Algo 7 – ALGORITHME DE DIJKSTRA

I Algorithme de Dijkstra

L'algorithme de Dijkstra (prononciation proche de "Dekstra") du nom de son inventeur l'informaticien néerlandais Edsger Dijkstra, a été publié en 1959. Il permet de déterminer un plus court chemin entre deux sommets d'un graphe pondéré. Nous allons le présenter par un exemple.

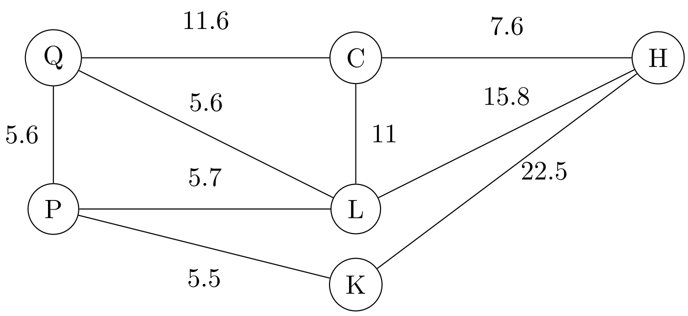
I.1 Exemple d'itinéraire routier

On souhaite déterminer un itinéraire entre Caudan et la plage de Kerguélen.

Voici les données dont on dispose :

- Lorient - Quéven : 5.6 km, 12 min.
- Lorient - Hennebont : 15.8 km, 25 min.
- Lorient - Ploemeur : 5.7 km, 13 min.
- Lorient - Caudan : 11 km, 22 min.
- Quéven - Caudan : 11.6 km, 13 min.
- Quéven - Ploemeur : 5.6 km, 8 min.
- Ploemeur - Kerguélen : 5.5 km, 10 min.
- Hennebont - Kerguélen : 22.5 km, 30 min.
- Caudan - Hennebont : 7.6 km, 13 min.

Si on s'interesse à la distance parcourue, on peux representer le plan par ce graphe pondéré :



I.2 Principe de l'algorithme de Dijkstra

L'algorithme s'appuie sur un tableau à double entrée. Dans la première ligne, par colonne, on liste tous les sommets du graphe et on rajouter un colonne « choix » qui désignera à chaque étape un sommet dit *actuel* associé à une distance dite *actuelle*. Par la suite, chaque ligne du tableau va décrire une étape de l'algorithme.

- Pour remplir la deuxième ligne, on affecte au sommet de départ la valeur 0 et à tous les autres sommets la valeur ∞. Pour remplir la case « choix », on choisit le sommet affecté à la plus petite valeur, ici forcément le sommet de départ. On note alors dans la case « choix », le nom du tel sommet avec entre parenthèse la valeur affectée à ce sommet, cela définit le sommet *actuel* et la distance *actuelle*.
- Pour passer du ligne à l'autre, on regarde les sommets adjacents au sommet *actuel*.
  - On grise la case du sommet *actuel*.
  - Pour tous les sommets non adjacents au sommet *actuel*, on recopie la valeur de la ligne précédente.
  - Pour tous les sommets adjacents,
    - On calcule la somme de la *distance actuelle* et de la distance entre le sommet *actuel* et le sommet adjacent que l'on regarde.
    - Si cette somme est supérieure à la valeur de la ligne précédente, on recopie uniquement la valeur de la ligne précédente.
    - Si cette somme est inférieure à la valeur de la ligne précédente, on indique cette valeur dans la case et on ajoute entre parenthèse le nom du sommet actuel.
  - On note alors dans la case « choix », le nom du sommet associé à la valeur la plus petite avec entre parenthèse la valeur affectée à ce sommet, cela définit un nouveau sommet *actuel* et distance *actuelle*.
- On s'arrête quand le sommet d'arrivée se retrouve dans la case « choix ». La valeur associée donne la distance minimale. On peut également remonter l'algorithme pour trouver le trajet associé à cette distance minimale.

Utiliser l'algorithme de Dijkstra pour trouver la distance minimale entre Caudan et Kerguélen.

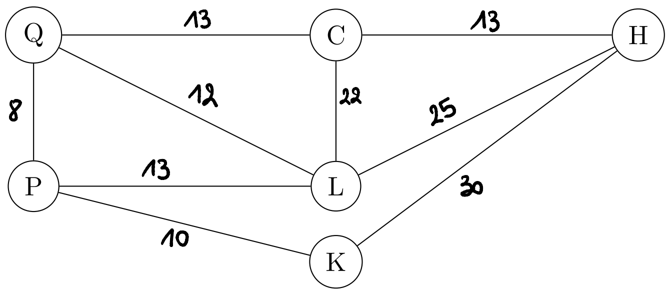
C départ	H	K arrivée	L	P	Q	Choix
0	∞	∞	∞	∞	∞	C(0)
	$0+7.6<\infty$ 7.6 (C)	∞	$0+11<\infty$ 11 (C)	∞	$0+11.6<\infty$ 11.6 (C)	H(7.6)
		$7.6+22.5<\infty$ 30.1(H)	$7.6+15.8>11$ 11(C)	∞	11.6 (C)	L(11)
		30.1(H)		$11+5.7<\infty$ 16.7(L)	$11+5.6>11.6$ 11.6(C)	Q(11.6)
		30.1(H)		$11.6+5.6>16.7$ 16.7(L)		P(16.7)
		$16.7+5.5<30.1$ 22.2(P)				K(22.2) STOP

Distance minimale entre C et K : 22.2
Chemin minimal entre C et K : C-L-P-K

I.3 Trajet le plus rapide

Recherchons maintenant le chemin le plus rapide pour aller de Caudan (C) à Kerguélen (K).

1. Compléter l'arbre pondéré correspondant.



2. Appliquer l'algorithme de Dijkstra.

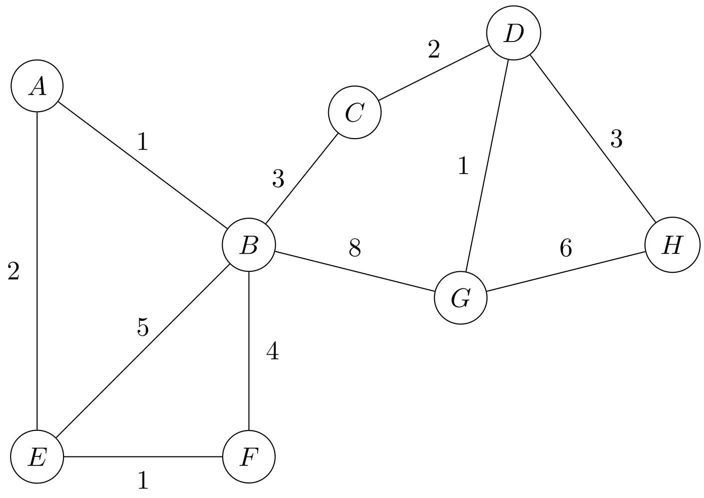
C départ	H	K arrivée	L	P	Q	Choix
0	∞	∞	∞	∞	∞	C(0)
	0+13<∞ 13(C)	∞	0+22<∞ 22(C)	∞	0+13<∞ 13(C)	H(13)
		13+30<∞ 43(H)	13+25>22 22(C)	∞	13(C)	Q(13)
		43(H)	13+12>22 22(C)	13+8<∞ 21(Q)		P(21)
		21+10<43 31(P)	21+13>22 22(C)			L(22)
		31(P)				K(31) STOP

Distance minimale entre C et L : 31

Chemin minimal entre C et L : C-Q-P-K

II Entrainement

Exercice 1 Appliquer à la main l'algorithme de Dijkstra au graphe pondéré suivant, pour touver le plus court chemin de H à E.

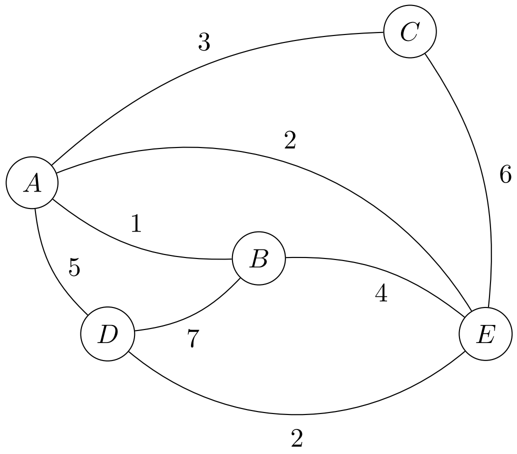


A	B	C	D	E arrivée	F	G	H départ	Choix
∞	∞	∞	∞	∞	∞	∞	0	H(0)
∞	∞	∞	$0+3<\infty$ 3(H)	∞	∞	$0+6<\infty$ 6(H)		D(3)
∞	∞	$3+2<\infty$ 5(D)		∞	∞	$3+1<6$ 4(D)		G(4)
∞	$4+8<\infty$ 12(G)	5(D)		∞	∞			C(5)
∞	$5+3<12$ 8(C)			∞	∞			B(8)
$8+1<\infty$ 9(B)				$8+5<\infty$ 13(B)	$8+4<\infty$ 12(B)			A(9)
				$9+2<13$ 11(A)	12(B)			E(11) STOP

Distance minimale entre H et E : 11

Chemin minimal entre H et E : H-D-C-B-A-E

Exercice 2 Considérons le graphe pondéré ci-dessous.



En appiquant l'algorithme de Dijkstra, trouver la distance de B à chacun des autres sommets.

A	B départ	C	D	E	Choix
∞	0	∞	∞	∞	B (0)
$0+1<\infty$ 1(B)		∞	$0+7<\infty$ 7(B)	$0+4<\infty$ 4(B)	A(1)
		$1+3<\infty$ 4(A)	$1+5<7$ 6(A)	$1+2<4$ 3(A)	E(3)
		$3+6>4$ 4(A)	$3+2<6$ 5(E)		C(4)
			5(E)		D(5)

Donc

$d(B,A)=1$ $d(B,C)=4$

$d(B,E)=3$ $d(B,D)=5$