Sort Contacts Module

FUNCTION sort(head):

IF head IS NULL OR head.next IS NULL THEN

RETURN head

END IF

middle = getMiddle(head)

nextOfMiddle = middle.next

middle.next = NULL // Split the list into two halves

left = sort(head) // Sort the left half

right = sort(nextOfMiddle) // Sort the right half

RETURN sortedMerge(left, right) // Merge the sorted halves

END FUNCTION

FUNCTION sortedMerge(left, right):

IF left IS NULL THEN

RETURN right

END IF

IF right IS NULL THEN

RETURN left

END IF

IF left.contact.firstName <= right.contact.firstName THEN

result = left result.next = sortedMerge(left.next, right)

ELSE

result = right result.next = sortedMerge(left, right.next)

END IF

RETURN result

END FUNCTION

FUNCTION getMiddle(head):

```
IF head IS NULL THEN

        RETURN NULL

END IF slow = head

    fast = head.next

    WHILE fast IS NOT NULL AND fast.next IS NOT NULL DO

        slow = slow.next

        fast = fast.next.next

    END WHILE

    RETURN slow // Slow is now at the middle node

END FUNCTION
```