

JACQUOT Thierry

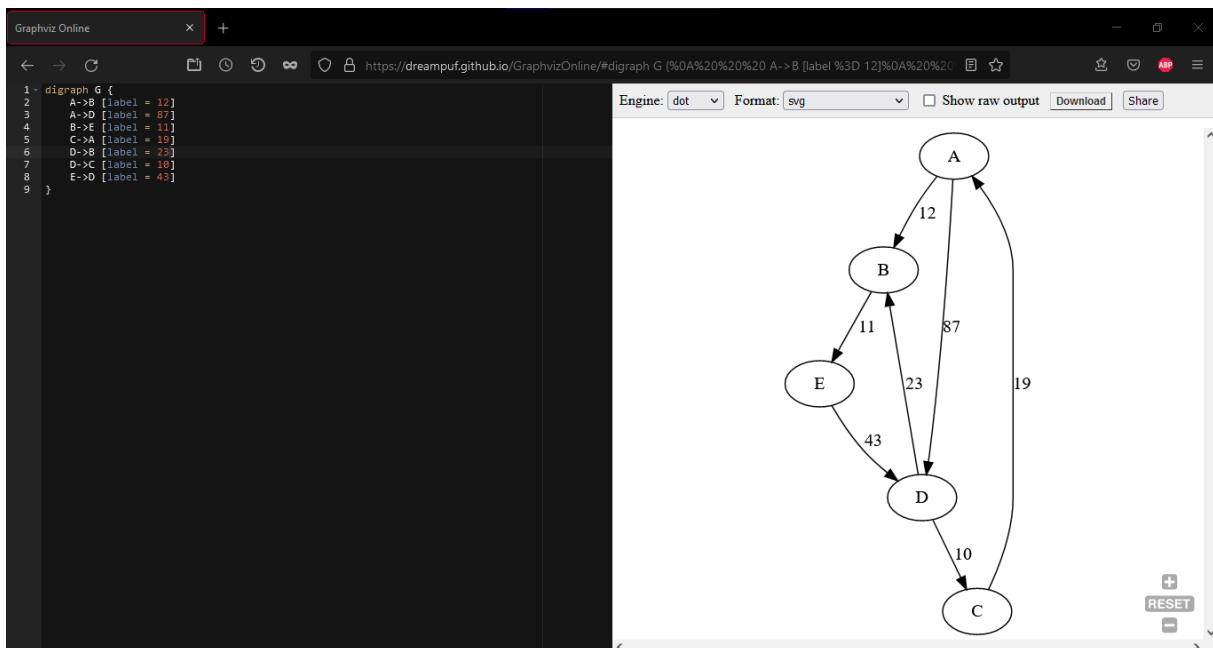
GUIFFAULT Elian

Rapport

SAE 2.02 Exploration algorithmique

Question 10

A partir du graphe généré à partir du programme main, nous arrivons à ce résultat à partir du site Web :



Question 13

L'algorithme de Bellman-Ford est le suivant :

Fonction BellmanFord (G : graphe, d : chaîne de caractères)

L(x) : int, distance

Parent : parent d'un nœud

N : nœud, sommet d'un graphe

Continuer : booléen, permettant de faire fonctionner la boucle principale

DEBUT

//Initialisation

Pour n nœuds dans G faire :

 L(n) <- +infini

 Parent(n) <- null

FINPOUR

L(d) <- 0

//boucle principale

continuer <- vrai

Tant que continuer faire :

 continuer <- faux

Pour chaque arc (u,v,poids) de G faire :

Si L(v) > L(u)+poids alors :

 L(v) <- L(u)+poids

 Parent(v) <- u

 continuer <- vrai

 FINSI

 FINPOUR

FINTANTQUE

FIN

Question 21

A chaque itération, l'algorithme de Bellman-Ford calcule chaque chemin s'il peut. Dans le graphe de la figure 10 : boucle, il prend comme chemin minimal AB au début, puis prend le chemin minimal ADCB, plus court.

L'algorithme de Dijkstra ne calcule pas chaque chemin à chaque fois. Il choisit une valeur minimale et calcule la valeur du prochain nœud concerné.

Question 22

Nous pouvons conclure que l'algorithme de Dijkstra effectue moins de calculs que l'algorithme de Bellman-Ford. L'algorithme de Dijkstra est donc potentiellement plus efficace.

Question 23

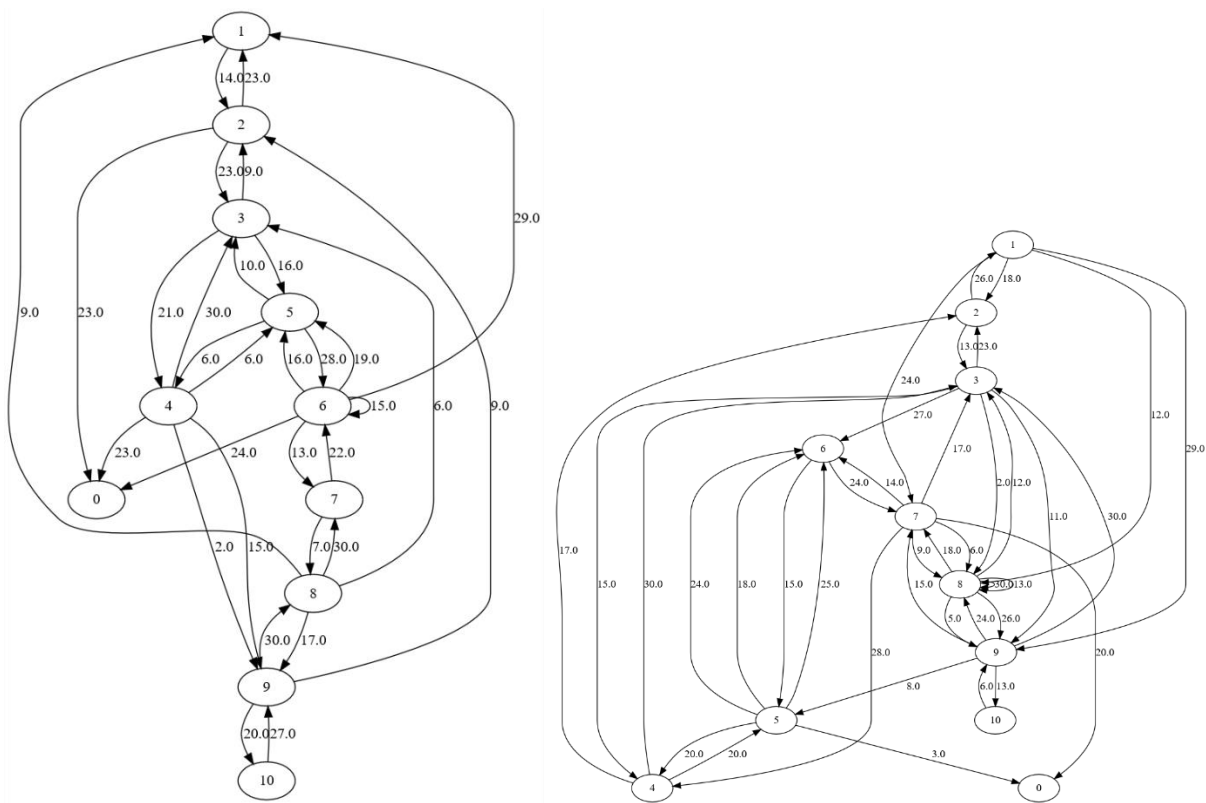
Voici un algorithme qui compare en secondes le temps d'exécution de chaque algorithme :

Nombre moyen de nœuds	Point fixe / Bellman-Ford	Dijkstra
45 (Graphe11)	0,00013 secondes	0,000095 secondes
250 (Graphe34)	0,0010 secondes	0,00075 secondes
2500 (Graphe101)	0,034 secondes	0,0085 secondes

Avec les résultats obtenus, l'algorithme de Dijkstra semble plus efficace que celui de Bellman-Ford.

Question 25

Voici le rendu de 2 petits graphes générés en Graphviz :



Question 26

Quel que soit le nombre de nœuds, l'algorithme de Dijkstra est au moins aussi efficace que l'algorithme de Bellman-Ford, et est très souvent bien plus efficace en termes de performances.

Question 27

Le ratio de performances de l'algorithme de Dijkstra est de 1.5 par rapport à l'algorithme de Bellman-Ford, pour un graphe de taille moyenne (autour de 150 nœuds)

Ce rapport tend à se réduire lorsqu'il y a moins de nœuds, et à augmenter lorsque le graphe est plus grand. Sur certains grands graphes, l'algorithme de Dijkstra est plus que 2 fois plus puissant.

Question 28

Pour conclure, la SAE nous a permis d'implémenter des algorithmes utilisés en mathématique dans des programmes afin de comparer leurs performances.

Nous avons appris à créer un programme créant des graphes, à traduire un concept mathématique en code, et à expérimenter avec ces algorithmes (différences de fonctionnement notamment).

Nous avons noté que l'algorithme de Dijkstra est meilleur pour trouver le chemin le plus court surtout lors de grands graphes.