Name: ________________ **Date:** ________________

Answer Key: Wrangle Algorithmic Complexity: Your Senior-Level CS Challenge

Synthesize advanced data structures and evaluate amortized cost analysis during this rigorous review of non-linear problem-solving strategies.

---

1. When designing a memory-efficient system for an autocomplete feature on a mobile device, which data structure provides the best balance of prefix search speed and space optimization compared to a standard Hash Map?

Answer: B) Compressed Trie (Radix Tree)

A Compressed Trie or Radix Tree reduces space complexity by merging nodes with a single child while maintaining O(k) search time where k is the string length, making it superior to Hash Maps for prefix matching.

2. In the context of the Master Theorem, if the work done at the root of a recursive sub-problem grows significantly faster than the work done by the leaves, the overall complexity is dominated by the cost of the initial split/combine step.

Answer: A) True

This describes Case 3 of the Master Theorem, where the work at the top level (f(n)) grows faster than the number of leaves (n^log_b(a)), leading to a complexity of Theta(f(n)).

3. During the synthesis of a network routing protocol, you decide to use Dijkstra's algorithm. To achieve an optimal time complexity of O(E + V log V), you must implement the priority queue using a ______.

Answer: B) Fibonacci Heap

A Fibonacci Heap allows for O(1) amortized decrease-key operations, which reduces the Dijkstra complexity from O(E log V) to O(E + V log V).

4. Which algorithmic design paradigm is most appropriate for a problem that exhibits both 'optimal substructure' and 'overlapping subproblems' but does not necessarily satisfy the 'greedy choice property'?

Answer: C) Dynamic Programming

Dynamic Programming is specifically designed for problems where the same subproblems are solved repeatedly (overlapping) and can be combined to form an optimal solution (optimal substructure).

Name: ________________ Date: ________________

5. Consider the Ford-Fulkerson method for finding maximum flow. The efficiency of the implementation depends heavily on the choice of the path-finding strategy. Using Breadth-First Search (BFS) transforms it into the ______ algorithm.

Answer: B) Edmonds-Karp

The Edmonds-Karp algorithm is a specific implementation of Ford-Fulkerson that uses BFS to find augmenting paths, ensuring a polynomial time complexity of O(VE^2).

6. Adding a heuristic to a Uniform Cost Search consistently guarantees that the search will explore fewer nodes, regardless of whether the heuristic is admissible or consistent.

Answer: B) False

A poor heuristic can actually lead the search astray or require more computations; only an admissible and consistent heuristic guarantees both efficiency and optimality in A* search.

7. In a distributed system where you need to check if a specific element exists across massive datasets with limited memory and can tolerate a small percentage of false positives, which would you implement?

Answer: C) Bloom Filter

Bloom Filters are probabilistic data structures that are extremely space-efficient for set membership tests, allowing for false positives but never false negatives.

8. When solving the 'All-Pairs Shortest Path' problem on a graph that may contain negative edge weights (provided there are no negative cycles), the most robust algorithm to apply is ______.

Answer: B) Floyd-Warshall

Floyd-Warshall handles negative weights and computes paths between all pairs of vertices in O(V^3) time using dynamic programming.

9. NP-Complete problems are a subset of NP-Hard problems that can be solved in polynomial time if any other NP-Complete problem is also solved in polynomial time.

Answer: A) True

The definition of NP-Completeness requires the problem to be in NP and for all problems in NP to be reducible to it in polynomial time; thus, solving one is equivalent to solving all.

Name: ________________ Date: ________________

10. You are tasked with optimizing a recursive algorithm that evaluates game states in Chess. To prune the search tree without affecting the final result, which technique is most effective?

Answer: B) Alpha-Beta Pruning

Alpha-Beta pruning reduces the number of nodes evaluated by the minimax algorithm in its search tree by eliminating branches that cannot possibly influence the final decision.