```
bool TDMSC-I (DJ-Graph )
{
  RequireAnotherPass = false
  while (in B(readth) F(irst) S(earch) order) do {
    Let n =  Next Node in BFS list
    for ( all incoming edges to n ) do {
      Let e = <snode,tnode> be the Incoming edge
      if ( e  is a J-edge ∧ e not visited) then {
        Visit(e)
        Let tmp    =  snode
        Let lnode  =  NULL
        while ( level(tmp) ≥ level(tnode) ) do {
          Merge(tmp) = Merge(tmp) ∪ Merge(tnode) ∪ {tnode}
          lnode = tmp
          tmp = parent(tmp) //dominator tree parent
        } // end while
        for (all incoming nodes to lnode) do { // lnode ancestor of snode
          Let e' = Incoming edge
          if (e' is a J-edge ∧ e' visited) then {
            Let snode' = Source Node of e'
            if ( Merge(snode') ≥ Merge(lnode) ) then { //Check inconsistency
              RequireAnotherPass = true
            } // end if
          } // end if
        } // end for
      } // end if
    } // end for
  } // end while
 return RequireAnotherPass
}
```