Cahier des charges : programme 1

- Programme résolvant le problème du plus court chemin à origine unique
- Doit pouvoir lire un graphe depuis un fichier contenant une matrice d'adjacence
- Doit prendre en arguments le chemin du fichier représentant le graphe, ainsi que le numéro des sommets origine s0 et destination sd
- Doit afficher la longueur et le chemin dans sa sortie standard :
    - longueur
    - s0 -> s1 -> … -> sd
- Doit produire des images png permettant d'allumer au fur et à mesure les sommets du graphe utilisés
    - Le chemin des images produites est déclaré dans une variable d'environnement du shell nommée DOT_PATH. Si cette variable n'existe pas, les images ne sont pas générées

NB : les graphes utilisés lors de l'évaluation de performance de vos algos seront :

- des graphes pondérés sans poids négatif ou nuls
- plutôt des graphes creux

Exemple 1 :

(la variable d'environnement DOT_PATH n'existe pas)

```
./prog1.exe ./graph1.adj  0  7
```

- Renvoie :

```
49
0 -> 5 -> 2 -> 4 -> 7
```

Exemple 2 :

```
export DOT_PATH=~/output
./prog1.exe ./graph1.adj  0  7
```

- Renvoie :

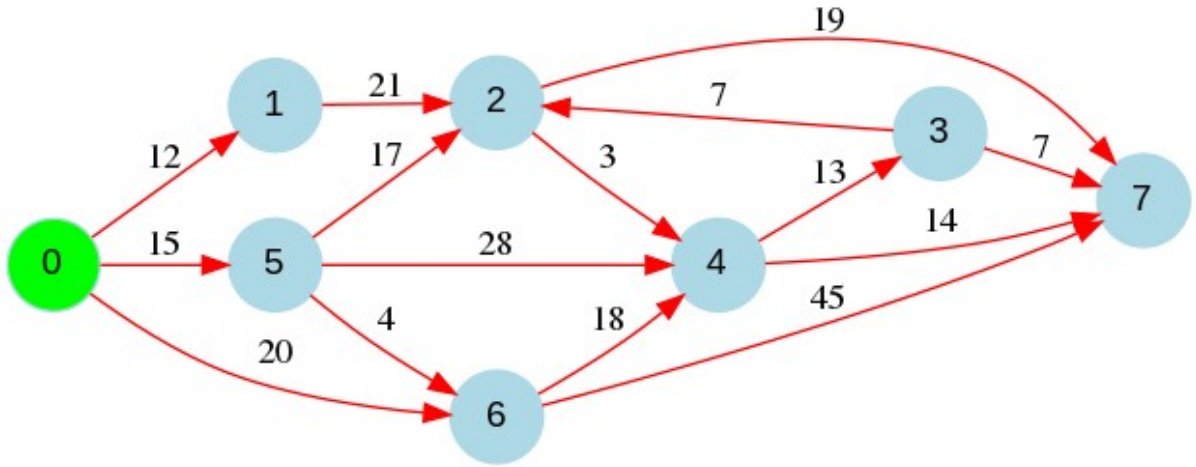
```
49
0 -> 5 -> 2 -> 4 -> 7
```

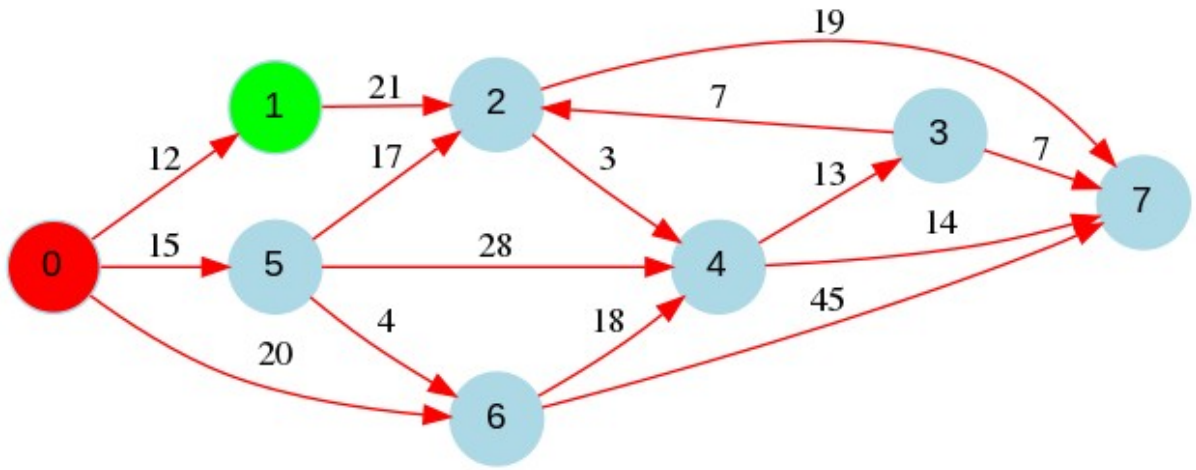
- Produit les fichiers suivants dans ~/output : (les noms des fichiers image doivent correspondre au nom du fichier contenant la matrice

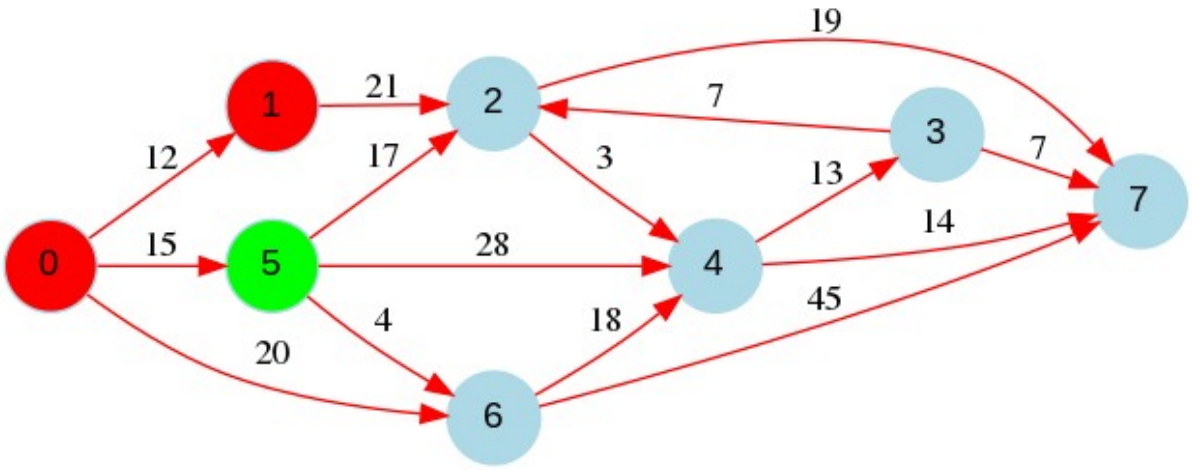
d’adjacence. Ici, les images s’appellent donc graph1_0.png, graph1_1.png, etc. puisque la matrice s’appelait graph1.adj)

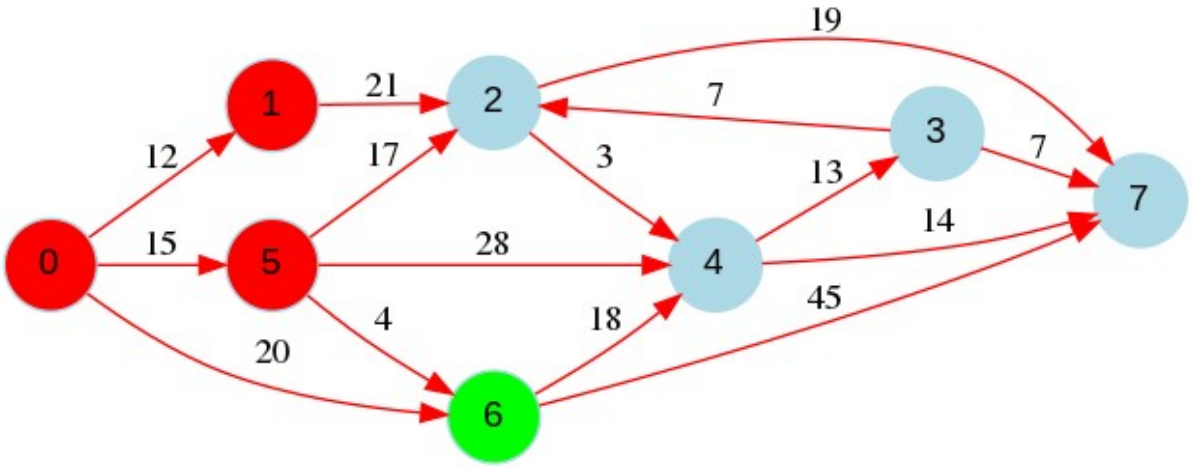
```
$ ls ~/output
graph1_0.png graph1_1.png graph1_2.png graph1_3.png graph1_4.png
graph1_5.png graph1_6.png graph1_7.png graph1_8.png
```

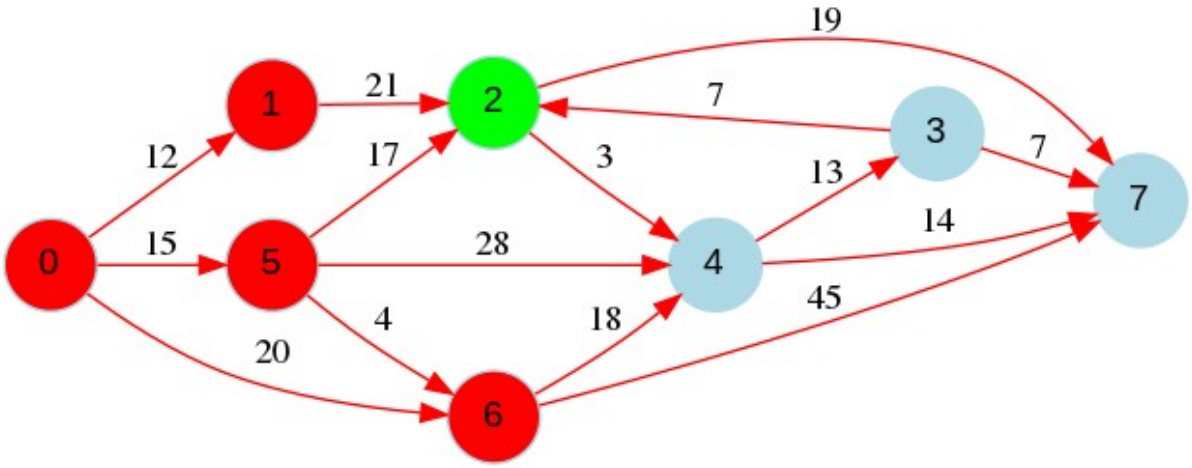
- Voici ce que contiennent les fichiers images produits, dans l’ordre :

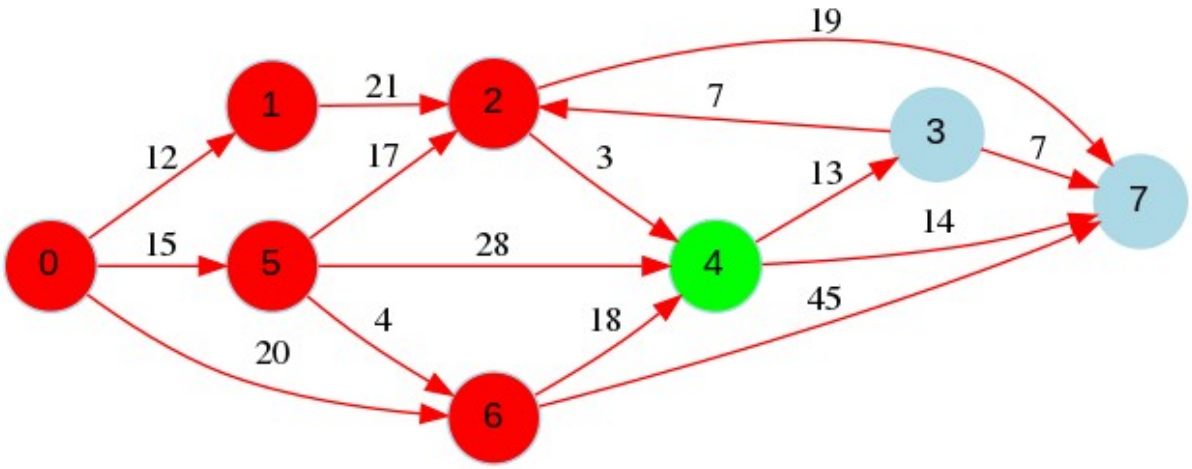


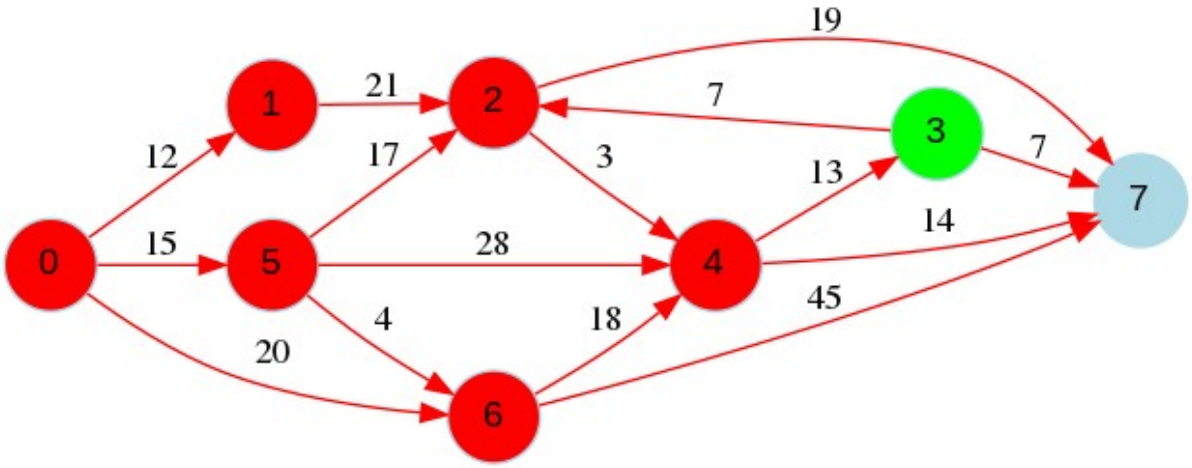


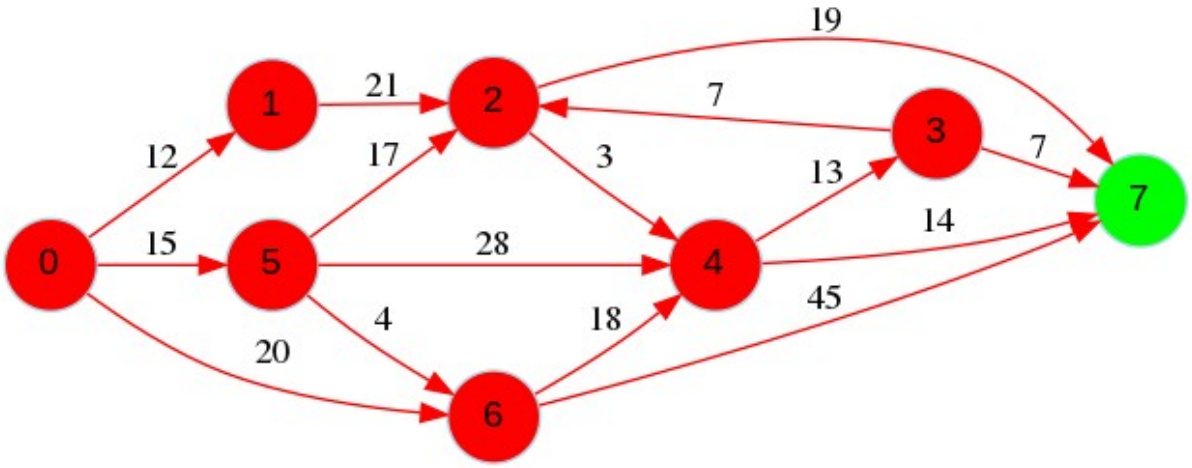

19
1
21
2
7
3
7
12
17
3
13
7
14
0
15
5
28
4
45
4
18
20
6

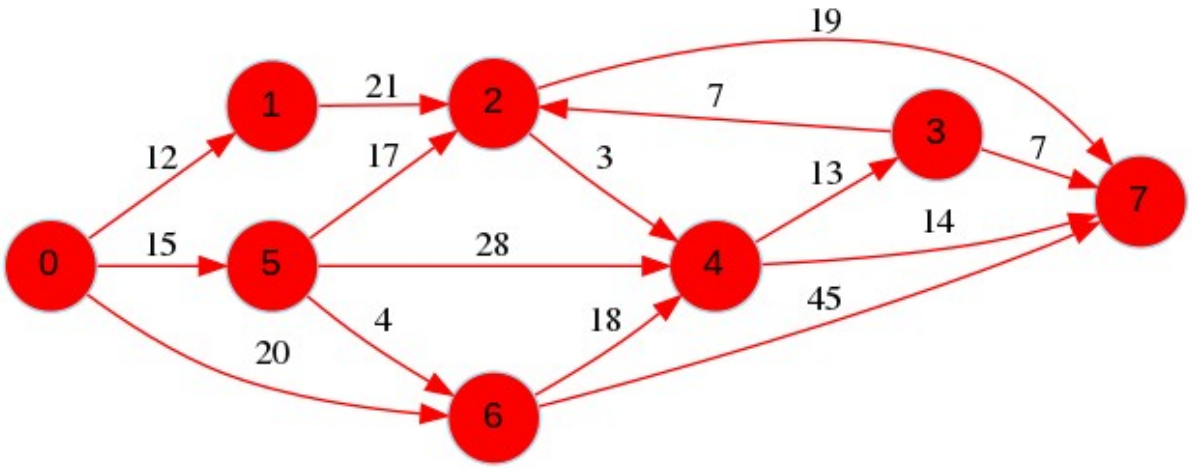

19
1
21
2
7
3
7
12
17
3
13
7
14
0
15
5
28
4
45
4
18
20
6


19
1
21
2
7
3
7
12
17
3
13
7
14
0
15
5
28
4
45
4
18
20
6


0
1
2
3
4
5
6
7
12
15
20
21
17
4
28
3
18
19
7
13
14
45
7


0
1
2
3
4
5
6
7
12
15
20
21
17
4
28
3
18
19
7
13
14
45
7


0
1
2
3
4
5
6
7
12
15
20
21
17
4
28
3
18
19
7
13
14
45
7


0
1
2
3
4
5
6
7
12
15
21
17
3
19
7
7
13
14
28
4
18
20
45