Advanced Data Structures and Algorithms

Comprehensive Assignment Solutions

Arya Raag
Roll No: A125002
M.Tech (Computer Science and Engineering)

January 5, 2026

1 Complexity Classes

Question 2. Given a Boolean circuit instance whose output evaluates to true, explain how the correctness of the result can be verified in polynomial time using Depth First Search (DFS).

Detailed Solution:

Problem Context

The **Circuit Value Problem** asks given a boolean circuit (a Directed Acyclic Graph of gates) and inputs, what is the output? We are discussing the *verification* aspect. However, since the problem is in **P**, verification is essentially just re-running the computation.

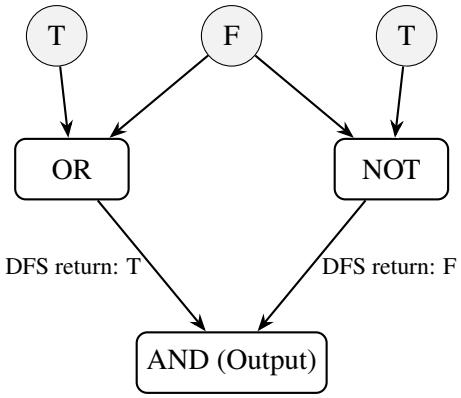
Verification Algorithm using DFS

We can verify the circuit's output by re-evaluating the circuit. Since a circuit is a DAG:

1. **Graph Representation:** Treat gates as nodes and wires as directed edges.
2. **Recursive DFS:** We define a function `Evaluate(gate)`:
   - If `gate` is an Input, return its assigned value.
   - If `gate` has been visited/memoized, return the stored value.
   - Recursively call `Evaluate` on all inputs to this gate.
   - Apply the logic operation (AND, OR, NOT) to the results.
   - Store and return the result.

Visualizing Circuit Evaluation via DFS

The diagram below shows a simple boolean circuit. A DFS traversal would start at the output node and recursively request values from inputs (leaf nodes), computing the logic at each step.



Final Result: False

Complexity Analysis

- **Vertices (V):** Each gate is a vertex.
- **Edges (E):** Each wire is an edge.

Using DFS with memoization (to handle fan-out, where one gate's output feeds multiple inputs), we visit each node and traverse each edge exactly once. The logic computation at each node takes $O(1)$ time. Total Time Complexity: $O(V+E)$.

Conclusion

Since the size of the circuit description is proportional to $V+E$, the verification runs in linear time relative to the input size. This is strictly polynomial time.
