Chapitre 2 Plus Court Chemin

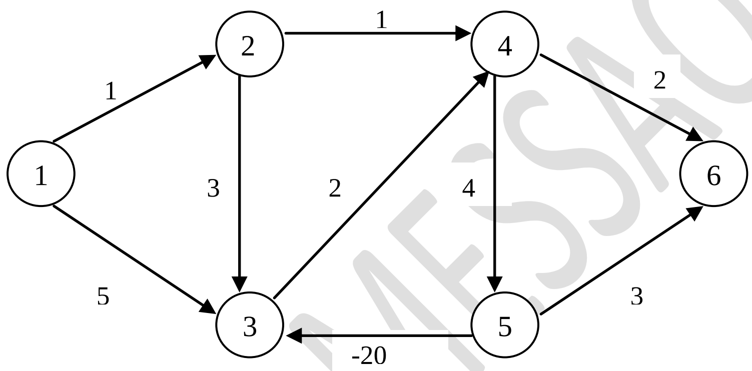
I- Problématique

1- Définition du problème

Etant donné un graphe orienté, on associe à chaque arc (*i*, *j*) un nombre C_{ij} appelé coût ou longueur de l'arc.
Le problème consiste généralement à trouver le plus court chemin allant d'un sommet fixé (à tire indicatif, le sommet 1) vers un autre sommet quelconque du graphe.

Exemple

L'objectif dans cet exemple est de trouver le Plus Court Chemin de 1 vers 6.



2- Remarques

- La longueur d'un arc peut être positive ou négative. Contrairement à une distance qui ne peut être que positive.
- Quand il y a un circuit de longueur négative (exemple 3 – 4 – 5 – 3), on ne peut pas trouver de chemin optimal de 1 vers 6 car on peut toujours trouver mieux : en effet, en commençant par le sommet 1 et en passant par ce circuit on réduit à chaque fois la longueur totale par **-14** (2 + 4 - 20 = -14).
- Ainsi, la condition nécessaire pour avoir un plus court chemin est l'absence d'un **circuit de longueur négative** (appelé aussi **circuit absorbant**) dans le graphe.

3- Principe de l'optimalité

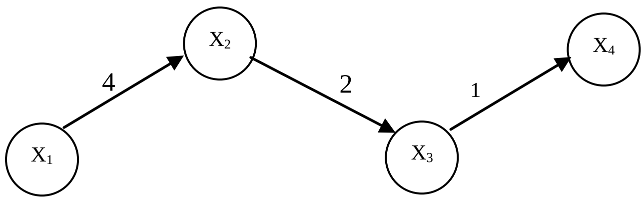
Principe : Tout chemin optimal est formé par des sous chemins optimaux.

Exemple : si le chemin 1 – 3 – 2 – 4 – 6 est optimal
alors les sous chemins suivants : 1 – 3, 1 – 3 – 2 et 1 – 3 – 2 – 4 sont optimaux.

4- Définitions

1) Soit G = (X, U) un graphe orienté ; on associe à chaque arc $u \in U$ un nombre réel noté L(u) = L(i, j) = C_{ij} où u=(i, j)
On appelle longueur d'un chemin *C*, L (*C*) = $\sum_{(i,j)} L(i, j)$

Exemple



$L(C) = L(X_1 – X_2 – X_3 – X_4) = L(X_1,X_2)+ L(X_2,X_3)+ L(X_3,X_4)=4 + 2 + 1= 7$

2) Soit G = (X, *U*) un graphe orienté muni de longueur C_{ij}.
Le chemin C^* entre deux sommets S et T est dit *le plus court chemin* si et seulement si :
$L(C^*) = min\ L(C)$ avec C est un chemin quelconque de S à T.

II- Algorithme de Roy-Bellman

Cet algorithme permet de trouver le plus court chemin d'un sommet fixé S vers un autre sommet T dans un graphe sans circuit. Il fonctionne en deux étapes. La première consiste à déterminer l'ordre topologique et le graphe à niveaux correspondant. Tandis que la deuxième s'intéresse au calcul de Plus Court Chemin (PCC).

1- Algorithme

Algorithme :Roy –Bellman

Début
    Mettre le graphe G sous l'ordre topologique ;
    L(S)=0
    Pour tout sommet j (selon l'ordre topologique)
        $L(j) = min\{L(i) + C_{ij}\}$, $i \in P(j)$
        $Prec(j) = \{i^*\}$
    Fin pour
Fin

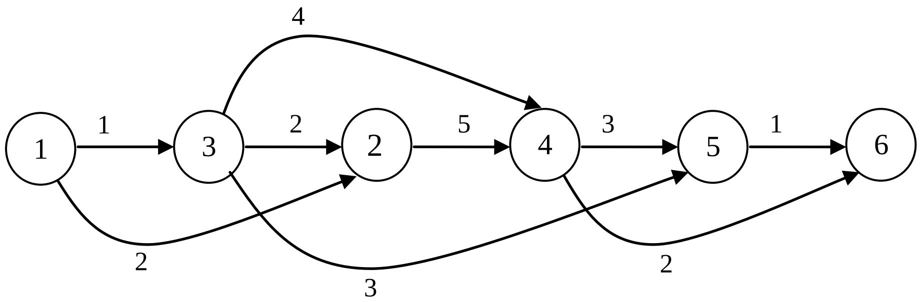
Remarque 1: la notation L(*j*) donne la longueur de PCC du sommet initial (par exemple le sommet 1) vers le sommet *j*.
Remarque 2: pour chaque itération, et parmi tous les précédents du sommet courant *i*, on garde dans la variable *Prec* celui (respectivement ceux) qui donne (respectivement qui donnent) la valeur minimale.

2- Exemples

__Exemple 1__

On se propose de déterminer le Plus Court Chemin de 1 vers 6 dans le graphe suivant.

Dans ce cas, le graphe initial est sous l'ordre topologique, on passe directement au calcul de PCC.



j=1 : L(1) = 0
j=3 : L(3) = min{L(1)+C_{13}}= L(1)+C_{13} = 0 +1 =1 $\Rightarrow$ *Prec*(3) = {1}
j=2 : L(2) = min{L(1) + C_{12}, L(3) + C_{32}} = min{2, 3} = 2 $\Rightarrow$ *Prec*(2) = {1}

Il est clair que, le sommet 2 admet les sommets 1 et 3 comme précédents. D'après la remarque 2, on garde en mémoire le sommet 1 parce qu'il donne la valeur minimale. De même, on pourra appliquer la remarque 2 pour trouver *Prec* (4), *Prec*(5) et *Prec*(6).

j=4 : L(4) = min{L(3) + C_{34}, L(2) + C_{24}}= min{5, 7} = 5 $\Rightarrow$ *Prec*(4) = {3}
j=5 : L(5) = min{L(3) + C_{35}, L(4) + C_{45}}= min{4, 8} = 4 $\Rightarrow$ *Prec*(5) = {3}
j=6 : L(6) = min{L(4) + C_{46}, L(5) + C_{56}}= min{7, 5} = 5 $\Rightarrow$ *Prec*(6) = {5}

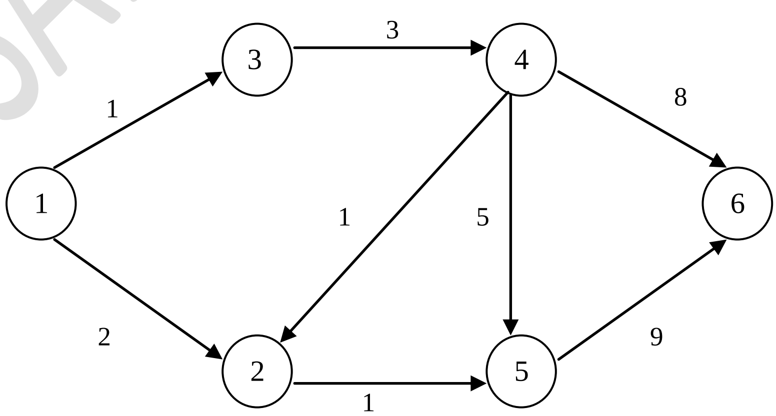
A partir de la remarque 1, on conclut immédiatement que L(C^*) =5

IL reste maintenant à définir le PCC de 1 vers 6. Pour ce faire, il suffit de construire ce chemin à l'envers. C'est-à-dire, on commence par le sommet final 6 et on utilise la variable *Prec* pour définir le sommet qui vient juste avant le sommet 6 et ainsi de suite jusqu'à atteindre le sommet initial 1.

Ainsi, on obtient le chemin optimal C^* = 1 – 3 – 5 – 6

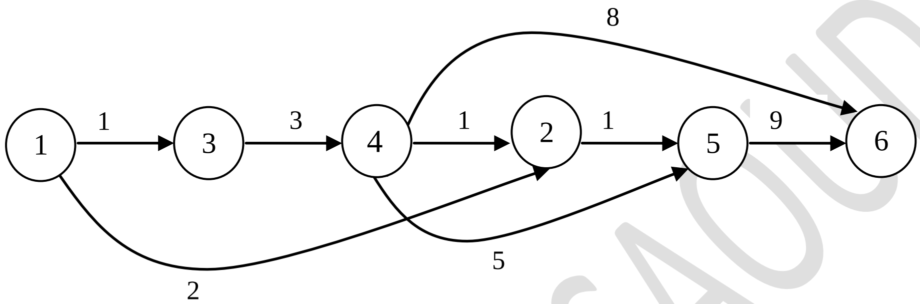
Exemple 2

Déterminer le PCC de 1 vers le sommet 6 dans le graphe suivant :



Etape 1 : ordre topologique

R_0 = {1}, T= {2, 3, 4, 5, 6}, k = 0
R1 = {3}, T= {2, 4, 5, 6}, k = 1
R2 = {4}, T= {2, 5, 6}, k = 2
R_3 = {2}, T= {5, 6}, k = 3
R_4 = {5}, T= {6}, k = 4
R_5 = {6}, T= ϕ , FIN



Etape 2 : calcul de PCC

L(1) = 0
j = 3 : L(3) = min{L(1)+C_{13}} = L(1)+C_{13}=1 $\Rightarrow$ *Prec*(3) = {1}
j = 4 : L(4) = min{L(3) + C_{34}} = L(3) + C_{34}= 4$\Rightarrow$ *Prec*(4) = {3}
j = 2 : L(2) = min{L(1) + C_{12}, L(4) + C_{42}} = min{2, 4} = 2 $\Rightarrow$ *Prec*(2) = {1}
j = 5 : L(5) = min{L(4) + C_{45}, L(2) + C_{25}) = min(9, 3) = 3 $\Rightarrow$ *Prec*(5) = {2}
j = 6 : L(6) = min{L(4) + C_{46}, L(5) + C_{56}} = min(12, 12) = 12$\Rightarrow$ *Prec*(6) = {4, 5}

Comme *Prec*(6) = {4, 5} (contient deux sommets), donc on aura (au moins) deux chemins optimaux (de même longueur).

L(C_1*) = L(C_2*) =12

On applique la même démarche utilisée dans l'exemple 1 et ce pour déduire les chemins optimaux :

C_1* = 1 – 3 – 4 – 6
C_2* = 1 – 2– 5 – 6

III- Algorithme de Dijkstra

Cet algorithme permet de déterminer les plus courts chemins d'un sommet fixé (à titre d'exemple le sommet 1) vers tous les autres sommets d'un graphe orienté uniquement dans le cas où toutes les longueurs sont positives.

Notation et principe

- On pose $\pi(j)$ la longueur du Plus Court Chemin allant du sommet 1 vers le sommet j.
- Les $\pi(j)$ sont d'abord initialisées à certaines valeurs puis elles seront modifiées au fur et à mesure de déroulement de l'algorithme.
- L'ensemble des sommets est subdivisé en deux sous ensembles disjoints :
- **P** = l'ensemble des sommets dont les valeurs $\pi(j)$ sont déterminées de manière **P**ermanente ;
- **T** = l'ensemble des sommets dont les valeurs $\pi(j)$ sont déterminées de manière **T**emporaire.
- PCC(j) : Plus Court Chemin de 1 à j.

1- Algorithme

On considère un graphe orienté formé de n sommets numérotés de 1 jusqu'à n. L'objectif consiste à chercher tous les plus courts chemins de 1 vers tous les autres sommets du graphe.

Algorithme de Dijkstra

Etape 0 : Initialisation

$\Pi(1)=0$, PCC(1)={(1,1)}

P = {1} ; T = {2, …., n}

$$\Pi(j)=\begin{cases}C_{ij} \text{ si l'arc } (1,j) \text{ existe,}\\ +\infty \text{ sinon}\end{cases} ;\ PCC(j)=\{(1,j)\}$$

Etape 1 : Fixation d'un sommet *k*

Déterminer le sommet $k \in$ T tel que $\pi(k)= \min\limits_{j\in T}\pi(j)$.

Faire P = P $\cup$ {k} et T = T \ {k}.

Si T = ϕ alors FIN (toutes les valeurs sont fixées de manière permanente).

Etape 2 : Mise à jour des $\pi(j)$

Pour tout ($j\in$ T) ET (successeur du k)

Si ($\pi(k)+C_{kj}<\pi(j)$)

$\pi(j)=\pi(k)+C_{kj}$

PCC(j) = PCC(k) $\cup$ (k,j)

Fin Si

Fin pour

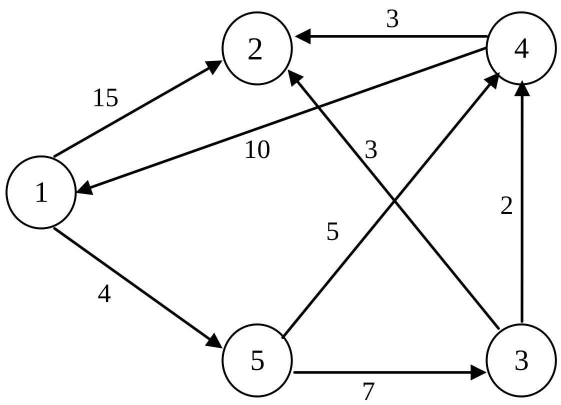
Aller à l'étape 1.

NB : Dans l'étape 1, si $\pi(j)=+\infty$, alors on initialise le PCC(j) avec l'arc fictif (1, j).

2- Exemples

<u>Exemple 1</u>

Déterminer tous les plus courts chemins de 1 vers les autres sommets du graphe ci-dessous.



Les résultats seront présentés sur le tableau suivant. La colonne « Itération0 » correspond à **l'étape 1** (initialisation) de l'algorithme. Tandis que les autres colonnes correspondent à l'exécution répétitive de **l'étape 2** et ce, jusqu'à atteindre la condition d'arrêt. Ainsi, il faut faire les calculs nécessaires pour chaque itération avant de remplir la colonne correspondante.

Sommet	**Itération0**		**Itération1**		**Itération2**		**Itération3**	
j	$\pi(j)$	PCC(*j*)	$\pi(j)$	PCC(*j*)	$\pi(j)$	PCC(*j*)	$\pi(j)$	PCC(*j*)
2	15	{(1,2)}	15	{(1,2)}	12	{(1,5,4,2)}	**<u>12</u>**	{(1,5,4,2)}
3	$+\infty$	{(1,3)}	11	{(1,5,3)}	**<u>11</u>**	{(1,5,3)}	-	-
4	$+\infty$	{(1,4)}	**<u>9</u>**	{(1,5,4)}	-	-	-	-
5	**<u>4</u>**	{(1,5)}	-	-	-	-	-	-

Itération0 : P ={1}, T = {2,3,4,5}
$\Pi(1) = 0$, PCC(1)={(1,1)}

Remarque a/ : pour fixer le sommet *k*, on souligne la valeur minimale dans la colonne de l'itération 0, et on récupère le sommet correspondant (dans ce cas, c'est le sommet **5**). Ensuite, on insère le sommet **5** dans l'ensemble P et on l'élimine de l'ensemble T. On utilisera cette même démarche pour toutes les itérations suivantes.

Remarque b/ : pour chaque itération, on donne les $\pi(j)$ et les PCC(*j*) seulement pour les sommets appartenant à l'ensemble T. On met des « - » pour les autres sommets.

Itération1 : P ={1, **5**}, T = {2, <u>3</u>, <u>4</u>}
On commence par fixer (souligner) les sommets appartenant à T et successeurs de 5.

j =3 : $\pi(5) + C_{53} = 4 + 7 = 11 < +\infty$
→ $\pi(3) = 11$ et PCC(3) = {(1,5,3)} (voir règle suivante)

Règle de mise à jour des PCC : PCC(3)= PCC(5) $\cup$(5,3)= (1,5) $\cup$(5,3)=(1,5,3).
Cette règle sera appliquée chaque fois qu'on aura besoin de faire la mise à jour de PCC.

j =4 : $\pi(5) + C_{54} = 4 + 5 = 9 < +\infty$
$\rightarrow \pi(4) = 9$ et PCC(4) = {(1,5,4)}

Itération2 : P ={1, 5, **4**}, T = {<u>2</u>, 3}
j =2 : $\pi(4) + C_{42} = 9 + 3 = 12 < 15$
$\rightarrow \pi(2) = 12$ et PCC(2) = {(1,5,4,2)}

Itération3: P ={1, 5, 4, **3**}, T = {<u>2</u>}
j =2 : $\pi(3) + C_{32} = 11+ 3 = 14 > 12$ (pas de mise à jour)

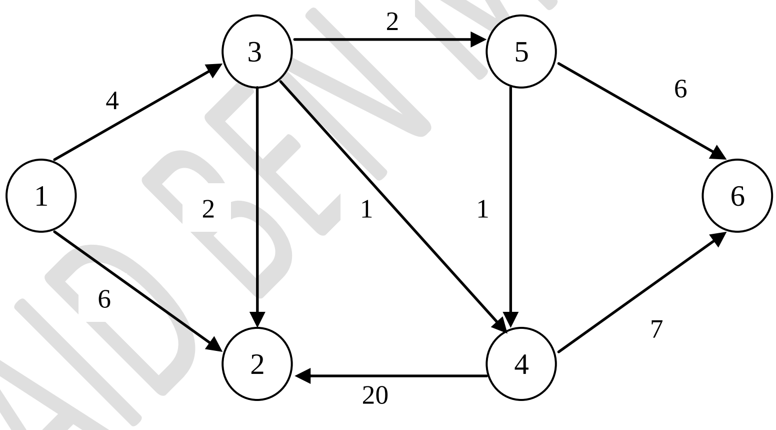
Itération4: P ={1, 5, 4, 3, **2**}, T = ϕ FIN

NB : inutile de rajouter une colonne pour l'itération 4, car elle contiendra uniquement des «-»

Conclusion : le tableau ci-dessus donne tous les plus courts chemins du sommet 1 vers tous les autres sommets du graphe.

<u>Exemple 2</u>

Déterminer tous les plus courts chemins de 1 vers les autres sommets du graphe ci-dessous.



	Itération0		Itération1		Itération2		Itération3		Itération4	
j	$\pi(j)$	PCC (j)	$\pi(j)$	PCC(j)	$\pi(j)$	PCC(j)	$\pi(j)$	PCC(j)	$\pi(j)$	PCC(j)
2	6	{(1,2)}	6	{(1,2), (1,3,6)}	**<u>6</u>**	{(1,2), (1,3,6)}	**-**	**-**	-	-
3	**<u>4</u>**	{(1,3)}	-	-	-	-	**-**	**-**	-	-
4	$+\infty$	{(1,4)}	**<u>5</u>**	{(1,3,4)}	-	-	**-**	**-**	-	-
5	$+\infty$	{(1,5)}	6	{(1,3,5)}	6	{(1,3,5)}	**<u>6</u>**	**{(1,3,5)}**	-	-
6	$+\infty$	{(1,6)}	$+\infty$	{(1,6)}	12	{(1,3,4,6)]	**12**	**{(1,3,4,6)}**	**<u>12</u>**	{(1,3,4,6), (1,3,5,6)}

Itération0 : P ={1}, T = {2,3,4,5,6}
$\Pi(1)= 0$, PCC(1)={(1,1)}

Itération1 : P ={1,**3**}, T = {<u>2</u>,<u>4</u>,<u>5</u>,6}
j =2 : $\pi(3)$ +C_{32} = 4 + 2 = 6=6
→PCC(2) = {(1,2), (1,3,2)}

Remarque c/ : Inutile de mettre à jour $\pi(2)$ parce que la nouvelle valeur est égale à l'ancienne (valeur de l'itération précédente). Par contre, on a intérêt à mettre à jour PCC(2). Dans ce cas, on garde l'ancien chemin (de l'itération précédente), et on ajoute le nouveau chemin (1,3,2). Ainsi, on obtient PCC(2) = {(1,2), (1,3,2)}.

j =4 : $\pi(3)$ +C_{34} = 4 + 1 = 5 < $+\infty$
→ $\pi(4)$ = 5 et PCC(4) = {(1,3,4)}
j =5 : $\pi(3)$ + C_{35} = 4 + 2 = 6
→ $\pi(5)$ = 6 et PCC(5) = {(1,3,5)}

Itération2 : P ={1,3,**4**}, T = {<u>2</u>,5,<u>6</u>}
j =2 : $\pi(4)$ +C_{42} = 5 + 20 = 25 > 6
Pas de mise à jour
j =6 : $\pi(4)$ + C_{46} = 5 + 7 = 12 < $+\infty$
→ $\pi(6)$ = 12 et PCC(6) = {(1,3,4,6)}

Remarque d/ On a deux valeurs identiques, on pourra alors commencer par la première.

Itération3: P ={1,3,4,**2**}, T = {5,6}
Pas de traitement, car il n'y a aucun sommet à la fois successeur de 2 et appartenant à T.

Itération4: P ={1,3,4,2,**5**}, T = {<u>6</u>}
j =6 : $\pi(5)$ + C_{56} = 6 + 6 = 12=12
→ PCC(6) = {(1,3,4,6), (1,3,5,6)}

On se retrouve dans le même cas de la **Remarque c/**. D'où on met à jour uniquement le PCC(6).

Itération5: P ={1,3,4,2,5,**6**}, T = ϕ FIN.

NB : inutile de rajouter une colonne pour l'itération 5, car elle contiendra uniquement des «-»

Conclusion : le tableau ci-dessus donne tous les plus courts chemins du sommet 1 vers tous les autres sommets du graphe.

IV- Algorithme de Ford-Bellman

Cet algorithme permet de déterminer le plus court chemin d'un sommet fixé (à titre d'exemple, le sommet 1) vers tous les autres sommets d'un graphe orienté où les longueurs sont de signe quelconques.

Notation

- P(j) : ensemble des précédents du sommet j.
- $\pi^k(j)$: longueur du plus court chemin de 1 vers j obtenue à l'itération k.
- $PCC^{k+1}(j)$: Plus Court Chemin de 1 vers j obtenue à l'itération k.

1- Algorithme : Ford-Bellman

On considère un graphe orienté formé de n sommets numérotés de 1 jusqu'à n. L'objectif consiste à chercher tous les plus courts chemins de 1 vers tous les autres sommets du graphe dans le cas où les longueurs sont de signes quelconques (positives et négatives).

.

Algorithme : Ford-Bellman
Etape 0 : initialisation

$k = 0$
$\pi^0(1) = 0$, $PCC^0(1)=\{(1,1)\}$
Pour $j = 2$ jusqu'à n

$$\pi^0(1) = \begin{cases} C_{ij} \text{ si l'arc } (1,j) \text{ existe} \\ +\infty \quad \text{sinon} \end{cases} \; ; \; PCC^0(j) = \{(1,j)\}$$

Fin Pour

Etape 1 : mise à jour des longueurs

Pour $j = 1$ jusqu'à n-1
$\pi^k(j) = min\,\{\pi^{k-1}(j), min_{l\in P(j)}\{\pi^{k-1}(l) + C_{lj}\}\}$
Mettre à jour $PCC^k(j)$
Fin pour

Etape 2 : test de convergence

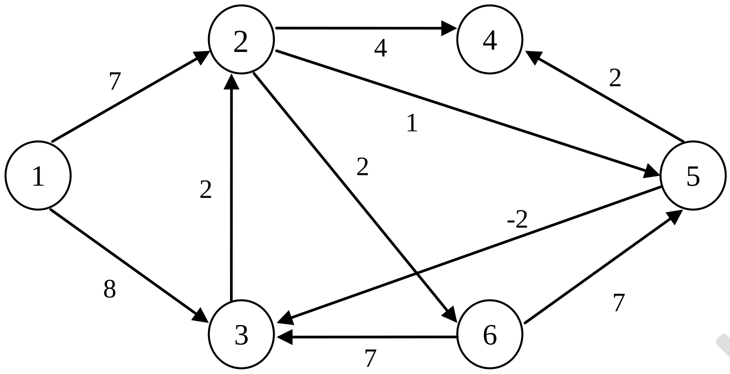
si $(\pi^k(j) = \pi^{k-1}(j), \forall\, j \in \{1, \ldots, n\})$ (deux colonnes successives identiques).
alors FIN.
sinon si $k = n$ STOP. (Pas de solution : il existe un circuit absorbant).
sinon $k = k + 1$ et revenir à l'étape 1.

Remarques :

i- un circuit absorbant est un circuit de longueur négative.
ii- la mise à jour de $PCC^k(j)$ est similaire à celle appliquée dans l'algorithme de Dijkstra.

2- Exemple

Déterminer tous les plus courts chemins de 1 vers les autres sommets dans le graphe suivant :



On a des longueurs positives et négatives, d'où on utilise l'algorithme de Ford-Bellman.

	k =0		k =1		k =2		k =3	
	$\pi^0(j)$	$PCC^0(j)$	$\pi^1(j)$	$PCC^1(j)$	$\pi^2(j)$	$PCC^2(j)$	$\pi^3(j)$	$PCC^3(j)$
1	0	{(1, 1)}	0	{(1, 1)}	0	{(1, 1)}	0	{(1, 1)}
2	7	{(1, 2)}	7	{(1, 2)}	7	{(1, 2)}	7	{(1, 2)}
3	8	{(1, 3)}	8	{(1, 3)}	6	{(1, 2 , 5 , 3)}	6	{(1, 2 , 5 , 3)}
4	$+\infty$	{(1, 4)}	11	{(1, 2, 4)}	10	{(1, 2, 5 , 4)}	10	{(1, 2, 5 , 4)}
5	$+\infty$	{(1, 5)}	8	{(1, 2, 5)}	8	{(1, 2, 5)}	8	{(1, 2, 5)}
6	$+\infty$	{(1, 6)}	9	{(1, 2, 9)}	9	{(1, 2, 6)}	9	{(1, 2, 6)}

***k*=0 :** initialisation

***k*=1 :**

j=1 : $\pi^1(1) = \pi^0(1) = 0$; $PCC^1(1) = \{(1,1)\}$

j=2 : $\pi^1(2) = min\,\{\pi^0(2), min\{\pi^0(1) + C_{12}, \pi^0(3) + C_{32}\}\}$

$= min\{7, 0 + 7, 8 + 2\} = 7$; $PCC^1(2) = \{(1,2)\}$

Remarque a : En pratique, il suffira de prendre le minimum sur tous les termes.

j=3 : $\pi^1(3) = min\,\{\pi^0(3), \pi^0(1) + C_{13}, \pi^0(5) + C_{53}, \pi^0(6) + C_{63}\}$

$= min\{8, 0 + 8, +\infty - 2, +\infty + 7\} = 8$; $PCC^1(3) = \{(1,3)\}$

j=4 : $\pi^1(4) = min\,\{\pi^0(4), \pi^0(2) + C_{24}, \pi^0(5) + C_{54}\}$

$= min\{+\infty, 7 + 4, +\infty + 2\} = 11$; $PCC^1(4) = \{(1,2,4)\}$

j=5 : $\pi^1(5) = min\,\{\pi^0(5), \pi^0(2) + C_{25}, \pi^0(6) + C_{65}\}$

$= min\{+\infty, 7 + 1, +\infty + 7\} = 8$; $PCC^1(5) = \{(1,2,5)\}$

j=6 : $\pi^1(6) = min\,\{\pi^0(6), \pi^0(2) + C_{26}\}$

$= min\{+\infty, 7 + 2\} = 9$; $PCC^1(6) = \{(1,2,6)\}$

(On remplie maintenant la colonne correspondante)

***k*=2 :**

j=1 : $\pi^2(1) = \pi^1(1) = 0$; $PCC^2(1) = \{(1,1)\}$

j=2 : $\pi^2(2) = min\,\{\pi^1(2), \pi^1(1) + C_{12}, \pi^1(3) + C_{32}\}$

$= min\{7, 0 + 7, 8 + 2\} = 7$; $PCC^2(2) = \{(1,2)\}$

j=3 : $\pi^2(3) = min\,\{\pi^1(3), \pi^1(1) + C_{13}, \pi^1(5) + C_{53}, \pi^1(6) + C_{63}\}$
$= mi\,n\{8, 0 + 8, 8 - 2, 9 + 7\} = 6$; $PCC^2(3) = \{(1{,}2{,}5{,}3)\}$
j=4 : $\pi^2(4) = min\,\{\pi^1(4), \pi^1(2) + C_{24}, \pi^1(5) + C_{54}\}$
$= mi\,n\{11, 7 + 4, 8 + 2\} = 10$; $PCC^2(4) = \{(1{,}2{,},5{,}4)\}$

j=5 : $\pi^2(5) = min\,\{\pi^1(5), \pi^1(2) + C_{25}, \pi^1(6) + C_{65}\}$
$= mi\,n\{8, 7 + 1, 9 + 7\} = 8$; $PCC^2(5) = \{(1{,}2{,}5)\}$
j=6 : $\pi^2(6) = min\,\{\pi^1(6), \pi^1(2) + C_{26}\}$
$= mi\,n\{9, 7 + 2\} = 9$; $PCC^1(6) = \{(1{,}2{,}6)\}$

(On remplie maintenant la colonne correspondante)

***k*=3 :**
j=1 : $\pi^3(1) = \pi^2(1) = 0$; $PCC^3(1) = \{(1{,}1)\}$
j=2 : $\pi^3(2) = min\,\{\pi^2(2), \pi^2(1) + C_{12}, \pi^2(3) + C_{32}\}$
$= mi\,n\{7, 0 + 7, 8 + 2\} = 7$; $PCC^3(2) = \{(1{,}2)\}$
j=3 : $\pi^3(3) = min\,\{\pi^2(3), \pi^2(1) + C_{13}, \pi^2(5) + C_{53}, \pi^2(6) + C_{63}\}$
$= mi\,n\{6, 0 + 8, 8 - 2, 9 + 7\} = 6$; $PCC^2(3) = \{(1{,}2{,}5{,}3)\}$
j=4 : $\pi^3(4) = min\,\{\pi^2(4), \pi^2(2) + C_{24}, \pi^2(5) + C_{54}\}$
$= mi\,n\{10, 7 + 4, 8 + 2\} = 10$; $PCC^3(4) = \{(1{,}2{,},5{,}4)\}$
j=5 : $\pi^3(5) = min\,\{\pi^2(5), \pi^2(2) + C_{25}, \pi^2(6) + C_{65}\}$
$= mi\,n\{8, 7 + 1, 9 + 7\} = 8$; $PCC^3(5) = \{(1{,}2{,}5)\}$
j=6 : $\pi^3(6) = min\,\{\pi^2(6), \pi^2(2) + C_{26}\}$
$= mi\,n\{9, 7 + 2\} = 9$; $PCC^3(6) = \{(1{,}2{,}6)\}$

(On remplie maintenant la colonne correspondante)

Conclusion : On constate que $\pi^3(j) = \pi^2(j), \forall\, j \in \{1, \dots, n\}$, d'où la condition d'arrêt est vérifiée. En pratique, il suffit de vérifier qu'on a obtenu deux colonnes successives identiques (de point de vue longueur). Ainsi, le tableau ci-dessus donne tous les plus courts chemins du sommet 1 vers tous les autres sommets du graphe.