Devoir 3

BLAIS REGOUT, Lucien
SAVOIE, Olivier

IFT436 - Algoritmes et structures de données

Faculté des Sciences
Université de Sherbrooke

Présenté à Pr BLONDIN, Michael

Jeudi, 10 octobre 2019

Première Question

a)

V = Les bassins du manège aquatique (sommets). ***E*** = Les corridors entre les bassins, potentiellement des glissades(arêtes). ***G*** = Manège aquatique, est dirigé, du à l'élevation. Élévation dénoté de l'énoncé suivant :

> Puisque les bassins sont situés de plus en plus bas, le long d'une colline, si on peut atteindre un bassin j à partir d'un bassin i, alors on ne peut pas atteindre le bassin i à partir du bassin j.

b)

G ne peut pas être un cycle dû contraintes fournis ci-dessous. Grâce à cette première contraites plus bas, il est impossible qu'un cycle soit établi avec un seul bassin considérant qu'il y a aucun corridor qui part d'un dit bassin en arrivant dans ce même bassin. Finalement, la seconde contrainte implique que dû à l'inclinaison, l'un ne peut pas revenir au bassin précédent. Cela implique qu'il y a aucun cycle possible et que de ce fait, le manège aquatique ***G*** est acyclique et dirigé.

> Il n'y a *pas* de corridor passant d'un bassin vers lui-même.

> Puisque les bassins sont situés de plus en plus bas, le long d'une colline, si on peut atteindre un bassin j à partir d'un bassin i, alors on ne peut pas atteindre le bassin i à partir du bassin j.

c)

Si le graphe ***G*** est considéré non dirigé, il y a plusieurs arrangements d'arêtes qui rendrait ***G*** cyclique. Toutefois, il y a toujours la possibilité qu'aucun cycle, simple ou non, ne soit présent et que le nombre d'arête soit équivalent au nombre de sommet -1. De sorte que les propriétés suivantes soit respectées.

— ***G*** est *connexe*
— ***G*** est *acyclique*
— $|E| = |V|$ **YALL C'EST PAS BON!!!!!**

d)

$$\forall n \geq 1$$
$$min = n - 1$$
$$max = \sum_{i=1}^{n} n - i = \sum_{i=1}^{n} n - \sum_{i=1}^{n} i = n^2 - \frac{n}{2}(n+1) = \frac{n^2 - n}{2}$$

e)

Bassin départ : De toutes les paires de sommets composants les arêtes, le sommet qui se retrouvera uniquement en index 0 de la paire, à la gauche, qui est donc toujours le sommet de départ, sera donc considéré comme sommet de départ.

Bassin d'arrivée : Suivre le même résonnement, mais pour le sommet qui se retrouve toujours en index 1 de la paire et qui est donc toujours le bassin d'arrivée dans toutes les paires, sera donc considéré comme sommet d'arrivée.

f)

Algorithme 1 : Calcul de la séquence qui possède un temps maximal passé dans une attraction ***G***

Entrées : Manège ***G*** tel que $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$ et sommet initial u tel que u appartien à $\boldsymbol{V}$.

Résultat : Une séquence S de bassins de temps maximal dans le manège ***G***.

$S \leftarrow []$
$Ttot \leftarrow 0$
$Parent \leftarrow u$
`parcours` (x) **:**
    $SPrime \leftarrow S$
    $Temps \leftarrow Ttot$
    si x *non marquée* **alors**
        $Temps \leftarrow Temps + c[Parent, x]$
        marquer x
        ajouter x à $SPrime$
        si $Temps > Ttot$ **alors**
            $Ttot \leftarrow Temps$
            $S \leftarrow SPrime$
        pour $u \in \boldsymbol{V} : x \rightarrow y$ **faire**
            $Parent \leftarrow x$
            `parcours` (y)
`parcours` (u) **retourner** S

g)

Algorithme 2 : Calcul de la séquence qui possède un temps maximal passé dans une attraction $\boldsymbol{G}$, donné 5 remontées

Entrées : Manège $\boldsymbol{G}$ tel que $\boldsymbol{G} = (\boldsymbol{V},\boldsymbol{E})$, que le sommet initial u tel que $u \in \boldsymbol{V}$, que $v \in \boldsymbol{V}$, ainsi que v est le bassin finale.

Résultat : Une séquence S de bassins, de temps maximal dans le manège $\boldsymbol{G}$, considérant que l'utilisateur peut remonter au bassin initial un maximum de 5 fois.

$S \leftarrow [\,]$
$Ttot \leftarrow 0$
$Parent \leftarrow u$
$Ascension \leftarrow 5$
`parcours` *(x)* **:**
    $SPrime \leftarrow S$
    $Temps \leftarrow Ttot$
    si x *non marquée* **alors**
        $Temps \leftarrow Temps + c[Parent, x]$
        marquer x
        ajouter x à $SPrime$
        si $Temps > Ttot$ **alors**
            $Ttot \leftarrow Temps$
            $S \leftarrow SPrime$
        si *prochain voisin de* $x = v$ **alors**
            $Ascension \leftarrow Acension - 1$
            retirer les marqueurs
            $Parent \leftarrow S[0]$
            `parcours` $(Parent)$
        sinon
            pour $u \in \boldsymbol{V} : x \rightarrow y$ **faire**
                $Parent \leftarrow x$
                `parcours` (y)
`parcours` (u) **retourner** S

Ici, il fera exactement comme l'algorithme précédent, à l'exception que lorsqu'il se rendra au plus long parcours précédent le bassin final, et se 5 fois. La 5e fois, le sera possiblement différent mais restera cependant couvers par l'algorithme.

Seconde Question

a)

Algorithme 3 : Tri d'une séquence et calcul des valeurs modales

Entrées : Une séquence S non vide d'éléments comparables.

Résultat : Les valeurs modales$m$ trouvées dans la séquence S

```
mode ← [ ]
modeQty ← 0
trouverMode (x) :
    si |x| = 0 alors
        retourner mode
    sinon
        pivot ← mediane(x)
        gauche ← [x ∈ S : x < pivot]
        centre ← [x ∈ S : x = pivot]
        droite ← [x ∈ S : x > pivot]
        si |centre| = modeQty alors
            ajouter pivot à mode
        si |centre| > modeQty alors
            modeQty ← |centre|
            mode ← [pivot]
        sinon
            trouverMode(gauche)
            trouverMode(droite)
retourner trouverMode(S)
```

b)