**Merge Sort Algorithm**

**Original Unsorted Array:**  
[13] [12] [22] [16] [54] [32] [60] [92] [88]

Call 1: Mergesort (0, 8, list);

Call 2a to sort indices 0-4:

[13] [12] [22] [16] [54]

Call 2b to sort indices 5-8:

[32] [60] [92] [88]

Call 2a: Mergesort (0, 4, list);

[13] [12] [22] [16] [54]

Call 3a to sort indices 0-1:

[12] [13]

Call 3b to sort indices 2-3:

[16] [22]

When 3a and 3b return, merge indices (0-1), (2-3), and (4)

[12] [13] [16] [22] [54]

Call 2b: Mergesort (5, 8, list);

[32] [60] [92] [88]

Call 3c to sort indices 5-6:

[32] [60]

Call 3d to sort indices 7-8:

[88] [92]

When 3c and 3d return, merge indices (5-6) and (7-8)

[32] [60] [88] [92]

When 2a and 2b return, merge indices (0-4) and (5-8)

[12] [13] [16] [22] [54] **|** [32] [60] [88] [92]

[12] [13] [16] [22] [54] [32] [60] [88] [92]

Call 3a: Mergesort (0, 1, list);

[12] [13]

Call 3b: Mergesort (2, 3, list);

[16] [22]

Call 3c: Mergesort (4, 5, list);

[32] [54]

Call 3d: Mergesort (6, 7, list);

[60] [88]

Call 4a: Mergesort (0, 0, list);

[12]

Return to call 3a.

Call 4b: Mergesort (1, 1, list);

[13]

Return to call 3b.

Call 4c: Mergesort (2, 2, list);

[16]

Return to call 3b.

Call 4d: Mergesort (3, 3, list);

[22]

Return to call 3c.

Call 4e: Mergesort (4, 4, list);

[32]

Return to call 3c.

Call 4f: Mergesort (5, 5, list);

[54]

Return to call 3d.

Call 4g: Mergesort (6, 6, list);

[60]

Return to call 3d.

Call 4h: Mergesort (7, 7, list);

[88]

All recursive calls have now returned.

Sorted List:

[12] [13] [16] [22] [32] [54] [60] [88] [92]