

Rapport du projet Metro

TAILY Pierrick 21600098– GAIZI Hasnae 21807397 – AKROUNE Feriel 21807878

06 mai 2020

1 Implémentation

On a choisit d'utiliser le langage C afin de réaliser le projet.

2 Démarche du projet

-Introduire deux stations de métro différentes -Vérifier que les stations font partis de celles existantes -Chercher tous les chemins possibles qui existent - Définir le chemin le plus court parmi les autres chemins -Afficher le trajet le plus optimal .

3 Structures de données

les structures ont été déclarées dans le fichier *graph.h*.

elem Liste toutes les stations du réseau .

SUBWAY contient le graphe qui représente le réseau parisien .

GRAPH Un tableau avec 2 stations à choisir par l'utilisateur .

nb_{point} : nombre total des stations.

** station : tableau des stations (nom et num de ligne).*

***system : tableau 2d (tmp du trajet pour chaque chemin).*

POINT Représente le sommet

Contient des informations concernant la station

nom_{station} : Son nom.

number_{station} : Son id.

number_{link} : Son numéro dans le réseau de métro.

PATH Représente le chemin entre deux stations terminal : la station du terminus (son numéro) .
timeSec : le temps en seconde du chemin .

DIJKSTRA Calcul et stock le plus court chemin possible
link_{start} : iddestationdedépart.
link_{end} : iddestationd'arrivée.
duration : duréeentrestationdedépartetstationd'arrivée.

4 Principale procédure : Algorithme Dijkstra

L'implémentation de l'algorithme de Dijkstra a été réalisée avec un tas binaire pour obtenir une complexité logarithmique. Le tas binaire est stocké sous forme de tableau et est utilisé comme une file de priorité.

5 Fonctionnement de l'Algorithme Dijkstra

- On stocke le fait que aucune station n'ait été traité, aucune station n'ait encore de père, les plus petites distances sont encore indéfinis.
- On traite la station de départ et on met à jour les plus petites distances avec ses successeurs en utilisant les temps en secondes.
- Tant qu'on a des stations à traiter, on prend la plus petite des plus petites distances et on regarde si on peut améliorer les plus petites distances de ces successeurs en passant par ce sommet. Si c'est le cas alors ce dernier devient son père.
- Le tableau des pères est alors initialisé complètement.

6 Fonctionnement général

Tout d'abord, on initialise les différentes variables que l'on va utiliser pour lancer nos différentes fonctions ensuite on initialise le graphe du metro avec "init_subway" grâce au nombre de station qu'on a récupéré précédemment avec "count_nb_points". Les chemins sont initialisés à nul au début.

Le tableau des stations est alors initialisé pour contenir les informations de chaque station (le nom de la station, l'id de la station, le numéro de sa ligne). Ce processus est réalisé par la fonction "init_station".

Le tableau des chemins entre les stations est aussi initialisé avec le numéro du terminus et le temps en seconde de chaque chemin. Sachant que le metro est à double sens, le chemins opposé est également initialisé avec le même temps en seconde mais avec le terminus opposé. Ce processus est réalisé par la fonction "init_system".

Suite à cela on demande à l'utilisateur d'entré la gare à partir de laquelle il souhaite partir.

Une fois que l'utilisateur valide la gare de départ, on vérifie que son entré, qui peut soit être le nom de la station ou son id, correspond aux gares récupérer dans le fichier metro.txt puis l'id de la gare est stocké dans une variable.

Ensuite on demande à l'utilisateur d'entré la gare d'arrivé où il souhaite se rendre et on répète le processus de vérification et de stockage de la donnée.

Puis on lance la fonction "operation_short_way" qui va calculer le plus court chemin entre la station de départ et la station d'arrivé avec la fonction "operation_dijkstra" puis écrit dans le terminal le plus court chemin avec la fonction "write_way".

Finalement, la mémoire est libérée grâce au fonction "free" pour les variables initialisé au début, "free_graph" pour libérer la mémoire de la structure graph ainsi que "free_way_dijkstra" qui libère la mémoire stocker par l'initialisation de la structure dijkstra dans la fonction "operation_dijkstra".

Le dépôt git du projet : Github