Name: _____ **Date:** _____

Answer Key: Glitch in the Matrix: 7th Grade Algorithmic Logic Quest

Evaluate multi-step heuristic efficiency and troubleshoot recursive simulations through high-level computational thinking exercises.

---

1. You are designing a routing algorithm for a global delivery drone. To ensure the drone finds the fastest path through a shifting wind storm while avoiding skyscrapers, which advanced strategy is most vital?

Answer: B) Heuristic-based search to prioritize paths closer to the destination

In complex, large-scale problems, a heuristic-based search (like A*) is more efficient than linear searching because it uses an 'informed' guess to navigate toward the goal faster.

2. When a developer analyzes how the execution time of an algorithm grows as the input size increases toward infinity, they are measuring _____.

Answer: B) Computational complexity

Computational complexity (specifically time complexity) describes the amount of time an algorithm takes to run as a function of the length of the input.

3. A 'Greedy Algorithm' always produces the globally optimal solution for any complex problem because it makes the best choice at each small step.

Answer: B) False

False. While greedy algorithms are fast, they often find 'local' optima and may miss the best overall (global) solution by failing to look ahead.

4. You are creating an algorithm for a Smart Home system to manage energy consumption. The system should prioritize high-power appliances only when solar production is above 80%. Which logic structure is being used?

Answer: B) Conditional Branching

Conditional branching (If/Then) allows the algorithm to make decisions based on specific environmental data or variables.

5. In a simulation of a colony of ants, each individual ant follows a set of simple rules to perform complex tasks. This is an example of _____ behavior in algorithm design.

Name: _____ **Date:** _____

Answer: A) Emergent

Emergent behavior occurs when simple local rules followed by individuals lead to complex, organized global patterns.

6. Abstraction is the process of hiding the complex background details of an algorithm and focusing only on the high-level logic needed to solve the problem.

Answer: A) True

True. Abstraction allows computer scientists to manage complexity by ignoring unnecessary details.

7. To optimize a search engine for a library containing 10 billion research papers, which data structure would provide the fastest access to a specific keyword?

Answer: C) An inverted index (map)

An inverted index maps keywords to their locations, allowing for near-instant retrieval (O(1) or O(log n)) compared to scanning the whole set.

8. An algorithm that calls itself within its own definition to solve smaller versions of the same problem is known as a _____ function.

Answer: B) Recursive

Recursion is a common technique used in algorithms like QuickSort or tree traversal to break problems down into identical sub-problems.

9. Standard algorithmic testing only requires checking if the code works with 'perfect' data that follows all instructions.

Answer: B) False

False. Robust testing must include 'edge cases' or 'stress tests'—unusual or extreme inputs—to ensure the algorithm fails gracefully or handles errors.

10. In 'Parallel Processing,' how is an algorithm's execution changed to improve efficiency over a large dataset?

Answer: C) It splits the problem into sub-tasks performed simultaneously on multiple processors.

Parallelism increases efficiency by dividing the workload across multiple processing cores at the same time.