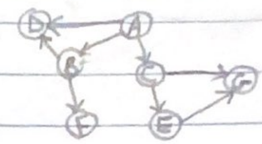
$i = 0 \quad k = 0$

Parcours En profondeur

Ex : voir (exo 1) SN2

```
DFSrecursif (G, x)
{   etat[x] = 1;   // i++; Prefix[x] = i
    ∀y ∈ Γ+(x)
    {  si (etat[y] = 0)
            DFSrecursif(G, y)  // k++; suffix[x] = k
    }
}
```

Sommet	Prefix[x]	Suffix[x]
x = A	1	7
B	2	5
C	7	6
D	3	1
E	5	3
F	4	4
G	6	2



(graph 1)

Parcours En largeur

Entrées : G : graphe, S : un sommet, F : file

```
debut
  enfiler (F, S)
  Tant que (F ≠ ∅)
  {  X = S
     defiler (F)
     ∀y ∈ Γ+(x)
     {  si (etat[y] = 0)
           {  etat[y] = 1
              enfiler (F, y)
           }
     }
  }
Fin
```

Note : on affiche juste apres X =

Ex : (graph 1)

enfiler(F, A)

X = A
defiler (F)
| B | C | D |

X = B
defiler(F)
| C | D | F |

X = C
defiler (F)
| D | F | E | G |

(suite)

X = D
defiler (F)
| F | E | G |

X = F
Defiler(F)
| E | G |

X = E
Defiler(F)
| G |

X = G
defiler(F)

on arret à A, car déjà visité

print : A B C D F E G
1 2 3 4 5 6 7

Comment trouver l'arbre couvrant de poids minimum ?

Algorithme de Kruskal

Entrée : U : les arcs d'un graphe trié par ordre croissant

Sortie : T : les arcs de l'arbre couvrant de poids minimum

$T = \emptyset$

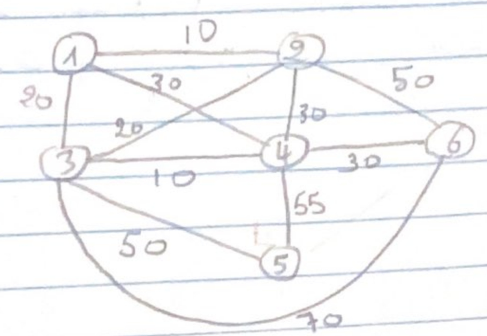
$\forall u \in U$

si ($T \cup \{u\}$ ne crée pas un cycle)

$T = T \cup \{u\}$;

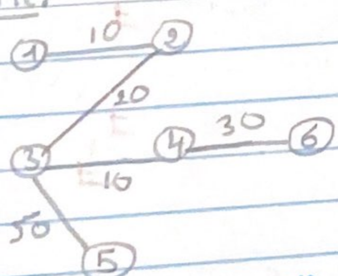
arc	Poids	
{1,2}	10	✓
{3,4}	10	✓
{3,2}	20	✓
{1,3}	20	✗
{2,4}	30	✗
{1,4}	30	✗
{4,6}	30	✓
{2,6}	50	✗
{3,5}	50	✓
{4,5}	55	✗
{5,6}	60	✗
{3,6}	70	✗

Ex :



(graphe 2)

Sortie :



Algorithme de PRIM

Entrées : U : les arcs d'un graphe ; x_0 : sommet quelconque du graphe

n : nombre de sommet du graphe

Sorties : T : les arcs de l'arbre couvrant ; S : sommets de l'arbre couvrant

$T = \emptyset$; $S = \{x_0\}$; $k = 1$; // k nbr sommet de l'arbre couvrant

Tantque ($k < n$)

{ choix d'une arrête $u_0 = \{x_0, y_0\}$ tq $x_0 \in S$ et $y_0 \notin S$

Poids → $coût(u_0) = \min(coût(x_0, y_0))$

$T = T \cup \{u_0\}$

$S = S \cup \{y_0\}$

k++

}

Ex : appliquer PRIM sur (graphe 2)

$x_0 = 4$ $S = \{4\}$ $T = \emptyset$; $k = 1$

$x_0 = 3$ $S = \{4,3\}$ $T = \{(4,3)\}$; $k = 2$

$x_0 = 2$ $S = \{4,3,2\}$ $T = \{(4,3),(2,3)\}$; $k = 3$

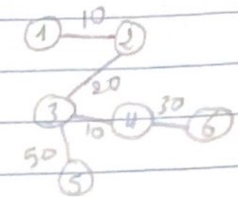
$x_0 = 3$

$x_0 = 2$ $S = \{4,3,2,1\}$ $T = \{(4,3),(2,3),(1,2)\}$; $k = 4$

$x_0 = 3$ $S = \{4,3,2,1,5\}$ $T = \{(4,3),(2,3),(1,2),(3,5)\}$; $k = 5$

$x_0 = 4$ $S = \{4,3,2,1,5,6\}$ $T = \{(4,3),(2,3),(1,2),(3,5),(4,6)\}$; $k = 6$

Sortie :



$10+20+10+30+50$
$= 120$
coût(arbre) = 120

Variant d'algorithme de Kruskal

Entrée : U : les arcs d'un graphe trié par ordre décroissant

Sortie : T : les arcs de l'arbre couvrant de poids minimum

$T = U$

$\forall u \in U$

si($(X, T\setminus\{u\})$ est connexe)

$T = T\setminus\{u\}$

Plus cours chemin de S a tous les sommets

Algorithme de Dijkstra

$S = \{s\}$; $\Pi(s) = 0$; $\forall x \neq s \Rightarrow \Pi(x) = \infty$; $k = 1$; $x_k = s$;

Tant que $(k < n)$ faire

{

Pour tout u tq $I(u) = x_k$ et $T(u) \notin S$ faire

{ si $(\Pi(T(u)) > \Pi(x_k) + d(u))$ alors

{ $\Pi(T(u)) = \Pi(x_k) + d(u)$

$A(T(u)) = u$

}

}

Choisir $x \notin S$ tq $\Pi(x) = \min_{y \notin S}(\Pi(y))$

$k{+}{+}$; $x_k = x$; $S = S \cup \{x_k\}$;

}

S : sommets traités

Π : potentiel (Poids cummulé)

$I(u)$: sommet initial de l'arc u

$T(u)$: sommet terminal de l'arc u

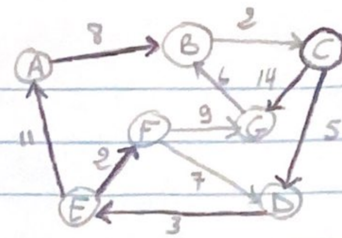
x_k : sommet pivot

k : nbr sommet dans S

n : nbr sommet du graphe

$A(t) = (i,t)$
arc i à t $i \rightarrow t$

Ex : appliquer Djikistra



n=7

S={C,G,D,E,F,A,B}

x	C	D	G	E	F	A	B		
	$\boxed{0}$	∞	∞	∞	∞	∞	∞	K=1	x_k = C
		$\boxed{5}$	14	∞	∞	∞	∞	K=2	x_k = D
			14	$\boxed{8}$	∞	∞	∞	K=3	x_k = E
			14		$\boxed{10}$	19	∞	K=4	x_k = F
			$\boxed{14}$			19	∞	K=5	x_k = G
						$\boxed{19}$	20	K=6	x_k = A
							$\boxed{20}$	K=7	x_k = B

x				
C	-	-	-	-
D	(C,D)	(C,D)	(C,D)	(C,D)
G	(C,G)	(C,G)	(C,G)	(C,G)
E	-	(D,E)	(D,E)	(D,E)
F	-	-	(E,F)	(E,F)
A	-	-	(E,A)	(E,A)
B	-	-	-	(A,B)

Sortie :

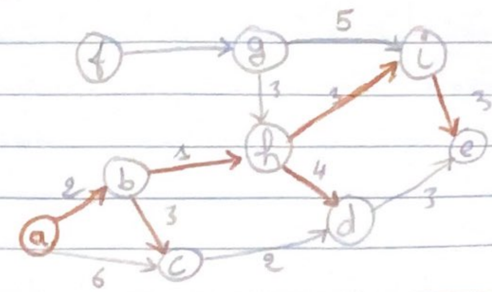
S = { C, G, D, E, F, A, B }

A(u) = { (C,D), (C,G), (D,E), (E,F), (E,A), (A,B) }

Algorith de Bellman-Ford

```
Π(s) = 0 ; ∀x ∈ X , x ≠ s   Π(x) = ∞ ; Compteur = 0
Repeter
{  Fini = vrai ;
   compteur++ ;
   Pour tout u = (x,y) ∈ U faire
   {  si ( Π(y) > Π(x) + d(u) ) alors
        {  Π(y) = Π(x) + d(u) ;
           Fini = Faux ;
        }
   }
} Jusqu'à ( compteur ≥ n-1  ou Fini = vrai )
Pour tous u = (x,y) ∈ U faire
{   si ( Π(x) + d(u) < Π(y) ) alors
        retourner faux   // présence d'un circuit
}
retourner Vrai
```

Ex : appliquer Bellman-Ford



$S' = \{a\}$ $x = a$

$y = b$

$S' = \{a, b\}$

$\Pi(b) = \min(\Pi(b), \Pi(x) + d(x,y))$

$\Pi(b) = 2$

$y = c$; $S' = \{a, b, c\}$

$\Pi(c) = \min(\Pi(c), \Pi(x) + d(x,y))$

$\Pi(c) = 6$

$x = b$

$y = c$

$\Pi(c) = \min(\Pi(c), \Pi(x) + d(x,y))$

$\Pi(c) = 5$

$y = h$; $S' = \{a, b, c, h\}$

$\Pi(h) = \min(\Pi(h), \Pi(x) + d(x,y))$

$\Pi(h) = 3$

$x = h$

$y = d$ $S' = \{a, b, c, h, d\}$

$\Pi(d) = \min(\Pi(d), \Pi(x) + d(x,y))$

$\Pi(d) = 7$

$y = i$ $S' = \{a, b, c, h, d, i\}$

$\Pi(i) = \min(\Pi(i), \Pi(x) + d(x,y))$

$\Pi(i) = 6$

$x = i$

$y = e$ $S' = \{a, b, c, h, d, i, e\}$

$\Pi(e) = \min(\Pi(e), \Pi(x) + d(x,y))$

$\Pi(e) = 9$

Note : (f) et (g) ne sont pas reliés au sommet (a)

x	a	b	c	d	e	h	i
Π	0	2	5	7	9	3	6

x	a	b	c	d	e	h	i
μ	–	(a,b)	(b,c)	(h,d)	(i,e)	(b,h)	(h,i)