Question 3 [12 marks]:

Use the divide-and-conquer strategy to design a recursive algorithm for finding the smallest value in a nonempty sequence of n numbers and specify it in a recursive structure.

Python

```
def find_smallest(arr, start, end):
    # Base case: if there's only one element
    if start == end:
        return arr[start]

    # Find the middle point of the sequence ( Divide )
    mid = (start + end) // 2

    # Recursively find the smallest value in the left half ( Conquer )
    left_min = find_smallest(arr, start, mid)

    # Recursively find the smallest value in the right half ( Conquer )
    right_min = find_smallest(arr, mid + 1, end)

    # Return the smaller value between left_min and right_min ( Combine )
    return min(left_min, right_min)

# Example usage:
arr = [3, 1, 4, 1, 5, 9, 2, 6, 5, 3, 5]
print(find_smallest(arr, 0, len(arr) - 1))
```