23. Trees :: Binary Search Trees :: *find*()

With *findNode*() implemented, implementing *find*() is trivial:

```
Method BST.find(In: key: K) Returns E
  Node<K, E> node = findNode(this.root, key)
  If node is not null and node.key = key Then
      Return node.data
  Else
    Return null
  End If
End Method BST.find
```

23. Trees :: Binary Search Trees :: *add*()

add() adds a new element containing *data* with key *key* into the BST at the proper location based on *key*. If an element with key *key* already exists in the BST, the data in that element will be replaced with *data*:

```
Method BST.add(In: key: K; In: data: E) Returns Nothing
  Node<K, E> node = findNode(this.root, key)
  If key == node.key Then
    node.data = data
  Else
    Node<K, E> newNode = new Node<K, E>(key, data)
    If node is null Then
      this.root = newNode
    ElseIf key < node.key Then
      node.leftChild = newNode
    ElseIf
      node.rightChild = newNode
    End If
  End If
End Method BST.add
```