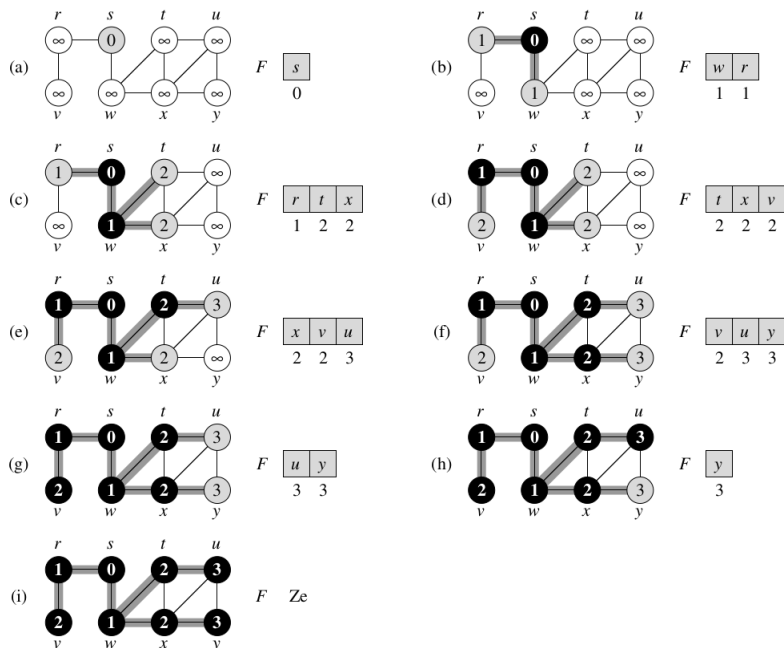


Implémentation des graphes (synthèse):

- le parcours en largeur d'abord: (https://www.youtube.com/watch?v=NrQGxfFMYzs&ab_channel=%C3%80lad%C3%A9couvertedesgraphes)

```

PL( $G, s$ )
1  pour chaque sommet  $u \in S[G] - \{s\}$ 
2    faire  $couleur[u] \leftarrow$  BLANC
3     $d[u] \leftarrow \infty$ 
4     $\pi[u] \leftarrow$  NIL
5   $couleur[s] \leftarrow$  GRIS
6   $d[s] \leftarrow 0$ 
7   $\pi[s] \leftarrow$  NIL
8   $F \leftarrow \{s\}$ 
9  tant que  $F \neq \emptyset$ 
10   faire  $u \leftarrow$  tête[ $F$ ]
11     pour chaque  $v \in Adj[u]$ 
12       faire si  $couleur[v] =$  BLANC
13         alors  $couleur[v] \leftarrow$  GRIS
14            $d[v] \leftarrow d[u] + 1$ 
15            $\pi[v] \leftarrow u$ 
16           ENFILE( $F, v$ )
17           DÉFILE( $F$ )
18    $couleur[u] \leftarrow$  NOIR
    
```



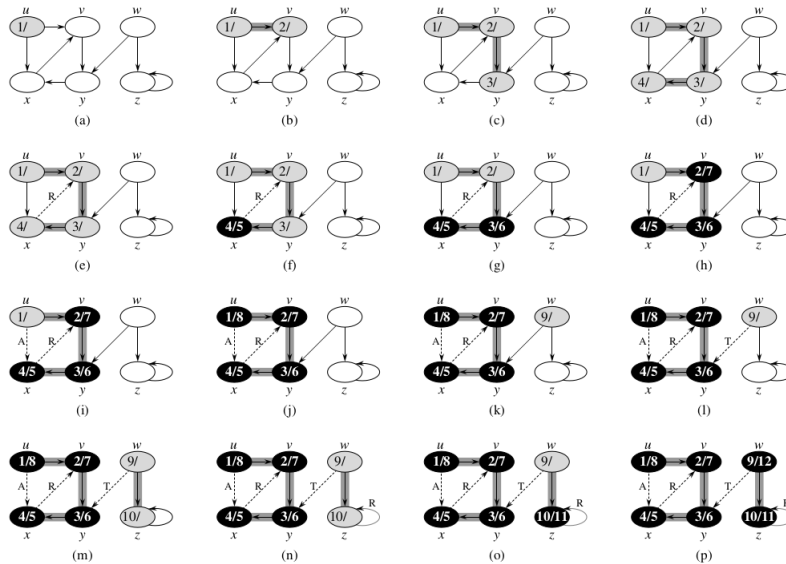
2. le parcours en profondeur d'abord: (https://www.youtube.com/watch?v=kcedjJ0jDpg&ab_channel=%C3%80lad%C3%A9couvertedesgraphes)

PP(G)

- 1 **pour** chaque sommet $u \in S[G]$
- 2 **faire** $couleur[u] \leftarrow$ BLANC
- 3 $\pi[u] \leftarrow$ NIL
- 4 $date \leftarrow 0$
- 5 **pour** chaque sommet $u \in S[G]$
- 6 **faire si** $couleur[u] =$ BLANC
- 7 **alors** VISITER-PP(u)

VISITER-PP(u)

- 1 $couleur[u] \leftarrow$ GRIS \triangleright sommet blanc u vient d'être découvert.
- 2 $date \leftarrow date + 1$
- 3 $d[u] \leftarrow date$
- 4 **pour** chaque $v \in Adj[u] \triangleright$ Exploration de l'arc (u, v) .
- 5 **faire si** $couleur[v] =$ BLANC
- 6 **alors** $\pi[v] \leftarrow u$
- 7 VISITER-PP(v)
- 8 $couleur[u] \leftarrow$ NOIR \triangleright noircir u , car on en a fini avec lui.
- 9 $f[u] \leftarrow date \leftarrow date + 1$



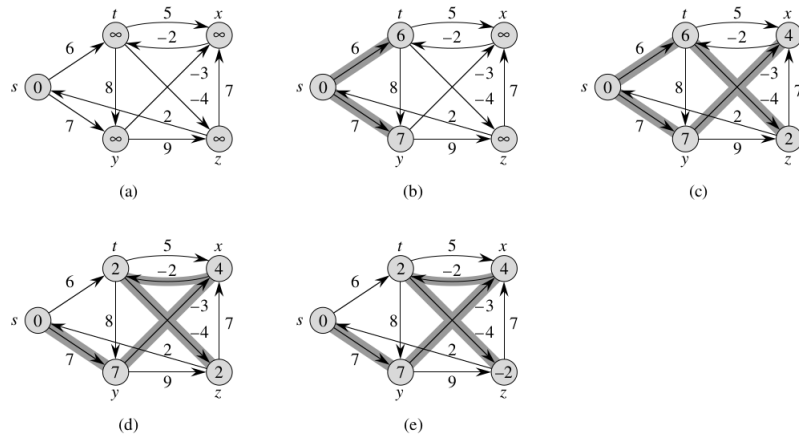
3. Recherche d'une chaîne dans un graphe

3.1 Algorithme de Bellman-Ford: (<https://www.youtube.com/watch?v=Pn7rCyqwZbQ>)

BELLMAN-FORD(G, w, s)

```

1 SOURCE-UNIQUE-INITIALISATION( $G, s$ )
2 pour  $i \leftarrow 1$  à  $|S[G]| - 1$ 
3   faire pour chaque arc  $(u, v) \in A[G]$ 
4     faire RELÂCHER( $u, v, w$ )
5 pour chaque arc  $(u, v) \in A[G]$ 
6   faire si  $d[v] > d[u] + w(u, v)$ 
7     alors retourner FAUX
8 retourner VRAI
  
```



3.2 Algorithme de Dijkstra: (<https://youtu.be/JPeCmKFrKio>)

DIJKSTRA(G, w, s)

```

1 SOURCE-UNIQUE-INITIALISATION( $G, s$ )
2  $E \leftarrow \emptyset$ 
3  $F \leftarrow S[G]$ 
4 tant que  $F \neq \emptyset$ 
5   faire  $u \leftarrow$  EXTRAIRE-MIN( $F$ )
6    $E \leftarrow E \cup \{u\}$ 
7   pour chaque sommet  $v \in Adj[u]$ 
8     faire RELÂCHER( $u, v, w$ )
  
```

