Distances

Malika More
(malika.more@u-clermont1.fr)

IREM Clermont-Ferrand

Formation ISN

20 octobre 2011

Plan du cours

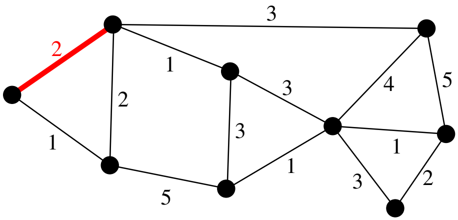
1. Introduction
2. Algorithme de Dijkstra

1 Introduction

2 Algorithme de Dijkstra

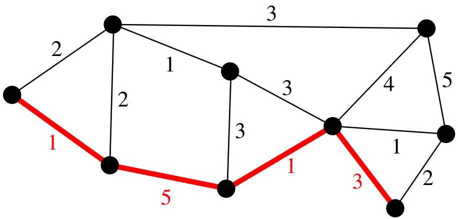
Graphe valué

- Chaque arête a un poids (> 0)
- Le poids d'une chaîne est la somme des poids des arêtes qui la composent.



Graphe valué

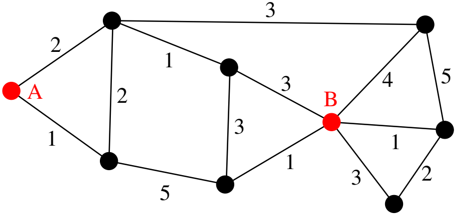
- Chaque arête a un poids (> 0)
- Le poids d'une chaîne est la somme des poids des arêtes qui la composent.



$$1 + 5 + 1 + 3 = 10$$

Distance

Entre A et B
on trouve des chaînes de
poids diffrents



distance entre A et B =
plus petit poids d'une chaîne
entre A et B

$$d(A, B) = 6$$

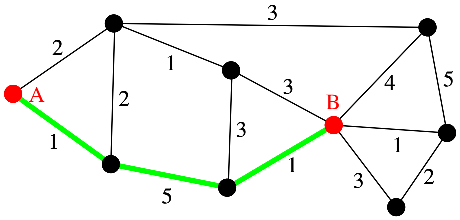
$1 + 5 + 1 = 7$
$2 + 3 + 5 + 1 = 11$
$2 + 1 + 3 = 6$
$2 + 1 + 3 + 1 = 7$
etc.

Distance

Entre A et B
on trouve des chaînes de
poids diffrents

distance entre A et B =
plus petit poids d'une chaîne
entre A et B

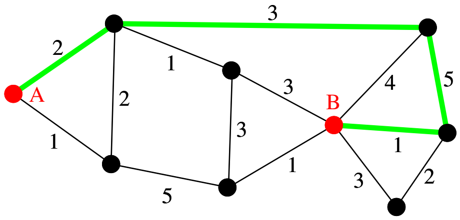
$$d(A, B) = 6$$



$1 + 5 + 1 = 7$
$2 + 3 + 5 + 1 = 11$
$2 + 1 + 3 = 6$
$2 + 1 + 3 + 1 = 7$
etc.

Distance

Entre A et B
on trouve des chaînes de
poids diffrents



distance entre A et B =
plus petit poids d'une chaîne
entre A et B

$$d(A, B) = 6$$

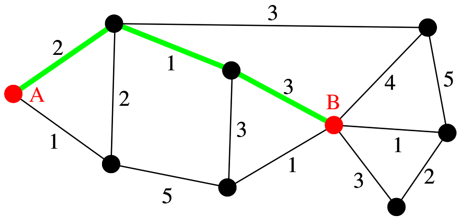
$1 + 5 + 1 = 7$
$2 + 3 + 5 + 1 = 11$
$2 + 1 + 3 = 6$
$2 + 1 + 3 + 1 = 7$
etc.

Distance

Entre A et B
on trouve des chaînes de poids diffrents

distance entre A et B =
plus petit poids d'une chaîne entre A et B

$$d(A, B) = 6$$



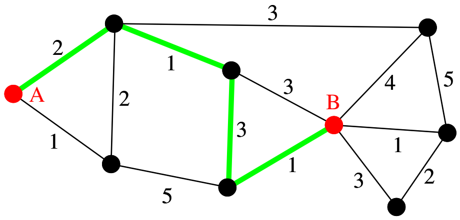
$1 + 5 + 1 = 7$
$2 + 3 + 5 + 1 = 11$
$2 + 1 + 3 = 6$
$2 + 1 + 3 + 1 = 7$
etc.

Distance

Entre A et B
on trouve des chaînes de poids diffrents

distance entre A et B =
plus petit poids d'une chaîne entre A et B

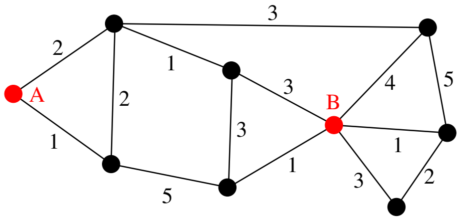
$$d(A, B) = 6$$



$1 + 5 + 1 = 7$
$2 + 3 + 5 + 1 = 11$
$2 + 1 + 3 = 6$
$2 + 1 + 3 + 1 = 7$
etc.

Distance

Entre A et B
on trouve des chaînes de poids diffrents



distance entre A et B =
plus petit poids d'une chaîne entre A et B

$$d(A, B) = 6$$

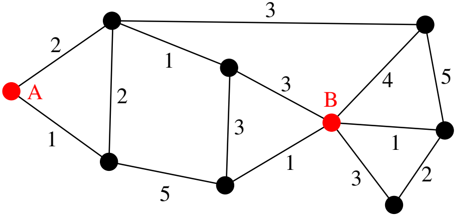
$1 + 5 + 1 = 7$
$2 + 3 + 5 + 1 = 11$
$2 + 1 + 3 = 6$
$2 + 1 + 3 + 1 = 7$
etc.

Distance

Entre A et B
on trouve des chaînes de poids diffrents

distance entre A et B =
plus petit poids d'une chaîne entre A et B

$$d(A,B) = 6$$

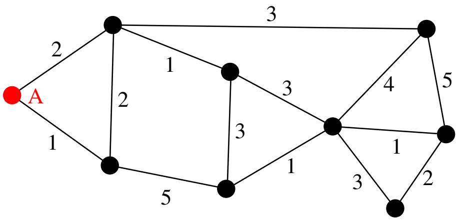


$1 + 5 + 1 = 7$
$2 + 3 + 5 + 1 = 11$
$2 + 1 + 3 = 6$
$2 + 1 + 3 + 1 = 7$
etc.

Question

Étant donnés $\begin{cases} \text{un graphe connexe valué} \\ \text{et un sommet de départ} A \end{cases}$

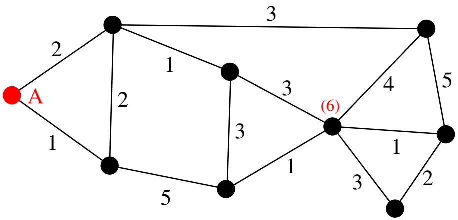
On veut calculer la distance entre A
et n'importe quel autre sommet



Question

Étant donnés $\begin{cases}\text{un graphe connexe valué} \\ \text{et un sommet de départ} A\end{cases}$

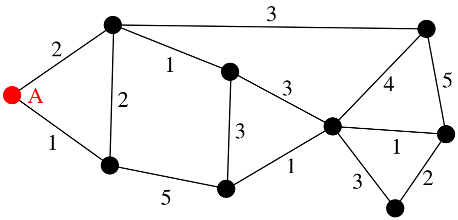
On veut calculer la distance entre A
et n'importe quel autre sommet



1 Introduction

2 Algorithme de Dijkstra

Idée générale

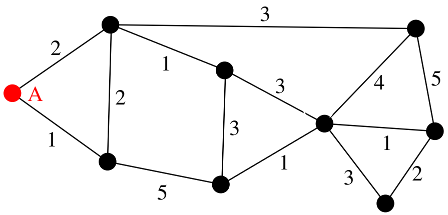


- Au départ, la distance vaut 0 pour A et ∞ pour les autres sommets.
- À chaque itération, on fixe la valeur définitive de la distance pour un sommet de plus et on fait décroître les valeurs provisoires des distances pour d'autres sommets.
- On stoppe quand les valeurs de toutes les distances sont définitives.

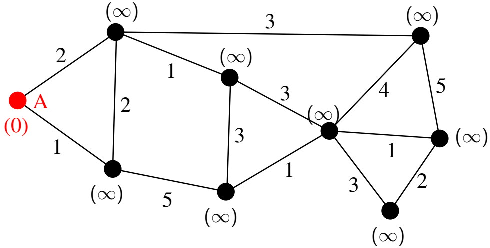
Méthode

- Initialiser les distances :
  $$\begin{cases} \text{pour } A \text{ à } (0) \\ \text{pour les autres sommets à } (\infty) \end{cases}$$
- Marquer le sommet A
- Répéter :
  - Trouver toutes les arêtes qui relient un sommet marqué et un sommet non marqué
  - Mettre à jour les distances des sommets non marqués incidents à ces arêtes
  - Choisir un sommet non marqué dont la distance est minimale
  - Marquer ce sommet
- S'arrêter lorsque tous les sommets sont marqués
- Retourner les distances

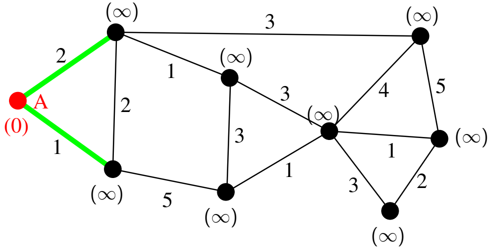
Exemple



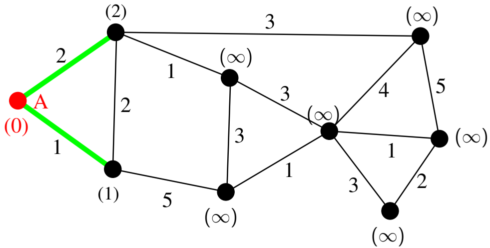
Exemple



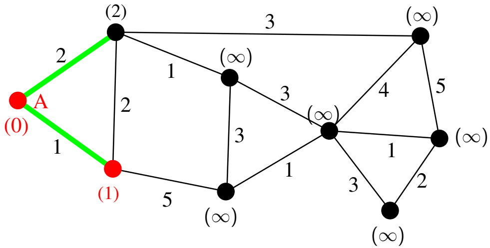
Exemple



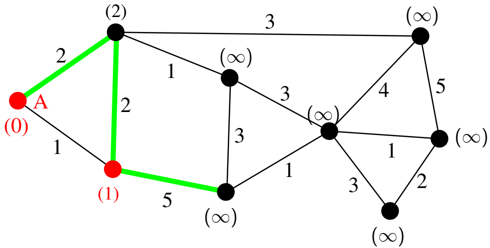
Exemple



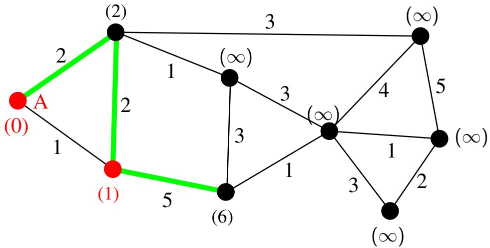
Exemple



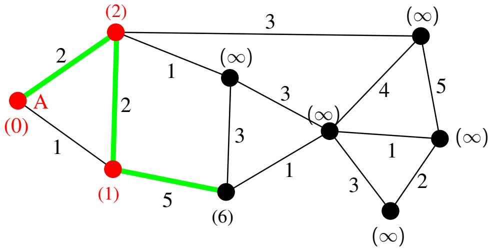
Exemple



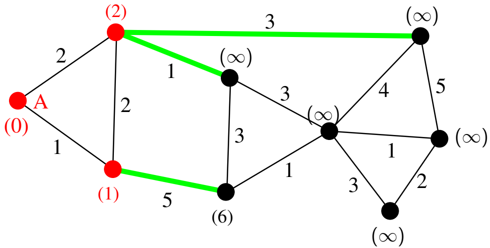
Exemple



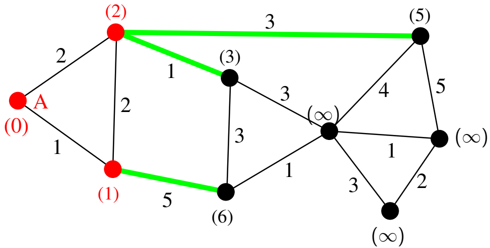
Exemple



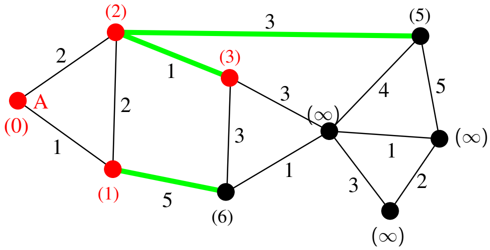
Exemple



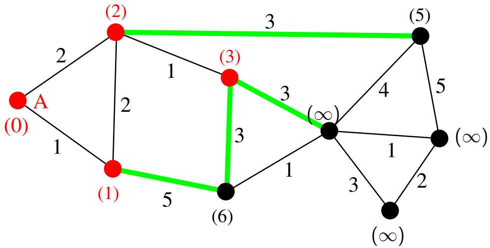
Exemple



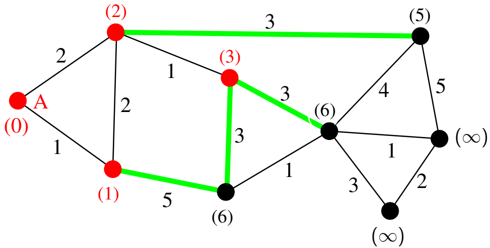
Exemple



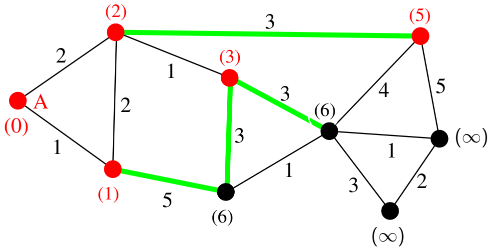
Exemple



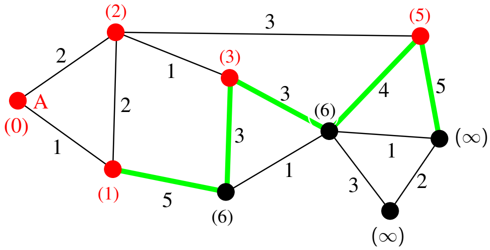
Exemple



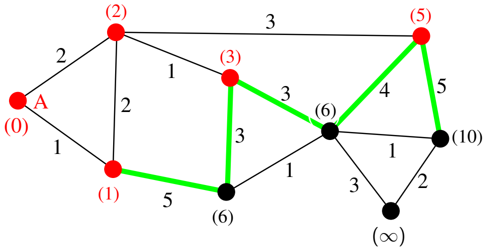
Exemple



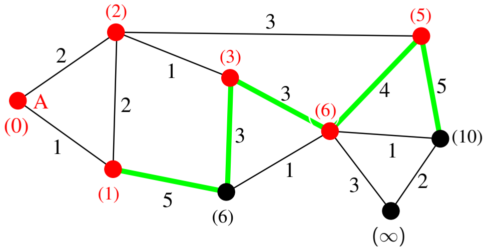
Exemple



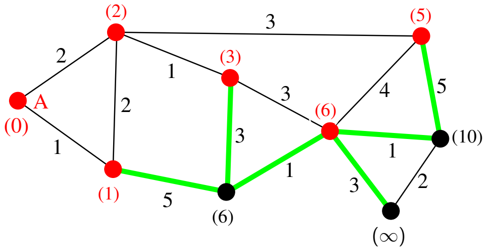
Exemple



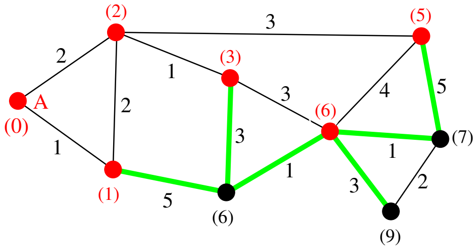
Exemple



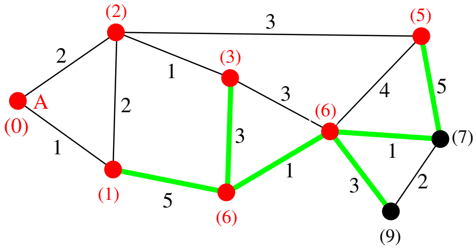
Exemple



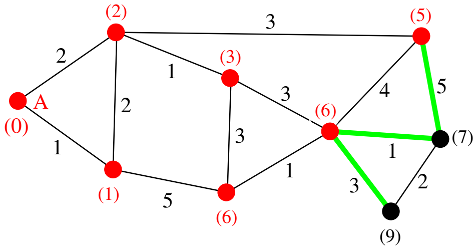
Exemple



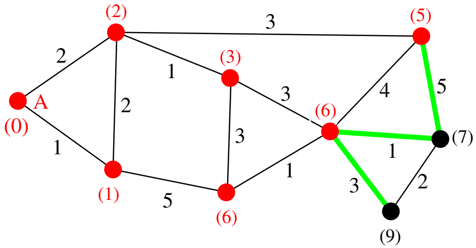
Exemple



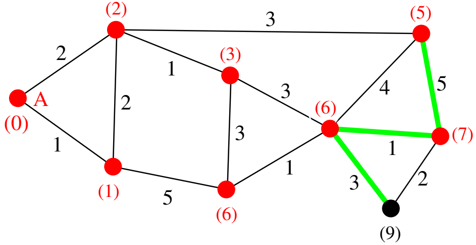
Exemple



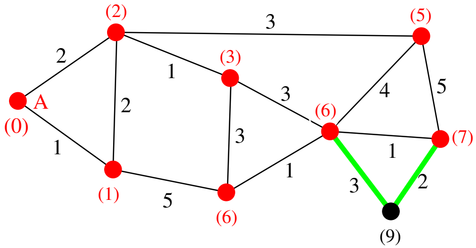
Exemple



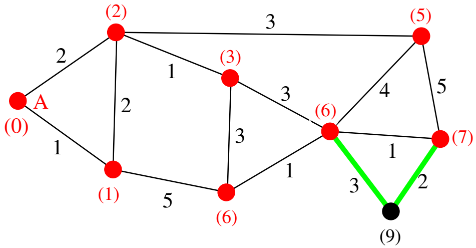
Exemple



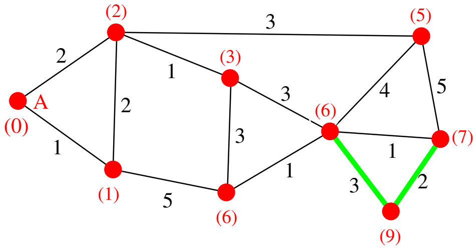
Exemple



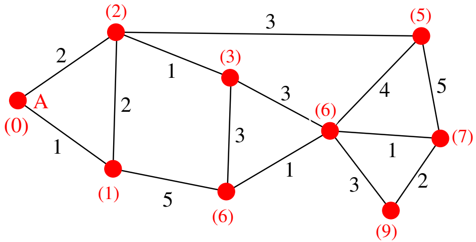
Exemple



Exemple



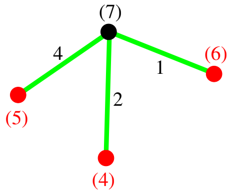
Exemple



Rappel de l'algorithme de Dijkstra

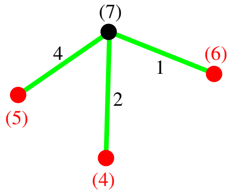
- Initialiser les distances :
  $\begin{cases} \text{pour } A \text{ à } (0) \\ \text{pour les autres sommets à } (\infty) \end{cases}$
- Marquer le sommet A
- Répéter :
  - Trouver toutes les arêtes qui relient un sommet marqué et un sommet non marqué
  - Mettre à jour les distances des sommets non marqués incidents à ces arêtes
  - Choisir un sommet non marqué dont la distance est minimale
  - Marquer ce sommet
- S'arrêter lorsque tous les sommets sont marqués
- Retourner les distances

Mise à jour des distances

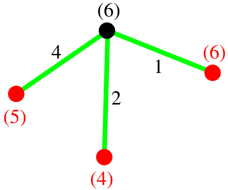


- On compare (7) avec
  - $(5) + 4 = 9$
  - $(4) + 2 = 6$
  - $(6) + 1 = 7$
- On trouve plus petit $4 + 2 = 6 < 7$
- On remplace (7) par (6)

Mise à jour des distances



- On compare (7) avec
  - $(5) + 4 = 9$
  - $(4) + 2 = 6$
  - $(6) + 1 = 7$
- On trouve plus petit $4 + 2 = 6 < 7$
- On remplace (7) par (6)



Pourquoi l'algorithme de Dijkstra calcule-t-il les distances avec le sommet de départ ?

- Dijkstra s'arrête toujours
- la distance des sommets marqués ne change plus
- la distance des sommets marqués est correcte

Pourquoi l'algorithme de Dijkstra calcule-t-il les distances avec le sommet de départ ?

- Dijkstra s'arrête toujours
- la distance des sommets marqués ne change plus
- la distance des sommets marqués est correcte

À chaque étape, on marque un sommet non marqué.
On ne démarque jamais un sommet marqué.
On s'arrête quand tous les sommets sont marqués, ce qui finit forcément par se produire.

Pourquoi l'algorithme de Dijkstra calcule-t-il les distances avec le sommet de départ ?

- Dijkstra s'arrête toujours
- la distance des sommets marqués ne change plus
- la distance des sommets marqués est correcte

On met à jour les distances uniquement pour des sommets non marqués.
On ne démarque jamais un sommet marqué.

Pourquoi l'algorithme de Dijkstra calcule-t-il les distances avec le sommet de départ ?

- Dijkstra s'arrête toujours
- la distance des sommets marqués ne change plus
- la distance des sommets marqués est correcte

À chaque étape, la valuation ($< \infty$) d'un sommet y non marqué représente le poids d'une chaîne entre le sommet de départ et y.
Quand on met à jour les distances, on essaye de faire diminuer cette valuation, càd de trouver une chaîne de poids plus faible.
Quand on marque un sommet, on décide qu'une chaîne de poids minimum a été trouvée. C'est ce qu'il faut vérifier.

Pourquoi l'algorithme de Dijkstra calcule-t-il les distances avec le sommet de départ ?

- Dijkstra s'arrête toujours
- la distance des sommets marqués ne change plus
- la distance des sommets marqués est correcte

Pour commencer, (0) est bien la distance entre le sommet de départ et lui-même.
Hypothèse de récurrence : la valuation des sommets déjà marqués est la distance.

Pourquoi l'algorithme de Dijkstra calcule-t-il les distances avec le sommet de départ ?

- Dijkstra s'arrête toujours
- la distance des sommets marqués ne change plus
- la distance des sommets marqués est correcte

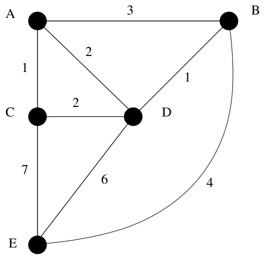
Pour choisir le sommet qu'on marque, on étudie les valuations des sommets non marqués.
Càd on compare des valeurs du type $Val(y) = d(A,x) + Poids(x,y)$ pour x un sommet marqué et y un sommet non marqué.
Quand on choisit de marquer le sommet de plus faible valuation, on est donc certain d'avoir trouvé une chaîne de poids le plus faible possible le reliant au sommet de départ.

Pourquoi l'algorithme de Dijkstra calcule-t-il les distances avec le sommet de départ ?

- Dijkstra s'arrête toujours
- la distance des sommets marqués ne change plus
- la distance des sommets marqués est correcte

Remarque :
Si on faisait tourner l'algorithme de Dijkstra sur un graphe non connexe, les sommets inatteignables à partir du sommet de départ resteraient à distance (∞), ce qui est naturel.
Mais il faudrait changer la condition d'arrêt en "S'arrêter lorsqu'on ne peut plus marquer de nouveau sommet".

Présentation du calcul

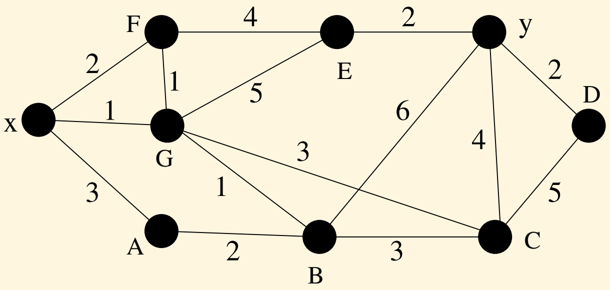


On applique l'algorithme de Dijkstra à partir du sommet A sur le graphe ci-contre.

Sommet	A	B	C	D	E
Voisins	B C D 3 1 2	A D E 3 1 4	A D E 1 2 7	A B C E 2 1 2 6	B C D 4 7 6
Initialisation	0	∞	∞	∞	∞
	•	3	1	2	∞
		3	•	2	∞
		3		•	8
		•			7
					•

Exercice 1

Soit G_5 le graphe valué suivant.

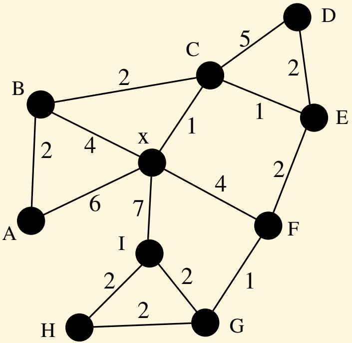


Pour le graphe G_5 ci-dessus, utiliser l'algorithme de DIJKSTRA pour trouver les distances $d(x, a)$ pour $a \in G_5$. En particulier, déterminer une chaîne de poids minimum reliant x à y.

Exercice 2

1. Proposer une (petite) modification de l'algorithme de Dijkstra permettant de renvoyer non seulement les distances entre le sommet de départ et tous les autres sommets, mais aussi des chaînes réalisant ces distances.
2. Utiliser votre algorithme sur le graphe G_5.

Exercice 3



Pour le graphe ci-dessus, utiliser l'algorithme de DIJKSTRA pour trouver les distances $d(x, a)$ pour $a \in G$.

FIN