

Projet algorithmique

Le Rush hour

25 Octobre 2010

Ce projet est à faire en binome et devra être rendu au chargé de Td le 25/11/2011

Présentation du problème

Le plateau de jeu est une grille de taille $N \times M$ représentant un parking (voir figure 1). On y trouve plusieurs types de véhicules de différentes longueurs, de différents poids mais tous larges d'une case. Par exemple :

- Les voitures sont longues de 2 cases,
- Les camionnettes sont longues de 3 cases

Le parking n'a qu'une seule sortie large d'une case et qui se situe à l'est. Le but du jeu est de faire sortir une voiture particulière : le véhicule 1. Chaque véhicule est orienté dans une des deux directions (horizontale ou verticale) et ne peut se déplacer qu'en ligne droite (en marche avant ou en marche arrière). Chaque véhicule reste donc toujours sur une même ligne si sa direction est horizontale ou sur une même colonne si sa direction est verticale. Le véhicule 1 est donc placé sur la ligne de la sortie mais il est bloqué par d'autres véhicules qu'il faut déplacer.

Le codage du jeu

Les véhicules sont numérotés de 1 à m . Les caractéristiques du i^{me} véhicule ($1 \leq i \leq m$) sont données par un quadruplet $(l_i, d_i, \text{num}_i, p_i)$ où

- ▷ l_i est sa longueur (en cases)
- ▷ d_i sa direction
 - h pour horizontale
 - v pour verticale
- ▷ num_i son numéro de ligne ou colonne le long de laquelle il peut se déplacer
- ▷ p_i son poids (en tonnes)

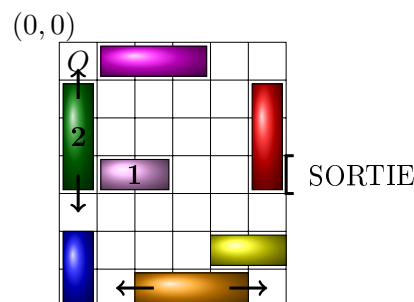


FIGURE 1 – Exemple

La case de coordonnées $(0,0)$ (noté O dans la figure 1) se situe en haut à gauche (comme pour les matrices).

La position du i^{me} véhicule est codée par un entier \mathbf{pos}_i . Si le véhicule est orienté horizontalement \mathbf{pos}_i est la coordonnée du point le plus à droite du véhicule et $\mathbf{pos}_i \in \{l_i - 1, \dots, M - 1\}$. Si le véhicule est orienté verticalement \mathbf{pos}_i est la coordonnée du point le plus en bas du véhicule $\mathbf{pos}_i \in \{l_i - 1, \dots, N - 1\}$.

Dans l'exemple de la figure 1, la voiture $\mathbf{pos}_1=2$ et $\mathbf{pos}_2=3$.

La liste ordonnée des positions des véhicules de 1 à m à un moment donné est appelée **configuration**. Nous contraignons l'orientation de la voiture 1 en position horizontale face à la sortie. C'est-à-dire que la sortie du parking est donnée par la ligne de la voiture 1.

– Certaines configurations sont impossibles à réaliser, donnez en un exemple.

Un déplacement d'un véhicule est codé par un couple (i, s) où i est l'indice du véhicule à déplacer et s est le sens dans lequel on déplace le véhicule $s \in \{N, S, E, O\}$ (*respectivement* Nord, Sud, Est, Ouest). Certains déplacements sont impossibles, soit parce que le sens n'est pas compatible avec la direction du véhicule soit parce que le véhicule rencontre un obstacle.

Par exemple, dans la figure 1, le déplacement $(1, O)$ est impossible.

Les véhicules se déplacent en émettant plus ou moins de CO^2 . Le déplacement d'un véhicule de k tonnes émet k kilogrammes de CO^2 . On introduit le **graphe des configurations** dont les sommets sont les configurations, la source est la configuration initiale et les arêtes correspondent aux déplacements pondérés par le poids du véhicule.

Le but du programme

L'objectif de ce projet est d'écrire un programme en C qui donne les déplacements à faire pour présenter la voiture 1 devant la sortie. Ce programme devra vérifier une des deux options suivantes :

- ▷ faire le moins de déplacements possibles, notée : "option 0"
- ▷ avoir une émission de CO^2 minimale notée : "option 1"

Détails techniques

Vous pouvez utiliser les structures de données des bibliothèques standards fournies par C (C ANSI).

Votre programme prendra en entrée un fichier texte (dont le nom sera passé en argument).

Ce fichier texte devra contenir les informations suivantes :

- ▷ Les dimensions du parking N et M ,
- ▷ Le nombre de véhicules,
- ▷ Les quadruplets caractéristiques des véhicules,
- ▷ La configuration initiale des véhicules.

En plus du nom du fichier, votre programme prendra en entrée un entier qui vaut 0 ou 1 selon l'option et devra afficher en fonction des cas :

- ▷ Si la configuration initiale est incorrecte : "configuration initiale incorrecte" (*par exemple : la voiture 1 est à la position verticale*).
- ▷ S'il n'y a pas de solution au problème : "impossible de sortir !"
- ▷ S'il y a une solution (sur l'exemple donné plus haut)
 - Si l'option est 0 : "Le nombre minimale de déplacements pour présenter devant la sortie la voiture 1 est **nombre minimum de déplacements** , ce minimum est réalisé par la suite de coups $(i_1, s_1), (i_2, s_2) \dots$
 - Si l'option est 1 : "L'émission minimale de CO^2 pour présenter la voiture 1 devant la sortie est de **X kgs**, ce minimum est réalisé par la suite de coups $(i_1, s_1), (i_2, s_2) \dots$ "

Prenons un exemple : Soit un fichier contenant :

```
4 3
2

2 h 1 1
2 v 2 3

1
2
```

Ce fichier permet de coder le parking de taille 4×3 et les deux véhicules 1 et 2. La configuration de départ (*ie.* le sommet source) ressemble à

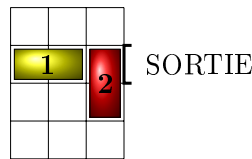


FIGURE 2 – Configuration de départ de l'exemple ci dessus

Le programme affichera

- Pour l'option 0 : "Le nombre minimale de déplacements pour présenter devant la sortie la voiture 1 est 2 , ce minimum est réalisé par la suite de coups (2,S)(1,E)."
- Pour l'option 1 : "L'émission minimale de CO^2 pour présenter la voiture 1 devant la sortie est de 4 kgs, ce minimum est réalisé par la suite de coups (2,S)(1,E)."

Remise du projet

Rapport

Vous devez rédiger un rapport de 5 pages maximum. Ce rapport devra contenir (de façon non exhaustive) les solutions que vous proposez pour traiter ce problème, les méthodes utilisées, l'analyse des résultats que vous aurez obtenu. Ce rapport devra obligatoirement contenir des exemples d'applications de votre algorithme (Au moins un exemple par cas de figure est demandé).

modalité de remise du rapport

1. Votre rapport (en format papier) devra être remis à votre chargé de Td ou à défaut au Secrétariat au plus tard le **25/11/2011**. Ce délai dépassé, vous perdez 1 point par jour de retard.
2. Vous devrez marquer les deux noms constituant le binome ainsi que votre groupe.

Code

Vous devez envoyer **une seule archive** au format **ZIP** (aucune autre ne sera acceptée ; ni tar.gz, ni rar, ni autre) à votre chargé de TD au plus tard le **25/11/2011**.

L'archive doit porter le **nom** NumeroGroupe.NomEtudiant1-NomEtudiant2.zip.

Si les deux étudiants font partie de deux groupes différents, l'archive doit porter le **nom** NumeroGroupe.NomEtudiant1-NumeroGroupe.NomEtudiant2.zip

Votre archive doit contenir

- Le rapport en format **PDF**
- Vos **programmes commentés** avec **makefile** et **readme** si nécessaire.
- Vos codes sources en **.c** et **.h** et **.txt**
- Les fichiers d'exemples **.txt**
- le sujet du mail doit être impérativement **Rush hour**

Si vous ne respectez pas ces critères, nous considérons que le code n'est pas rendu.

Avertissement

Copier entrainera immédiatement le 0