

Prim's Algorithm

```
procedure prim(G,W,s)

  for each vertex  $v \in V[G]$  do
     $d[v] \leftarrow \infty$ 
     $\pi[v] \leftarrow \text{NIL}$ 
  end for
  outside  $\leftarrow V[G]$ 

   $d[s] \leftarrow 0$ 
  while outside  $\neq \phi$  do
     $u \leftarrow \text{Extract\_Min}(\text{outside})$ 
    for each  $v$  adjacent to  $u$  do
      if  $v \in \text{outside}$  and  $W[u,v] < d[v]$  then
         $d[v] \leftarrow W[u,v]$ 
         $\pi[v] \leftarrow u$ 
      end if
    end for
  end while

end procedure
```

Prim's Algorithm Dense Graphs

```
procedure prim(G,W)

  for i = 1 to n do
    d[i] ← ∞
    outside[i] ← true
    π[i] ← NIL
  end for
  d[0] ← ∞

  d[1] ← 0
  for i = 1 to n do
    k ← 0
    for j = 1 to n do if outside[j] and d[j] ≤ d[k] then k ← j
    outside[k] := false
    for j = 1 to n do if outside[j] and W[j,k] < d[j] then
      d[j] ← W[j,k]
      π[j] ← k
    end for
  end for

end procedure
```

Prim's Algorithm Sparse Graphs

```
procedure prim(G,W)

  for i = 1 to n do
    MinHeap[i] ← i
    WhereInHeap[i] ← i
    d[i] ← ∞
    outside[i] ← true
    π[i] ← NIL
  end for

  d[1] ← 0
  for i = n downto 1 do
    u ← MinHeap[1]
    MinHeap[1] ← MinHeap(n)
    WhereInHeap[MinHeap[1]] ← SiftDown(MinHeap,1,n-1,d)
    for each v ∈ adj[u] do
      if v ∈ outside and W[u,v] < d[v] then
        d[v] ← W[u,v]
        π[v] ← u
        WhereInHeap[v] ← SiftUp(MinHeap,WhereInHeap[v],d)
      end if
    end for
  end for

end procedure
```