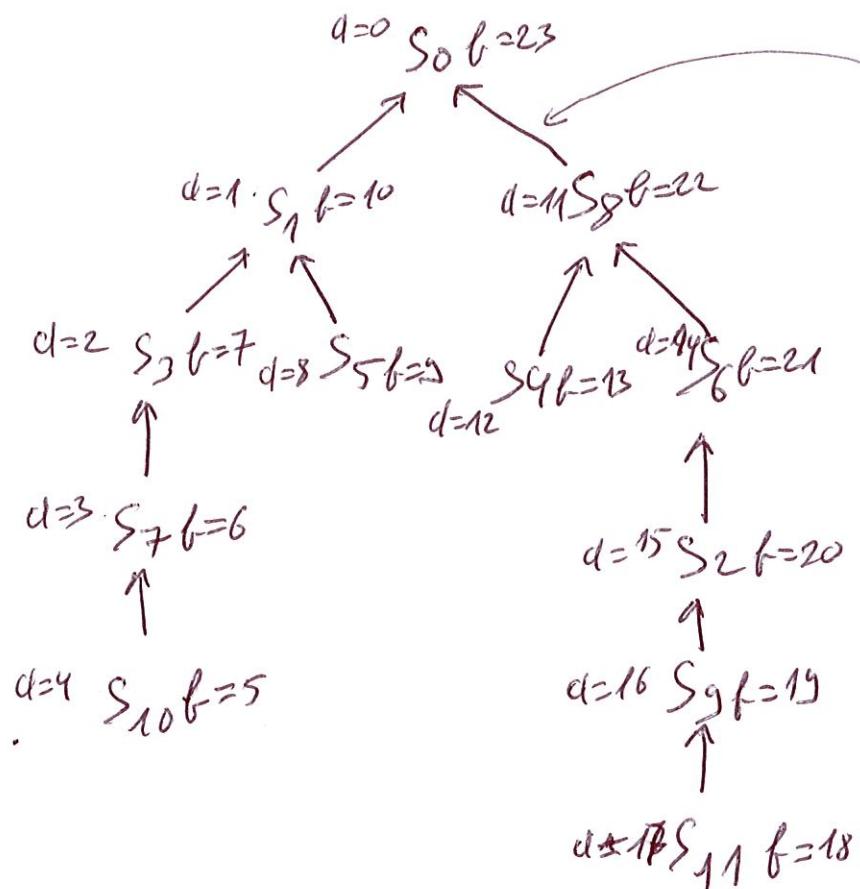


# Concours DS 2016 - 2017

Exercice 1 :



Exercice 2 :

1) Oui, quand le graphe possède un circuit, il n'est pas possible de réaliser les tâches du circuit à cause d'une dépendance circulaire.

Par exemple, dans  $\textcircled{A} \rightarrow \textcircled{B} \rightarrow \textcircled{A}$  la tâche A dépend de B et B celle de A.

①

les graphes dont les tâches ne peuvent pas être réalisées sont les graphes qui possèdent un ~~circuit~~ circuit.

2) Pour obtenir un ordre des tâches à réaliser, il suffit de faire un tri-topologique du graphe :

tri-topologique ( $a$ ) :

PP( $a$ )

retourner les sommets de  $G$  par ordre croissant des  $f$  calculé par PP( $a$ ).

3) Pour détecter si un graphe n'est pas réalisable, il suffit de détecter un circuit dans le graphe. Voici deux façons de procéder =

Méthode 1 = on réalise un tri-topologique du graphe et on vérifie que les arcs, dérivés sur le tri, vont tous de la droite vers la gauche. ②

Dit autrement, on calcule PP( $a$ ) et on vérifie que pour tout arc  $(i, j)$   $b[j] < b[i]$ .

tache\_dnt-realisable ( $G$ ) :

```
PP( $G$ )
pour tout arc  $(i,j)$  de  $G$ 
    si  $f(j) > f(i)$  alors
        retourner FAUX
    fin si
fin pour
retourner VRAI
```

Méthode 2: On modifie le parcours en profondeur pour détecter un circuit. On rencontre un circuit si, dans le parcours en profondeur, on rencontre un sommet gris.

③

tache\_dnt-realisable ( $G$ ):

```
Pour chaque sommet  $u$  de  $G$ 
    couleur [ $u$ ]  $\leftarrow$  blanc
    pere [ $u$ ]  $\leftarrow$  nul
fin pour
temp  $\leftarrow$  0
Pour chaque sommet  $u$  de  $G$  faire
    si couleur [ $u$ ] = BLANC alors
        si !(visite_PP( $u$ )) alors
            retourner FAUX
        fin si
    fin si
fin pour
retourner VRAI
```

Visite\_PP( $u$ ):

```
[couleur [ $u$ ]  $\leftarrow$  gris
d[u]  $\leftarrow$  temps + 1
pour chaque  $V$  de Adj [ $u$ ] faire
    si couleur [ $V$ ] = BLANC
        pere [ $V$ ]  $\leftarrow u$ 
        visite_PP( $V$ )
    fin si
    fin pour
    [couleur [ $u$ ]  $\leftarrow$  GRIS]
    si (couleur [ $V$ ] == GRIS)
        et pere [ $V$ ] !=  $u$ 
    alors
        retourner FAUX
    fin si
fin pour; couleur [ $u$ ]  $\leftarrow$  NOIR]
```

$f(u) \leftarrow \text{temp} + 1$

### Exercice 3 :

Avec la matrice d'adjacence =

fonction  $m \times m$  -> sommet-de-degré-1 ( $A$ ) :

$m = \text{taille}(t)$

pour  $i$  allant de 0 à  $m-1$  faire

$\text{degré} = 0$

    pour  $j$  allant de 0 à  $m-1$  faire

$\text{degré} = \text{degré} + A[i][j]$

    fin pour

    si  $\text{degré} == 1$  alors

        retourner VKT

    fin si

fin pour

retourner FAUX

Avec la liste d'adjacence =

fonction  $m \times m$  -> sommet-de-degré-1 ( $Adj$ ) :

    pour tout  $s$  de  $Adj$  :

$\text{degré} = \text{taille}(Adj[s])$

        si  $\text{degré} == 1$

            retourner VRAC

        fin degré

    fin pour

retourner FAUX

Avec la matrice d'incidence :

fonction - un - permet - de - degrés - 1 (B)

$m = \text{taille}(B)$

si  $n == 0$  alors

    | retourner Flux

fin m

$m = \text{taille}(B[0])$

pour tout i allant de 0 à  $m-1$  faire :

    | degré = 0

    | pour tout e allant de 0 à  $m-1$  faire :

        | degré = degré +  $B[i][e]$

    | fin pour

    | si degré == 1 :

        | retourner Vrai

    | fin m

fin pour

retourner faux

(5)