

**Algorithm Design**  
***M. T. Goodrich and R. Tamassia***  
John Wiley & Sons  
**Solution of Exercise C-2.9**

**Algorithm** preorderNext(Node  $v$ ):

```
if visInternal() then
    return  $v$ 's left child
else
    Node  $p$  = parent of  $v$ 
    if  $v$  is left child of  $p$  then
        return right child of  $p$ 
    else
        while  $v$  is not left child of  $p$  do
             $v = p$ 
             $p = p.parent$ 
        return right child of  $p$ 
```

**Algorithm** inorderNext(Node  $v$ ):

```
if visInternal() then
    return  $v$ 's right child
else
    Node  $p$  = parent of  $v$ 
    if  $v$  is left child of  $p$  then
        return  $p$ 
    else
        while  $v$  is not left child of  $p$  do
             $v = p$ 
             $p = p.parent$ 
        return  $p$ 
```

The worst case running times for these algorithms are all  $O(\log n)$  where  $n$  is the height of the tree  $T$ .

**Algorithm** postorderNext(Node  $v$ ):

```
if  $v$  is Internal() then  
     $p$  = parent of  $v$   
    if  $v$  = right child of  $p$  then  
        return  $p$   
    else  
         $v$  = right child of  $p$   
        while  $v$  is not external do  
             $v$  = left child of  $v$   
        return  $v$   
else  
     $p$  = parent of  $v$   
    if  $v$  is left child of  $p$  then  
        return right child of  $p$   
    else  
        return  $p$ 
```