arête est visitée au plus une fois ligne 13. Donc l'algorithme termine et la complexité est linéaire en $O(n+m)$.

On appelle hauteur $h(x)$ d'un sommet la longueur maximale d'un chemin terminant par x. Par exemple, les sommets sans prédécesseur sont de hauteur 0.

De façon générale, pour tout sommet s, on a : $h(x) = 1 + \max_{y \to x} h(y)$.

En particulier, un sommet est de hauteur finie ssi il est en de même de tous ses prédécésseurs. On en déduit par induction que tout sommet ajouté à la pile est de hauteur finie (car ses précédesseurs ont été ajoutés précédemment dans la pile).

Récirpoquement, tout sommet de hauteur finie est au cours de l'algorithme ajouté à la pile. On le montre à nouveau par récurrence sur la valeur de $h(x)$: par hypothèse de récurrence, ses prédécesseurs ont été ajoutés à la pile, puis enlevés de la pile, donc le degré entrant de x est alors nul, donc x est ajouté à la pile.

On en déduit que les sommets restants à la fin sont exactement les sommets de hauteur infinie. Or, il existe un cycle ssi il existe des sommets de hauteur infinie. D'où la correction de l'algorithme.