

**Algorithm Design**  
*M. T. Goodrich and R. Tamassia*  
John Wiley & Sons  
**Solution of Exercise C-2.16**

**Algorithm** eulerTour(Tree  $T$ , Position  $v$ ):

```

state ← start
while state ≠ done do
  if state = start then
    if  $T.isExternal(v)$  then
      left action
      below action
      right action
      state ← done
    else
      left action
      state ← on_the_left
       $v \leftarrow v.leftchild$ 
  if state = on_the_left then
    if  $T.isExternal(v)$  then
      left action
      below action
      right action
      state = from_the_left
       $v \leftarrow v.parent$ 
    else
      left action
       $v \leftarrow v.leftchild$ 
  if state = from_the_left then
    below action
    state ← on_the_right
     $v \leftarrow v.right$ 
  if state = on_the_right then
    if  $T.isExternal(v)$  then
      state = from_the_right
      left action
      below action
      right action
       $v \leftarrow v.parent$ 
    else
      left action
      state ← on_the_left
       $v \leftarrow v.left$ 
  if state = from_the_right then
    right action
    if  $T.isRoot(v)$  then
      state ← done
    else
      if  $v$  is left child of parent then
        state ← from_the_left
      else

```