

```

function BFS( $G = (V, E)$  : listes d'adjacence,  $s$  : sommet)
  for chaque sommet  $u \in V - \{s\}$  do
    Couleur[u] = blanc
    Distance[u] =  $\infty$ 
    Parent[u] = NIL
  end for
  Couleur[s] = gris
  Distance[s] = 0
  Parent[s] = NIL
   $F = \{s\}$ 
  while  $F \neq \emptyset$  do
     $u = \text{tete}(F)$ 
     $F = F - \{u\}$ 
    for chaque  $v \in \mathcal{N}(u)$  do
      if Couleur[v] = blanc then
        Couleur[v] = gris
        Distance[v] = Distance[u] + 1
        Parent[v] = u
         $F = F + \{v\}$ 
      end if
    end for
    Couleur[u] = rouge
  end while
end function

```

```
function DFS( $G = (V, E)$  : listes d'adjacence,  $s$  : sommet)
   $Z = \{s\}$ 
   $P = \{s\}$ 
  while  $P \neq \emptyset$  do
     $v =$  sommet de la pile  $P$ 
    if il existe  $u \notin Z$  et  $u \in \mathcal{N}(v)$  then
       $Z = Z + \{u\}$ 
       $P = P + \{u\}$ 
    else
       $P = P - \{v\}$ 
    end if
  end while
end function
```

```
function PRIM( $G = (V, E, W)$  : graphe valué ,  $s$  : sommet)
   $F = V$ 
  for chaque sommet  $v$  do
     $d[v] = \infty$ 
  end for
   $d[s] = 0$ 
   $\text{parent}[s] = \text{NIL}$ 
  while  $F \neq \emptyset$  do
     $u = \text{Min}(F)$ 
    for chaque  $v \in \mathcal{N}(u)$  do
      if  $v \in F$  et  $W(u, v) < d[v]$  then
         $\text{Parent}[v] = u$ 
         $d[v] = W(u, v)$ 
      end if
    end for
  end while
end function
```