TD Algo2 – session 12 – Graphes orientés

28 mai 2025

Objectifs d'apprentissage :
— trouver des graphes orientés présentant des propriétés particulières ;
— concevoir des algorithmes relatifs aux graphes orientés.

Exercice 1 : Énumération

Combien y a t-il de graphes orientés avec 2 sommets et sans label sur les sommets ?

Exercice 2 : Mauvais arcs

Lorsqu'on exécute TRI-TOPOLOGIQUE(G) sur un graphe orienté G contenant au moins un circuit, on pourrait espérer qu'il s'agit d'un ordre sur les sommets qui minimise le nombre de "mauvais" arcs, c'est-à-dire avec le moins possible d'arcs dans le mauvais sens. Trouver un exemple qui montre que ce n'est pas le cas.

Exercice 3 : Nombre de chemins

Concevoir un algorithme en temps $O(V+E)$ qui prend en entrée un graphe orienté sans circuit $G=(V,E)$ et deux sommets s et t, puis qui retourne le nombre de chemins allant de s à t dans G. L'algorithme n'est pas obligé d'énumérer les chemins, il suffit de les compter.

Exercice 4 : Chemin critique

Pour la planification de projets, on cherche à déterminer le chemin critique, c'est-à-dire le plus long chemin d'un graphe orienté sans circuit. Par ailleurs, la représentation des tâches par des arcs va un peu contre l'intuition. Il serait plus naturel que les sommets représentent les tâches et que les arcs représentent les contraintes de séquencement ; autrement dit, l'arc (u,v) indiquerait qu'il faut faire la tâche u avant la tâche v. Les poids w seraient alors affectés aux sommets, et non aux arcs. Modifier la procédure PCC-GSS pour qu'elle trouve en temps $O(V+E)$ la longueur d'un plus long chemin dans un graphe orienté sans circuit à sommets pondérés.

Exercice 5 : Circuit hamiltonien

Concevoir un algorithme en temps $O(V+E)$ qui détermine s'il existe un chemin qui passe par chaque sommet exactement une fois dans un graphe orienté sans circuit.

Exercice 6 : Tri topologique itératif

Un autre moyen d'effectuer le tri topologique d'un graphe orienté sans circuit $G=(V,E)$ consiste à faire, de manière itérative, les opérations suivantes : trouver un sommet de degré entrant 0, l'imprimer, puis le supprimer du graphe ainsi que tous les arcs qui en partent. Expliquer comment implémenter cette idée par un algorithme à temps $O(V+E)$. Qu'advient-il de cet algorithme si G contient des circuits ?

Exercice 7 : Tri topologique unique

Concevoir un algorithme qui détermine si un graphe orienté sans circuit possède un tri topologique unique.

Exercice 8 : Plus court circuit

Concevoir un algorithme en temps $O(VE)$ pour trouver un circuit avec le plus petit nombre d'arcs dans un graphe orienté (ou indiquer que le graphe est sans circuit).

Exercice 9 : Accessibilité

Concevoir un algorithme en temps $O(V+E)$ qui détermine si un graphe orienté sans circuit possède un sommet accessible à partir de tous les autres.

Exercice 10 : Chemin de longueur donnée

Concevoir un algorithme qui détermine s'il existe un chemin de longueur L entre deux sommets s et t dans un graphe orienté sans circuit.