Деревья

class Node:  
 def \_\_init\_\_(self, data):  
 self.left = None  
 self.right = None  
 self.data = data  
  
 def insert(self, data):  
 if self.data:  
 if data < self.data:  
 if self.left is None:  
 self.left = Node(data)  
 else:  
 self.left.insert(data)  
 elif data > self.data:  
 if self.right is None:  
 self.right = Node(data)  
 else:  
 self.right.insert(data)  
 else:  
 self.data = data  
  
 def printTree(self):  
 if self.left:  
 self.left.printTree()  
 print(self.data)  
 if self.right:  
 self.right.printTree()  
  
 def search(self, find):  
 if find < self.data:  
 if self.left is None:  
 return str(find) + " не найдено"  
 return self.left.search(find)  
 elif find > self.data:  
 if self.right is None:  
 return str(find) + " не найдено"  
 return self.right.search(find)  
 else:  
 return str(self.data) + ' найдено'  
  
  
root = Node(int(input("введите корень= ")))  
arr = input("введите ветви = ").split()  
for i in range(len(arr)):  
 root.insert(int(arr[i]))  
print("двоичное дерево= ")  
root.printTree()  
  
a = int(input("введите иск. число= "))  
print(root.search(a))  
  
while (a != ""):  
 a = input("введите иск. число = ")  
 if (a != ""):  
 print(root.search(int(a)))

