Exercise:8

class Node:

def \_\_init\_\_(self, time, name):

self.time = time

self.name = name

self.left = None

self.right = None

class BST:

def \_\_init\_\_(self):

self.root = None

def insert(self, root, time, name):

if root is None:

return Node(time, name)

if time < root.time:

root.left = self.insert(root.left, time, name)

else:

root.right = self.insert(root.right, time, name)

return root

def inorder(self, root):

if root:

self.inorder(root.left)

print(root.time, "-", root.name)

self.inorder(root.right)

def search(self, root, time):

if root is None or root.time == time:

return root

if time < root.time:

return self.search(root.left, time)

return self.search(root.right, time)

def minValueNode(self, node):

if node.left:

return self.minValueNode(node.left)

return node

def delete(self, root, time):

if root is None:

return root

if time < root.time:

root.left = self.delete(root.left, time)

elif time > root.time:

root.right = self.delete(root.right, time)

else:

if root.left is None:

return root.right

if root.right is None:

return root.left

temp = self.minValueNode(root.right)

root.time, root.name = temp.time, temp.name

root.right = self.delete(root.right, temp.time)

return root

def count(self, root):

return 0 if root is None else 1 + self.count(root.left) + self.count(root.right)

bst = BST()

def menu():

print("\n1.Insert 2.Delete 3.Search 4.Traverse 5.Count 6.Exit")

ch = int(input("Enter choice: "))

if ch == 1:

t = int(input("Enter entry time: "))

n = input("Enter visitor name: ")

bst.root = bst.insert(bst.root, t, n)

if ch == 2:

t = int(input("Enter entry time to delete: "))

bst.root = bst.delete(bst.root, t)

if ch == 3:

t = int(input("Enter entry time to search: "))

res = bst.search(bst.root, t)

print("Found:", res.name if res else "Not found")

if ch == 4:

print("Log Book Entries:")

bst.inorder(bst.root)

if ch == 5:

print("Total Entries:", bst.count(bst.root))

if ch == 6:

return

menu()

menu()

output:

1.Insert 2.Delete 3.Search 4.Traverse 5.Count 6.Exit

Enter choice: 1

Enter entry time: 35

Enter visitor name: ameera

1.Insert 2.Delete 3.Search 4.Traverse 5.Count 6.Exit

Enter choice: 1

Enter entry time: 45

Enter visitor name: renitha

1.Insert 2.Delete 3.Search 4.Traverse 5.Count 6.Exit

Enter choice: 1

Enter entry time: 55

Enter visitor name: varsha

1.Insert 2.Delete 3.Search 4.Traverse 5.Count 6.Exit

Enter choice: 2

Enter entry time to delete: 55

1.Insert 2.Delete 3.Search 4.Traverse 5.Count 6.Exit

Enter choice: 3

Enter entry time to search: 35

Found: ameera

1.Insert 2.Delete 3.Search 4.Traverse 5.Count 6.Exit

Enter choice: 4

Log Book Entries:

35 - ameera

45 - renitha

1.Insert 2.Delete 3.Search 4.Traverse 5.Count 6.Exit

Enter choice: 5

Total Entries: 2

1.Insert 2.Delete 3.Search 4.Traverse 5.Count 6.Exit

Enter choice: 6