ISM MPSI/PCSI, Informatique
Année 2023/2024

Corrigé du TP Informatique 31

Exercice 1

1. On saisit :

```
def dfs_rec(S,A,sorg,tps,couleur,deb,fin,predec):
    deb[sorg]=tps[0]
    couleur[sorg]="gris"
    tps[0]+=1
    for s in A[sorg]:
        if couleur[s]=="blanc":
            predec[s]=sorg
            dfs_rec(S,A,s,tps,couleur,deb,fin,predec)
    couleur[sorg]="noir"
    fin[sorg]=tps[0]
    tps[0]+=1
```

2. On saisit :

```
def dfs_init(S,A,s0):
    tps,couleur,deb,fin,predec=[0],{},{},{},{}
    for s in S:
        couleur[s]="blanc"
    dfs_rec(S,A,s0,tps,couleur,deb,fin,predec)
    return deb,fin,predec,couleur
```

3. On obtient :

```
>>> dfs_init(S,A,4)
{2: 1, 3: 2, 4: 0, 5: 4, 7: 8, 8: 5},
{2: 10, 3: 3, 4: 11, 5: 7, 7: 9, 8: 6},
{8: 5, 2: 4, 3: 2, 5: 2, 7: 2},
{0: 'blanc', 1: 'blanc', 2: 'noir', 3: 'noir',
 4: 'noir', 5: 'noir', 6: 'blanc', 7: 'noir', 8: 'noir'}
```

Exercice 2

1. On saisit :

```
def nb_cc(S,A):
    """nb_cc(S:list,A:list)->int
    G=(S,A) graphe non orienté
    S : liste de sommets, A : liste d'arêtes
    Renvoie le nombre de composantes connexes de G"""
    tps,couleur,deb,fin,predec=[0],{},{},{},{}
    for s in S:
        couleur[s]="blanc"
    nb=0
    for s in S:
        if couleur[s]=="blanc":
            nb+=1
            dfs_rec(S,A,s,tps,couleur,deb,fin,predec)
    return nb
```

2. On obtient

```
>>> nb_cc(S,A)
3
```

Exercice 3

1. On saisit

```
def circuit_rec(S,A,sorg,tps,couleur,deb,fin,predec):
    deb[sorg]=tps[0]
    couleur[sorg]="gris"
    tps[0]+=1
    for s in A[sorg]:
        if couleur[s]=="blanc":
            predec[s]=sorg
            circuit_rec(S,A,s,tps,couleur,deb,fin,predec)
        elif couleur[s]=="gris":
            print("circuit :",sorg,"-",s)
    couleur[sorg]="noir"
    fin[sorg]=tps[0]
    tps[0]+=1

def circuit_deb(S,A,s0):
    tps,couleur,deb,fin,predec=[0],{},{},{},{}
    for s in S:
        couleur[s]="blanc"
    circuit_rec(S,A,s0,tps,couleur,deb,fin,predec)
```

On observe :

```
>>> circuit(S,A,4)
circuit : 3 - 4
circuit : 2 - 4
circuit : 5 - 2
circuit : 7 - 2
```

2. On saisit :

```
def circuit(S,A):
    tps,couleur,deb,fin,predec=[0],{},{},{},{}
    for s in S:
        couleur[s]="blanc"
    for s in S:
        if couleur[s]=="blanc":
            circuit_rec(S,A,s,tps,couleur,deb,fin,predec)
```

3. Dans le cas d'un graphe non orienté, il faut qu'on exclut le cas où l'on rencontre un sommet en cours de visite depuis un autre sommet en cours de visite qui en est le prédécesseur. On saisit :

```
def cycle_rec(S,A,sorg,tps,couleur,deb,fin,predec):
    deb[sorg]=tps[0]
    couleur[sorg]="gris"
    tps[0]+=1
    for s in A[sorg]:
        if couleur[s]=="blanc":
            predec[s]=sorg
            cycle_rec(S,A,s,tps,couleur,deb,fin,predec)
        elif couleur[s]=="gris" and s!=predec[sorg]:
            print("circuit :",sorg,"-",s)
    couleur[sorg]="noir"
    fin[sorg]=tps[0]
    tps[0]+=1

def cycle(S,A):
    tps,couleur,deb,fin,predec=[0],{},{},{},{}
    for s in S:
        couleur[s]="blanc"
    for s in S:
        if couleur[s]=="blanc":
            cycle_rec(S,A,s,tps,couleur,deb,fin,predec)
```