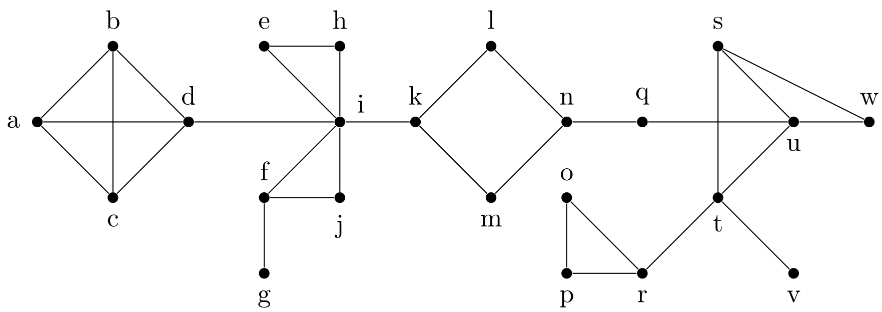
Algorithmique II
Année 2007-2008

Jean-François Pineau
Jean-Baptiste Rouquier

TD 4
Recouvrement d'arbre

1 Et les coudes, et les genoux, et les chevilles



Soit $G = (V, E)$ un graphe non orienté connexe. Un *point d'articulation* de G est un sommet dont la suppression déconnecte le graphe. Un *isthme* est une arête dont la suppression déconnecte le graphe. Une *composante 2-connexe* de G est un ensemble maximal connexe de sommets tel que le sous-graphe induit par ces sommets n'a pas de point d'articulation.

On notera $T = (V, F)$ un arbre de parcours de G obtenu par parcours en profondeur et $deb(v)$ le début de visite de v dans la numérotation associée à ce parcours en profondeur.

Question 1 Sur l'exemple ci-dessus, déterminer les points d'articulation, les isthmes et les composantes 2-connexes.

Question 2 Expliquer pourquoi dans le parcours en profondeur d'un graphe non orienté et connexe on n'obtient que deux sortes d'arcs : les arcs de l'arbre et les arcs retours.

Question 3 Montrer que la racine de T est un point d'articulation si et seulement si elle a au moins deux fils dans T.

Question 4 Dans toute la suite, « descendant » est pris au sens large : x est un descendant de x (contrairement aux conventions habituelles).

Soit x un sommet de T distinct de la racine. Prouver que x est un point d'articulation si et seulement s'il existe un fils y de x dans l'arbre tel qu'il n'existe pas d'arc retour de la forme (z, t) où z est un descendant de y et t un ancêtre propre de x.

Question 5 On définit la fonction `au_dessous` de V dans $\mathbb{N}$ par :

$$\begin{aligned} \texttt{au\_dessous}(x) &:= \min(\{deb(x)\} \cup A_x) \\ A_x &:= \{deb(z) \mid (y, z) \text{ est un arc retour avec } y \text{ descendant de } x\} \end{aligned}$$

Donner un algorithme en $\mathcal{O}(n + m)$ de calcul de la fonction `au_dessous`. On prouvera la complexité.

Question 6 Montrer comment calculer tous les points d'articulation en $\mathcal{O}(n + m)$.

Question 7 Prouver qu'une arête est un isthme si et seulement si elle n'appartient à aucun cycle élémentaire.

Question 8 Montrer comment calculer les isthmes en $\mathcal{O}(n+m)$.

Question 9 Considérer le graphe H dont les sommets sont les composantes 2-connexes de G et tel que deux composantes sont adjacentes si et seulement si elles ont au moins un sommet en commun. Que pouvez vous dire de ce graphe ?

Ceci explique pourquoi on n'a pas défini « composante 2 connexe » par le fait qu'il faille couper deux arêtes pour déconnecter le graphe.

2 L'hiver, il faut se couvrir

Exercice 1 Soit $T = (X, E)$ un arbre, on appelle *ensemble dominant* de T une partie $S \subseteq X$ vérifiant :

$$\forall x \in X \quad \exists y \in S \quad x = y \text{ ou } xy \in E$$

On cherche un ensemble dominant de cardinal minimum.

1. Pour $|T| \geqslant 3$, montrer qu'il existe toujours un ensemble dominant de cardinal minimum ne contenant aucune feuille de T.
2. Un élève imagine l'algorithme suivant.

   $S \leftarrow \varnothing, F \leftarrow T$
   while F *contient un sommet* x *de degré 1* **do**
       $y = \Gamma(x)$
       $S \leftarrow S \cup \{y\}$
       $F \leftarrow F - \Gamma(y) - \{y\}$
   if $F \neq \varnothing$ **then**
       ajouter tous les sommets restants (isolés) de F à S
   return S

   Montrer que cet algorithme ne fournit pas toujours un ensemble dominant de cardinal minimum.
3. Décrire un algorithme le plus efficace possible pour calculer un ensemble dominant S de cardinal minimum. (*Note : On peut le faire en* $O(n)$).

Exercice 2 (non corrigé)

Dans cet exercice, $G = (X, E)$ est un graphe connexe dont les arêtes sont munies d'une valuation p strictement positive, et $T = (X, F)$ est un arbre couvrant de poids minimum. Le but ici sera de calculer le second arbre couvrant de poids minimum T'.

1. Montrer qu'il existe un T' de la forme $T - \{e\} + \{f\}$, avec $e \in F, f \in E \backslash F$.
2. Étant donné un sommet x fixé, donner un algorithme en $O(n)$ calculant pour tout $y \in X - \{x\}$ l'arête zt de poids maximum sur l'unique chaîne de T reliant x à y. Le résultat sera stocké dans un tableau de taille n, arête[y] contiendra zt. Justifier la complexité de l'algorithme.
3. On appelle T_x un arbre recouvrant de plus petit poids de la forme $T_x = T - \{e\} + \{f\}$, où $e \in F$ et $f = xy \in E \backslash F$. Donner un algorithme en $O(n)$ calculant T_x et justifier la complexité.
4. En déduire un algorithme calculant T'. Quelle est la complexité ?