

**Algorithm Design**  
***M. T. Goodrich and R. Tamassia***  
John Wiley & Sons  
**Solution of Exercise C-2.13**

Examining the Euler tree traversal, we have the following method for finding  $\text{tourNext}(v, \alpha)$

- If  $v$  is a leaf, then:
  - if  $\alpha = \mathbf{left}$ , then  $w$  is  $v$  and  $\beta = \mathbf{below}$ ,
  - if  $\alpha = \mathbf{below}$ , then  $w$  is  $v$  and  $\beta = \mathbf{right}$ ,
  - if  $\alpha = \mathbf{right}$ , then
    - \* if  $v$  is a left child, then  $w$  is  $v$ 's parent and  $\beta = \mathbf{below}$ ,
    - \* if  $v$  is a right child, then  $w$  is  $v$ 's parent and  $\beta = \mathbf{right}$ .
- If  $v$  is internal, then:
  - if  $\alpha = \mathbf{left}$ , then  $\beta = \mathbf{left}$  and  $w$  is  $v$ 's left child,
  - if  $\alpha = \mathbf{below}$ , then  $\beta = \mathbf{left}$  and  $w$  is  $v$ 's right child,
  - if  $\alpha = \mathbf{right}$ , then
    - \* if  $v$  is a left child, then  $\beta = \mathbf{below}$  and  $w$  is  $v$ 's parent,
    - \* if  $v$  is a right child, then  $\beta = \mathbf{right}$  and  $w$  is  $v$ 's parent.

For every node  $v$  but the root, we can find whether  $v$  is a left or right child, by asking “ $v = T.\text{leftChild}(T.\text{parent}(v))$ ”. The complexity is always  $O(1)$ .