

## Problem 2: Search given Key in BST

class Node:

```
def __init__(self, key):  
    self.key = key  
    self.left = None  
    self.right = None
```

class BST:

```
def __init__(self):  
    self.root = None
```

```
def insert(self, key):  
    self.root = self._insert_recursive(self.root, key)
```

```
def _insert_recursive(self, root, key):  
    if root is None:  
        return Node(key)  
    if key < root.key:  
        root.left = self._insert_recursive(root.left, key)  
    elif key > root.key:  
        root.right = self._insert_recursive(root.right, key)  
    return root
```

```
def search(self, key):  
    return self._search_recursive(self.root, key)
```

```
def _search_recursive(self, root, key):  
    if root is None or root.key == key:  
        return root  
    if key < root.key:  
        return self._search_recursive(root.left, key)
```

```
return self._search_recursive(root.right, key)
```

```
bst = BST()
```

```
bst.insert(8)
```

```
bst.insert(3)
```

```
bst.insert(10)
```

```
bst.insert(1)
```

```
bst.insert(6)
```

```
bst.insert(14)
```

```
bst.insert(4)
```

```
bst.insert(7)
```

```
bst.insert(13)
```

```
key_to_search = 6
```

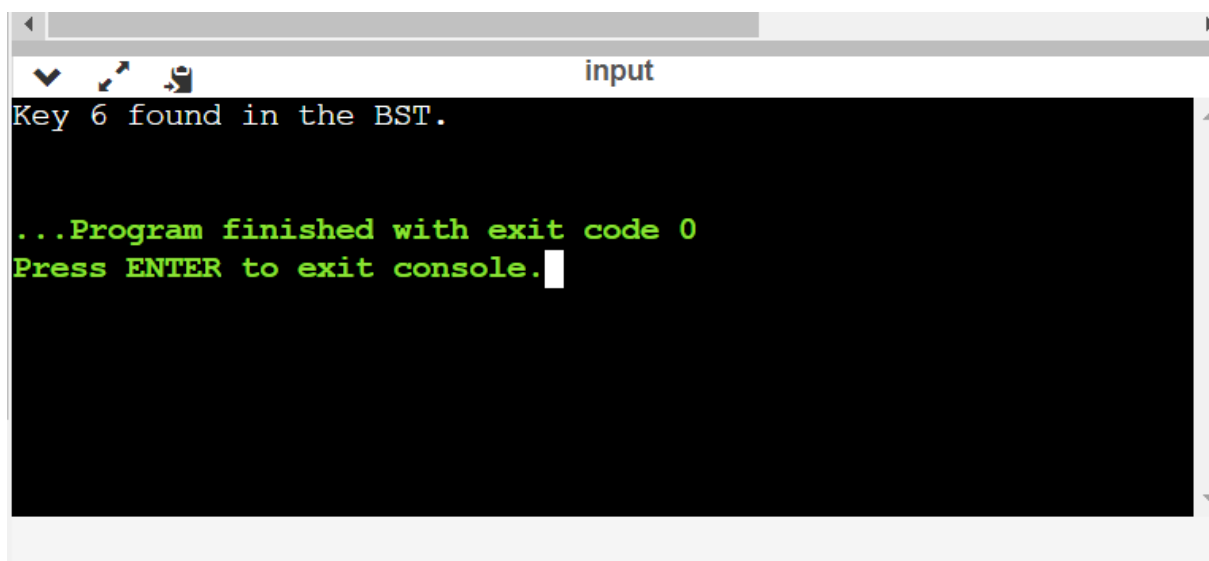
```
result = bst.search(key_to_search)
```

```
if result:
```

```
    print(f"Key {key_to_search} found in the BST.")
```

```
else:
```

```
    print(f"Key {key_to_search} not found in the BST.")
```



```
Key 6 found in the BST.  
  
...Program finished with exit code 0  
Press ENTER to exit console.
```