# Algorithms and Time Complexity

## 1. Explanation: Traveling Salesman Problem (TSP)

The Traveling Salesman Problem (TSP) is a classic optimization problem. The goal is to find the shortest possible route that visits a set of cities exactly once and returns to the starting point.

Problem complexity:

- There is no known efficient algorithm that solves TSP for all cases.  
- In a brute-force solution, all city permutations are checked, which gives factorial complexity: O(n!).  
- The problem is classified as NP-hard.

## 2. Time Complexity Analysis for Tasks 1–4

### Task 1

IF n == 1 THEN return  
FOR i = 1 to n  
 FOR j = 1 to n  
 print "\*"  
 BREAK

➡ Time Complexity: O(n)

### Task 2

FOR i = n/2 to n  
 FOR j = 2 to n, j \*= 2  
 a = a + n / 2

➡ Time Complexity: O(n log n)

### Task 3

FOR i = 0 to n  
 FOR j = i down to 0  
 a = a + i + j

➡ Time Complexity: O(n²)

### Task 4

WHILE i > 0  
 a = a + i  
 i = i / 2

➡ Time Complexity: O(log n)

## 3. Time Complexity Table

|  |  |
| --- | --- |
| Task | Time Complexity |
| Task 1 | O(n) |
| Task 2 | O(n log n) |
| Task 3 | O(n²) |
| Task 4 | O(log n) |