

M1 Bio-Info – Examen d’algorithmique

3 avril 2007

Durée : 2 heures. Documents autorisés : notes de cours et TD. Le barème est indicatif.

Dans toutes les questions on supposera que les graphes sont sans boucles et sans arcs multiples.

Question 1 (5 points)

1. Quel type de parcours est-il préférable d’utiliser pour savoir rapidement si un graphe orienté contient des cycles ? Justifiez votre réponse.
2. Même question que précédemment, si on considère un graphe non-orienté.
3. On fixe un sommet $s \in V$ dans un graphe orienté $G = (V, E)$ et on demande de trouver un cycle qui contient le sommet s . Expliquez comment on peut utiliser le parcours en largeur (BFS) pour répondre à cette question.

Question 2 (6 points)

L’entrée pour ce problème est la matrice d’adjacence $A \in \{0, 1\}^{n \times n}$ d’un graphe non-orienté G , dont l’ensemble de sommets est $\{1, \dots, n\}$. La procédure **CompConn** ci-dessous devra être complétée afin de calculer les composantes connexes de G . Autrement dit, on demande qu’à la fin de **CompConn** on ait $A[i, j] = 1$ ssi i et j sont connectés par un chemin.

Procédure **CompConn** ($A[1..n, 1..n]$ matrice d’adjacence)

```
debut
  t = log(n) // logarithme en base 2
  repeter t fois
    pour i de 1 a n faire
      pour j de 1 a n faire
        k = 1
        tant que (...) faire
          A[i,j] = A[i,k] et A[k,j] // 0 = faux, 1 = vrai
          A[j,i] = A[i,j]
          k = k + 1
        fin tantque
      fin pour
    fin pour
  fin repeter
fin
```

La procédure `CompConn` devra respecter l'invariant suivant : après k itérations de la boucle `répéter`, $A[i, j] = 1$ s'il existe un chemin de i à j de longueur au plus 2^k .

1. Complétez la condition de la boucle `tantque` pour que `CompConn` soit correcte (pensez aussi à la terminaison).
2. Trouvez un exemple qui montre qu'on ne peut pas choisir la valeur 1 pour t dans la première ligne du programme.
3. Indiquez le nombre minimal d'exécutions de la boucle `répéter` afin d'obtenir le bon résultat sur le graphe-ligne suivant : $V = \{1, \dots, 8\}$, $E = \{12, 23, 34, \dots, 78\}$. Justifiez votre réponse.

Question 3 (3 points)

L'algorithme de Dijkstra calcule les chemins de poids minimal en supposant que les poids sont positifs. Donnez un exemple qui montre que l'algorithme ne donne pas la réponse correcte si les poids ne sont pas positifs.

Question 4 (6 points)

1. Construisez l'automate fini qui correspond au motif $P = GCAGAGAG$ (sur l'alphabet $\{A, C, G, T\}$).
2. Exécutez l'algorithme basé sur l'automate ci-dessus sur le texte suivant :

$T = GCATCGCAGAGAGTATACAGTACG$