## Lecture 5 Algorithm Performance Analysis

1. What is the primary goal of algorithm performance analysis?

a) To measure execution time

b) To count the number of operations

c) To focus on how performance scales

d) All of the above

2. Which of the following is NOT a focus of asymptotic analysis?

a) Runtime as input size gets large

b) Rate of growth determined by the dominating highest-order term

c) Exact values for specific inputs

d) Leading coefficient and lower-order terms falling away

3. In Big-O notation, what does O(1) typically represent?

a) Linear time complexity

b) Constant time complexity

c) Logarithmic time complexity

d) Quadratic time complexity

4. Which of the following is the correct order of time complexities from fastest to slowest?

a) O(n!) < O(2^n) < O(n^2) < O(n log n) < O(n) < O(log n) < O(1)

b) O(1) < O(log n) < O(n) < O(n log n) < O(n^2) < O(2^n) < O(n!)

c) O(n) < O(1) < O(log n) < O(n log n) < O(n^2) < O(2^n) < O(n!)

d) O(1) < O(n) < O(log n) < O(n log n) < O(n^2) < O(2^n) < O(n!)

5. What is the time complexity of selection sort in the worst case?

a) O(n)

b) O(n log n)

c) O(n^2)

d) O(2^n)

6. Which sorting algorithm has a best-case time complexity of O(n)?

a) Selection Sort

b) Merge Sort

c) Insertion Sort

d) Quick Sort

7. What is the time complexity of binary search?

a) O(1)

b) O(log n)

c) O(n)

d) O(n^2)

8. Which case of algorithm analysis represents the worst possible performance for any input of a fixed size n?

a) Best case

b) Average case

c) Worst case

d) Amortized case

9. What is the primary advantage of using Big-O notation for algorithm analysis?

a) It provides exact runtime measurements

b) It focuses on how performance scales with input size

c) It accounts for all possible inputs

d) It measures the best-case scenario

10. In asymptotic analysis, what do we do with constant factors?

a) Keep them as they are

b) Multiply them

c) Drop them

d) Add them to the input size

11. What is the time complexity of merge sort in all cases?

a) O(n)

b) O(n log n)

c) O(n^2)

d) O(2^n)

12. Which of the following is NOT a factor that influences the running time of a program on real systems?

a) Java Compiler

b) Java Virtual Machine

c) Operating System

d) Programming Language Syntax

13. What is the purpose of benchmarking in performance analysis?

a) To prove the correctness of an algorithm

b) To measure actual execution time on real machines

c) To determine the space complexity

d) To calculate the Big-O notation

14. Which Java method is used to get the current time in nanoseconds for benchmarking?

a) System.currentTimeMillis()

b) System.nanoTime()

c) System.getTime()

d) System.benchmarkTime()

15. What is the time complexity of linear search in the worst case?

a) O(1)

b) O(log n)

c) O(n)

d) O(n^2)

16. Which sorting algorithm divides the list in half recursively?

a) Bubble Sort

b) Insertion Sort

c) Selection Sort

d) Merge Sort

17. What is the best-case time complexity of quick sort?

a) O(n)

b) O(n log n)

c) O(n^2)

d) O(2^n)

18. In Big-O notation, what does O(n^2) typically represent?

a) Linear time complexity

b) Logarithmic time complexity

c) Quadratic time complexity

d) Exponential time complexity

19. What is the primary limitation of using real-time measurements for algorithm analysis?

a) It's too accurate

b) It doesn't account for different input sizes

c) It can be inconsistent due to various system factors

d) It's too slow to perform

20. Which of the following is true about the average-case time complexity?

a) It's always equal to the worst-case complexity

b) It considers the performance of an algorithm on all possible inputs of size n

c) It's always O(1)

d) It's irrelevant in algorithm analysis