BRAINWARE UNIVERSITY
Class Test 1 (2nd Semester) – March, 2025
Program Name – Bachelor of Computer Applications
BCA47111(T) – Design and Analysis of Algorithm

Time - 60 minutes **Full Marks: 20**

(Multiple Choice Type Question)

1. Choose the correct alternative from the following: - **[8 x 1= 8]**

i) Define complexity the recurrence relation T(n) = 8T(n/2) +n2
a) O(n) b) O(n2) c) O(log2 n) d) O(n3)

ii) **Ω** - notation provides an asymptotic
a) Upper bound b) Lower bound c) One that is sandwiched between the two bounds d) None of these

iii) What is the result of the recurrences which fall under Case-1 of Master's theorem? let the recurrence be given by T(n)=aT(n/b)+f(n) and f(n)=nc?
a) T(n) = O(nlog$_b$a) b) T(n) = O(nc log n) c) T(n) = O(f(n)) d) T(n) = O(n2)

iv) What is the worst-case time complexity of binary search?
a) O(n) b) O(log n) c) O(n log n) d) O(n²)

v) Which sorting algorithm has the best worst-case time complexity?
a) Quick Sort b) Merge Sort c) Bubble Sort d) Insertion Sort

vi) Which searching algorithm is best suited for a sorted array?
a) Linear Search b) Binary Search c) Interpolation Search d) Jump Search

vii) In Quick Sort, the worst-case time complexity occurs when:
a) Pivot is always the smallest element b) Pivot is always the largest element c) Both a and b d) None of the above

viii) Which of the following is true about Merge Sort and Quick Sort?
a) Merge Sort is an in-place sorting algorithm, but Quick Sort is not b) Quick Sort is an in-place sorting algorithm, but Merge Sort is not c) Both are in-place sorting algorithms d) Neither is an in-place sorting algorithm

(Short Answer Type Question)

Answer all questions of the following :- **[6 x 2 = 12]**

2. Define Big-O notation with a proper diagram and example.
3. Explain the recurrence tree method with an example.
4. Explain the concept of pivot in Quick Sort.
5. Difference between Binary Search and interpolation search.
6. Solved using the Substitute Method: T(n) = T(n-1) + n
7. Solved using the Master Theorem: T(n) = 2T(n/2) + n log n