```
START Create an empty binary search tree (BST) FUNCTION insertContact(root, name, phone): IF root is
NULL THEN root = createNewNode(name, phone) ELSE IF name < root.name THEN root.left =
insertContact(root.left, name, phone) ELSE root.right = insertContact(root.right, name, phone) ENDIF RETURN
root END FUNCTION FUNCTION searchContact(root, name): IF root is NULL THEN RETURN "Contact not
found" ELSE IF name == root.name THEN RETURN root.phone ELSE IF name < root.name THEN RETURN
searchContact(root.left, name) ELSE RETURN searchContact(root.right, name) ENDIF END FUNCTION
FUNCTION inorderTraversal(root): IF root is NOT NULL THEN inorderTraversal(root.left) DISPLAY
root.name + ": " + root.phone inorderTraversal(root.right) ENDIF END FUNCTION FUNCTION
deleteContact(root, name): IF root is NULL THEN RETURN root ELSE IF name < root.name THEN root.left =
deleteContact(root.left, name) ELSE IF name > root.name THEN root.right = deleteContact(root.right, name)
ELSE IF root.left is NULL THEN temp = root.right free(root) RETURN temp ELSE IF root.right is NULL
THEN temp = root.left free(root) RETURN temp ELSE temp = findMinValueNode(root.right) root.name =
temp.name root.phone = temp.phone root.right = deleteContact(root.right, temp.name) ENDIF ENDIF RETURN
root END FUNCTION FUNCTION findMinValueNode(root): current = root WHILE current.left != NULL DO
current = current.left ENDWHILE RETURN current END FUNCTION END
```