Fast Sorting Worksheet

1. What is the worst-case time complexity of Insertion Sort?
   1. O(n log n)
   2. O(n)
   3. O(n²)
   4. O(log n)
2. What is the best-case time complexity of Insertion Sort?
   1. O(n log n)
   2. O(n)
   3. O(n²)
   4. O(log n)
3. Which sorting algorithm requires a temporary array for merging?
   1. QuickSort
   2. InsertionSort
   3. MergeSort
   4. CountingSort
4. What is the space complexity of MergeSort?
   1. O(1)
   2. O(n)
   3. O(log n)
   4. O(n log n)
5. In QuickSort, what is a pivot?
   1. The first element of the array
   2. The middle element of the array
   3. A value used to rearrange the array
   4. The smallest element in the array
6. What is the average time complexity of QuickSort?
   1. O(n)
   2. O(n²)
   3. O(n log n)
   4. O(log n)
7. What is the space complexity of QuickSort?
   1. O(n)
   2. O(1)
   3. O(log n)
   4. O(n log n)
8. Which statement is true about QuickSort?
   1. It is always stable
   2. Its performance is independent of pivot selection
   3. It may switch to InsertionSort for small arrays
   4. It requires additional space proportional to input size
9. What is the time complexity of CountingSort?
   1. O(n log n)
   2. O(n + k)
   3. O(n²)
   4. O(k)
10. What is the space complexity of CountingSort?
    1. O(n)
    2. O(k)
    3. O(n + k)
    4. O(1)
11. Which sorting algorithm is stable?
    1. QuickSort
    2. Both MergeSort and CountingSort
    3. Neither MergeSort nor CountingSort
    4. Only MergeSort
12. What happens in each level of MergeSort’s recursion tree?
    1. Only half of the array appears
    2. The entire array appears
    3. Only the sorted elements appear
    4. Only unsorted elements appear
13. Which sorting algorithm is traditionally considered the fastest classical sorting algorithm?
    1. MergeSort
    2. InsertionSort
    3. CountingSort
    4. QuickSort
14. When is CountingSort most efficient?
    1. With large ranges of values
    2. With small positive values
    3. With negative numbers
    4. With floating-point numbers
15. What is a disadvantage of CountingSort?
    1. It’s unstable
    2. It’s slow for small ranges
    3. It uses a lot of space if range is large
    4. It has poor best-case performance
16. What is true about MergeSort’s time complexity?
    1. It varies between best and worst case
    2. It’s O(n) in best case
    3. It’s O(n²) in worst case
    4. It’s O(n log n) in all cases
17. Which sorting algorithm has O(n) as its best-case time complexity?
    1. MergeSort
    2. QuickSort
    3. Both InsertionSort and MergeSort
    4. InsertionSort
18. In InsertionSort’s implementation, what happens to data[i]?
    1. It’s compared with the last element
    2. It’s swapped with the first element
    3. It’s swapped to the left until in sorted order
    4. It’s compared with the middle element
19. What is the relationship between k and n in CountingSort?
    1. k is the size of input array
    2. k is the range of elements
    3. k is always equal to n
    4. k is the number of sorted elements
20. Which sorting algorithm requires the elements to be comparable?
    1. CountingSort
    2. All except CountingSort
    3. None of them
    4. Only QuickSort