

IF111 - Algorithmes et structures de données

EI5 - Algorithmes gloutons et arbres couvrants

Rohan Fossé

rfosse@labri.fr

Plus grande valeur

On cherche à sélectionner cinq nombres de la liste suivante en cherchant à avoir leur somme la plus grande possible (maximiser une grandeur) et en s'interdisant de choisir deux nombres voisins (contrainte).

15 - 4 - 20 - 17 - 11 - 8 - 11 - 16 - 7 - 14 - 2 - 7 - 5 - 17 - 19 - 18 - 4 - 5 - 13 - 8

Comme on souhaite avoir le plus grand résultat final, la stratégie gloutonne consiste à choisir à chaque étape le plus grand nombre possible dans les choix restants.

1. Appliquez cet algorithme glouton sur le tableau;
2. Vérifiez que 20,18,17,16,15 est une autre solution possible.
3. Que dire de la solution gloutonne ?

Problème du voyageur de commerce

Un voyageur a ciblé plusieurs villes qu'il souhaite visiter. Il cherche un itinéraire passant par toutes ces villes et qui minimise la distance totale parcourue. Les villes peuvent être visitées dans n'importe quel ordre mais aucune ne doit être négligée, et le visiteur doit revenir à la fin à sa ville de départ.

Le voyageur part de Nancy et souhaite visiter Metz, Paris, Reims et Troyes, avant de retourner à Nancy.

Voici un tableau donnant les distances kilométriques entre chacune des ces villes.

	Nancy	Metz	Paris	Reims	Troyes
Nancy		55	303	188	183
Metz	55		306	176	203
Paris	303	306		142	153
Reims	188	176	142		123
Troyes	183	203	153	123	

1. Quelle est la stratégie gloutonne à mettre en oeuvre ?
2. Mettez en oeuvre cette stratégie et donnez la solution.
3. Calculez la distance totale pour le parcours Metz - Reims - Paris - Troyes (départ et arrivée à Nancy sous-entendus)
4. Que dire de la solution gloutonne ?

Le parc d'attraction

Vous visitez un parc d'attractions proposant des spectacles à différents horaires. Voici les horaires des différents spectacles :

spectacle	A	B	C	D	E	F	G
horaire	10h-11h	10h30-11h30	11h-12h30	11h30-12h	12h-13h	13h-15h	13h30-14h

H	I	J
14h-15h30	15h-16h	16h-17h30

Vous avez remarqué qu'il n'est pas possible d'assister à tous les spectacles puisque certains ont lieu à des moments communs. Vous souhaitez assister à un maximum de spectacles sur la journée. Quels spectacles devez-vous choisir ?

1. Trouvez deux algorithmes gloutons pour ce problème
2. Appliquez ces deux stratégies au problème.
3. Laquelle donne la meilleure solution ?

La clé USB

Nous disposons d'une clé USB qui est déjà bien remplie et sur laquelle il ne reste que 5 Go de libre. Nous souhaitons copier sur cette clé des fichiers vidéos pour l'emporter en voyage. Chaque fichier a un poids et chaque vidéo a une durée. La durée n'est pas proportionnelle à la taille car les fichiers sont de format différents, certaines vidéos sont de grande qualité, d'autres sont très compressées. Le tableau qui suit présente les 7 fichiers disponibles avec les durées données en minutes.

Nom	Durée en min (valeur)	Poids
Vidéo A	114	4.57 Go
Vidéo B	32	630 Mo
Vidéo C	20	1.65 Go
Vidéo D	4	85 Mo
Vidéo E	18	2,15 Go
Vidéo F	80	2,71 Go
Vidéo G	5	320 Mo

Quelles vidéos copier sur la clé USB pour que la durée des vidéos soient la plus grande possible tout en ne dépassant pas 5 Go ? Donner l'algorithme glouton.

Bipartition

Étant donné un ensemble de n nombres, répartissez les en 2 sous-ensembles tels que la somme des éléments du premier soit égal à la somme des éléments du second. Proposez des algorithmes gloutons pour résoudre ce problème. Appliquez les aux ensembles suivants :

$$\{2, 10, 3, 8, 5, 7, 9, 5, 3, 2\}$$
$$\{771, 121, 281, 854, 885, 734, 486, 1003, 83, 62\}$$

Arbres non-isomorphes

Donner tous les arbres non-isomorphes à :

1. 1, 2 ou 3 sommets

2. 4 sommets
3. 5 sommets

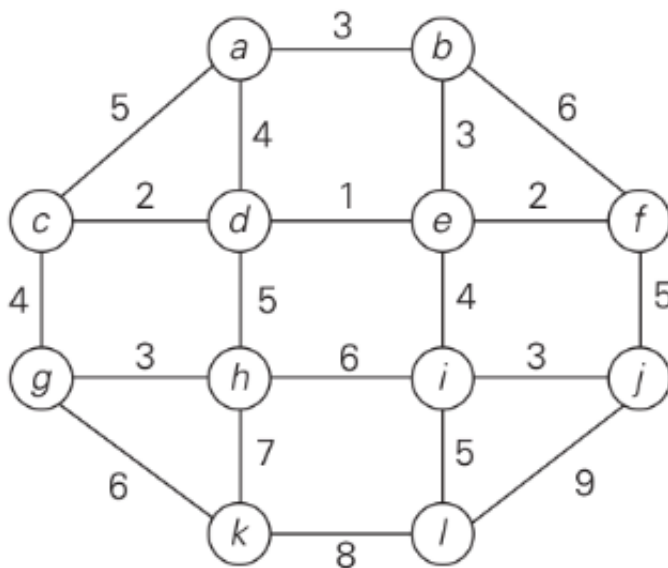
Arbres couvrants

Soit $G = (S, A)$ un graphe non orienté (pas forcément connexe). on appelle forêt couvrante maximale tout sous-graphe de G couvrant, sans cycle, et telle que l'ajout d'une arête quelconque crée un cycle.

1. Montrer que si G est connexe, toute forêt couvrante maximale est un arbre couvrant.
2. Montrer que si le nombre d'arêtes d'une forêt couvrante est $|S| - k$ où k est le nombre de composantes connexes de G .

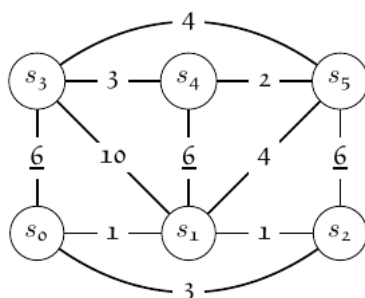
Algorithme de Prim

Appliquer l'algorithme de Prim pour trouver un arbre couvrant de poids minimum du graphe G représenté sur la figure ci-dessous.



Algorithme de Kruskal et Prim

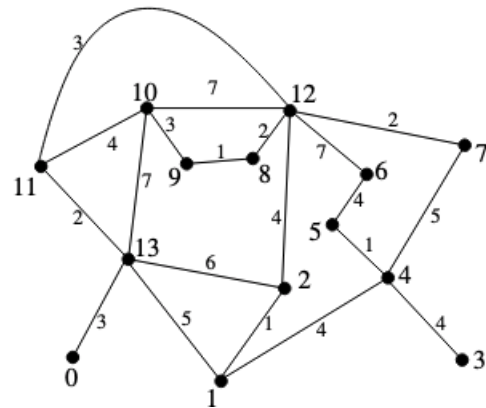
On considère le graphe non orienté valué G suivant:



Déterminer un arbre recouvrant de G de poids minimum depuis le sommet s_0 en utilisant l'algorithme de Kruskal. Déterminer l'arbre couvrant de G en utilisant l'algorithme de Pri

Algorithme de Kruskal

En détaillant toutes les étapes, appliquez l'algorithme de Kruskal sur le graphe suivant.



Quel est le poids d'un arbre couvrant de poids minimal ?

En utilisant le même graphe que la question précédente, déterminer comment calculer le poids d'un arbre couvrant de poids maximum. Donner un arbre de poids couvrant maximal.