

Feuille d'exercices no. 8

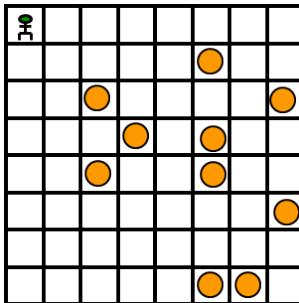
Algorithmes gloutons

Exercice 1. Stations-service

Vous partez de chez vous en voiture, par l'autoroute, pour aller à une destination précise. On suppose que votre réservoir d'essence vous permet de parcourir n kilomètres après un plein. Vous avez une carte indiquant toutes les stations-service de votre trajet ainsi que les distances entre votre domicile et la première station, entre des stations successives, et entre la dernière station et votre destination. On suppose que ces distances sont toutes inférieures à n . Au départ le réservoir est plein.

1. Décrivez en français un algorithme efficace qui minimise le nombre d'arrêts à faire dans les stations-service.
2. Justifiez que votre stratégie est optimale.
3. Écrire l'algorithme en pseudo-code.

Exercice 2. Un grille carrée a des pièces de monnaie sur certaines cases. Un robot situé dans le coin en haut à gauche se déplace vers le coin diamétralement opposé. Il ne peut se déplacer que dans deux directions, à droite et vers le bas. Quand il est sur une case, il récupère la pièce qui est sur la case. Exemple :



On veut récupérer toutes les pièces en utilisant un nombre minimum de robots.

- Proposer un algorithme glouton pour cela.
- Est-ce que votre algorithme trouve une solution optimale ? Justifier.

Exercice 3. On souhaiterait enregistrer sur une mémoire de taille L un groupe de fichiers $P = (P_1, P_2, \dots, P_n)$. Chaque fichier P_i nécessite une place a_i . Supposons que $\sum_{i=1}^n a_i > L$: on ne peut pas enregistrer tous les fichiers. On a donc besoin de choisir le sous-ensemble Q des fichiers à enregistrer, et cela peut être fait selon plusieurs critères.

Premier critère. On pourrait souhaiter le sous-ensemble qui contient le plus grand nombre de fichiers.

- a) Donner un algorithme glouton pour résoudre de manière exacte cette variante du problème.
- b) Quelle est sa complexité ? Justifier.
- c) Montrer que votre algorithme résout effectivement correctement le problème, par une preuve telle que celle vue en cours. Sinon, apporter quelques arguments pour justifier intuitivement la correction de l'algorithme.
- d) Soit Q le sous-ensemble obtenu. Bien évidemment, le sous-ensemble Q va occuper une partie de l'espace de taille L , et cette partie peut être plus ou moins importante en fonction des données en entrée. Si, sur une certaine donnée, très peu de place est occupée à la fin de l'algorithme, on peut penser que le critère considéré n'est pas très bon. Appelons *quotient d'utilisation* de Q par rapport à L la quantité

$$\text{Quotient}(Q, L) = \frac{\sum_{P_i \in Q} a_i}{L}.$$

Quelle est la valeur minimum possible du quotient d'utilisation à la fin de l'algorithme ? Est-ce qu'elle peut être de $1/3, 1/4, \dots, 1/k$ avec k entier positif arbitraire ? Justifier.

Deuxième critère. Supposons maintenant que l'on souhaite enregistrer le sous-ensemble Q de P qui maximise le quotient d'utilisation, c'est-à-dire celui qui remplit le plus de disque.

- e) Proposer un algorithme glouton pour cette nouvelle variante du problème.
- f) Quelle est la complexité de votre algorithme ? Justifier.
- g) Est-ce que votre algorithme fournit toujours la solution optimale, c'est-à-dire le sous-ensemble Q qui maximise le quotient d'utilisation ? Justifier.

Exercice 4. Soit un carrefour routier pour cinq routes notées 1, 2, 3, 4, 5 dans le sens des aiguilles d'une montre. La route 1 est en sens unique arrivant dans le carrefour, la route 3 est en sens unique partant du carrefour et les autres routes sont en double sens. Nous supposons que les routes sont assez larges à l'approche du carrefour pour prévoir jusqu'à cinq voies d'attente aux feux (chaque voie ayant son propre feu).

On souhaite mettre des feux dans le carrefour, qui permettent une circulation sans danger, mais qui minimisent aussi le temps passé dans le carrefour.

- Pour que la circulation soit sans danger, on va permettre le passage en même temps (c'est à dire dans le même *créneau* défini par les feux) de deux voitures venant de routes différentes seulement si les trajets empruntés sont complètement disjoints (arrivées dans le carrefour différentes, sorties de carrefour différentes et trajets qui ne se croisent pas). Deux voitures venant de la même route peuvent passer sans danger durant le même créneau.
- Pour minimiser le temps passé dans le carrefour, il faut minimiser le nombre de créneaux, c'est à dire le nombre de fois que la configuration des feux change avant de revenir à une configuration précédente.

On demande :

1. Proposer une modélisation du problème en identifiant les trajets possibles et les contraintes entre eux.
2. Proposer un algorithme glouton pour obtenir une solution au problème.
3. Exécuter l'algorithme sur l'exemple proposé.
4. Est-ce que votre algorithme est exact (en supposant que vous l'appliquez sur n'importe quel carrefour) ? Justifier.

Pour aller plus loin ...

Exercice 5. Rendu de monnaie

On considère le problème consistant à rendre n centimes en utilisant le moins de pièces possibles.

1. Décrire un algorithme glouton optimal permettant de rendre la monnaie en utilisant des pièces de 50, 20, 10, 5 et 1 centimes. Justifier le fait que l'algorithme aboutit à une solution optimale. Écrire l'algorithme proposé en pseudo-code et calculer sa complexité.
2. Donner un ensemble de valeurs de pièces pour lesquelles l'algorithme glouton ne donne pas une solution optimale.