```
ShortestPathThroughDAG(G,startNode)
   for I in 0 .. |V|-1 do
      d[I] := POSITIVE_INFINITY    # costs
      p[I] := NULL                 # predecessors
   od
   d[startNode] := 0

   topSortedNodes := topSort(G)

   foreach node n in topSortedNodes do
      foreach node i that n is adjacent to do
         if w(<i,n>) + d[i] < d[n]
            d[n] := w(<i,n>) + d[i]
            p[n] := i
         fi
      od
   od
   return d, p
end

Backtrack(p,endNode)
   node := endNode
   path := empty path
   while node != null do
      append node to start of path
      node := p[node]
   od
   return path
end
```