

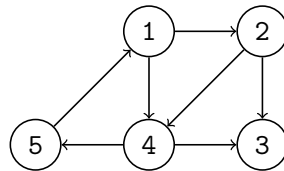
# TD 7 : Graphes

## 1 Calcul des composantes fortement connexes

Rappel : Les composantes fortement connexes d'un graphe sont les classes d'équivalence pour la relation  $i \equiv j \Leftrightarrow i \rightsquigarrow^* j$  et  $j \rightsquigarrow^* i$ .

### 1.1 Algorithme de Kosaraju

QUESTION 1 – Appliquer l'algorithme de Kosaraju au graphe ci-dessous.



### 1.2 Algorithme de Tarjan

#### Fonction CFC

```
date ← 0
Debut ← [0, 0, ..., 0]
P ← PILE_VIDE()
NCFC ← 0
CFC ← [0, 0, ..., 0]
pour tout i ∈ S faire
    si Debut[i] = 0 alors
        TARJAN(i)
renvoyer(CFC)
```

#### Fonction TARJAN(i)

```
date ← date + 1
Debut[i] ← date
min ← Debut[i]
EMPILER(P, i)
pour tout j ∈ Adj[i] faire
    si Debut[j] = 0 alors
        min ← MIN(min, TARJAN(j))
    sinon
        si CFC[j] = 0 alors
            min ← MIN(min, DEBUT[j])
si min = DEBUT(i) alors
    NCFC ← NCFC + 1
    faire
        k ← DEPILER(P)
        CFC[k] ← NCFC
    tant que k ≠ i
renvoyer(min)
```

QUESTION 2 – À l’instar de l’algorithme de Kosaraju, l’algorithme de Tarjan permet de calculer les CFC d’un graphe. Appliquer l’algorithme de Tarjan au graphe  $G_2$ . Quelles différences existe-t-il entre ces deux algorithmes ?

## 2 Tri topologique par élimination de sources

On appelle source tout noeud n’étant la destination d’aucune arête. Proposez une méthode alternative pour réaliser le tri topologique d’un graphe dirigé acyclique en remarquant que les sources correspondent à des tâches devant être effectuées en premier.

## 3 2-colorabilité

Une  $k$ -coloration d’un graphe non-orienté  $G = (S, A)$  est une fonction  $c : S \rightarrow \llbracket 0, k - 1 \rrbracket$  telle que  $c(u) \neq c(v)$  pour toute arête  $(u, v) \in A$ .

QUESTION 3 – Montrer que tout arbre est 2-coloriable.

QUESTION 4 – Proposer un algorithme permettant de décider si un graphe est 2-coloriable.