Q2.

1. Partition Function: Recursively split the list into two nearly equal halves by alternating elements between two lists.
2. Sorting Each Half: Recursively apply mergesort to both halves to ensure they are sorted before merging.
3. Merge Function: Merge two sorted lists by picking the smaller head element recursively. If one list is empty, return the other.
4. Mergesort Function:

- Base case: If the list has 0 or 1 element, return it.
- Otherwise, partition the list, recursively sort both halves, and merge them.

Q4.

1. Initial and Final State of Lists:

Start: Unsorted list has all elements; sorted list is empty.

End: All elements are in the sorted list, and the unsorted list is empty.

Termination Condition: When the unsorted list is empty.

2. Sorting Process:

Take the first element from the unsorted list.

Insert it into the sorted list in order.

Repeat until all elements are sorted.

3. Insertion Mechanism:

a. If the sorted list is empty or the element is smaller than the first item, insert at the start.

b. Otherwise, find the correct position recursively.

Termination Conditions:

- The unsorted list is empty (sorting is complete).
- The element is correctly inserted into the sorted list.