

# Rapport - Projet théorie des graphes

ABDELMOUMENE Djahid

&

Hadjer

May 1, 2018

## 1 Introduction

L'idée principale de ce projet est de trouver le plus court chemin entre deux sommets d'un graphe orienté.

Le graphe qui nous a été fourni - la carte des pistes de serre chevalier - correspond à un graphe orienté muni d'une fonction de pondération positive, c'est à dire que tous les arcs ont un poids positif. Ce graphe n'est pas un graphe simple à première vue (existence de plusieurs arcs entre les même sommets), mais on pourra voir dans les sections suivantes qu'on pourra le considérer ainsi.

## 2 Execution

Pour executer le program, il suffit d'ouvrir le fichier 'index.html' avec le navigateur firefox.

## 3 Graphe

La première étape de ce projet était d'analyser la carte de la station de ski fournie et d'en tirer le graphe correspondant. On a donc considéré les pistes de ski et les remontées mécaniques comme des arcs, et les points d'intersection, de départ et d'arrivée de ces pistes comme des sommets.

### 3.1 le graphe

le graphe est stocké comme un liste d'adjacence, donc le graphe est un tableau des objets des sommets.

### 3.2 Sommets du graphe

#### Considération des sommets :

On considère comme sommets du graphe, les points de départ et d'arrivée

des pistes et les points où les pistes se croisent. La carte fournie a été simplifiée puisque les sommets correspondant à des points où se croisent plusieurs pistes sont en fait des zones (et non des points) où se croisent ces pistes. Les sommets n'ont pas des noms, mais ils sont définis par leur indice dans le tableau de graph, et avec leurs coordonnées (cartésiennes) dans l'image des pistes (2362x1013).

#### Indices des sommets :

Chaque sommet a été repéré par un indice allant de 0 à 147 : **on a donc 148 sommets au total.**

#### Coordonnées des sommets :

Tout sommet a une coordonnée cartésienne dans l'image des pistes (2362x1013), et c'est représenté comme un tableau de taille 2 où le premier élément est le x et le deuxième est le y.

**Les voisins de sommet** Tout sommet a un objet (tableau de hachage) où chaque voisin est représenté par son indice dans le tableau de graph, et la valeur est l'objet de l'arc. Exemple d'un objet des voisins:

extrait de data.json

```
"voisins": {  
  "116": {  
    "coords": [  
      [  
        527,  
        387  
      ],  
      [  
        486,  
        393  
      ],  
      [  
        469,  
        398  
      ],  
      [  
        451,  
        399  
      ],  
      [  
        437,  
        404  
      ],  
    ]  
  }  
}
```

```

        [
            409,
            399
        ]
    ],
    "len": 120,
    "nom": "PREVOUX",
    "couleur": "V"
},
"123": {
    "coords": [
        [
            531,
            386
        ],
        [
            539,
            390
        ],
        [
            524,
            397
        ],
        [
            521,
            403
        ]
    ],
    "len": 32,
    "nom": "PREVOUX1",
    "couleur": "R"
}
}

```

### 3.3 Arcs du graphe

#### Considération des arcs :

Les arcs du graphe correspondent bien évidemment aux pistes de ski. On a alors deux types d'arcs : des *descentes* et des *remontées*.

Une simplification du graphe a été effectuée : s'il existe plusieurs arcs de même extrémité de départ et d'arrivée, on ne retient que l'arc de poids le plus faible. La raison est que pour calculer un plus court chemin, l'arc de poids faible est le seul qui va être pris en compte, peu importe le nombre des autres arcs liant ces mêmes sommets.

### Informations stockées dans les arcs :

**Le nom:** Chaque arc à un nom unique, vu que la majorité des arcs ont des noms, si un arc n'as pas un nom il prend le nom de l'arc le plus proche mais avec un indice ajouté. eg: arc ,arc1, arc2..

**Les coordonnées** Tout les arcs ont un tableau des points dans l'image (2362x1013), et ces coordonnées sont utilisées pour dessiner les arcs, où on connect chaque deux points consécutifs avec un ligne.

**La longueur de l'arc** Chaque arc a un longueur, qui est la somme des distance de chaque deux points consécutifs dans le tableau des coordonnées de l'arcs.

**Couleur et poids des arcs** les arcs ont un qui represente le couleur ou le type de l'arc, Les différents types d'arcs ont chacun leur propre couleur qui a été repérée par un chaine des caractres. voici la liste des couleurs et types possibles avec le poids:

**Les pistes** (la couleur, poids un skieur debutant, poids pour un skieur expert):

- "V": vert, longueur\*1.05/10, longueur\*1/10
- "B": blue, longueur\*1.4/10 , longueur\*1.1/10
- "R": rouge, longueur\*2.2/10 , longueur\*1.2/10
- "N": noir, longueur\*3/10 , longueur\*1.3/10

**Les remontées** (le type, le poids):

- "TELECABINE": telecabine, longueur\*0.5/10
- "OEUF": oeuf, longueur\*0.75/10
- "TELESIEGE": telesiege, longueur\*0.9/10
- "TELESKI": teleski, longueur\*1/10

Voici la fonction qui permet la conversion du temps de parcours de l'arc et de sa couleur en poids:

```
dijkstra.js
function calculateArcCost(v){
    switch(v.couleur){
        case"TELESKI": return 1*v.len/10;
        case"TELESIEGE": return 0.9*v.len/10;
        case"OEUF": return 0.75*v.len/10;
```

```

        case"TELECABINE": return 0.5*v.len/10;
    }
    if(sel.value == 'Expert'){
        switch(v.couleur){
            case"V": return 1*v.len/10;
            case"B": return 1.1*v.len/10;
            case"R": return 1.2*v.len/10;
            case"N": return 1.3*v.len/10;
        }
    } else {
        switch(v.couleur){
            case"V": return 1.05*v.len/10;
            case"B": return 1.4*v.len/10;
            case"R": return 2.2*v.len/10;
            case"N": return 3*v.len/10;
        }
    }
}

```

### 3.4 le fichier d'entrée

Le fichier d'entrée est en format JSON, où il existe les objets (tableau de hachage) représenté par des `{ }`, et des tableaux représenté par des `[ ]`.

cette format nous permet de structurer l'objet de graphe dans le fichier de l'information.

## 4 L'algorithme

rien