Stock Trading Engine Technical Report

1. Executive Summary
This document details the implementation of a high-performance, real-time stock trading engine designed to match buy and sell orders efficiently while maintaining thread safety without using standard library collections.

2. System Architecture

2.1 Core Components
- TradingSimulation: Central coordinator
- StockExchange: Order routing and management
- OrderBook: Order matching engine
- Order: Data structure for orders

2.2 Technical Specifications
- Support for 1,024 concurrent stock tickers
- Lock-free concurrent operations
- O(n) time complexity for order matching
- Custom data structure implementation

3. Data Structures Implementation

3.1 Primary Data Structures

Arrays
Usage: Ticker and OrderBook Storage
- Implementation: Fixed-size array (1024 elements)
- Access Time: O(1) with index
- Memory Footprint: Constant
- Thread Safety: Read-only after initialization

Lock-Free Linked Lists
Usage: Order Management
- Implementation: Custom linked list with atomic references
- Operations:
  - Insertion: O(n) maintaining sort order
  - Deletion: O(1) with node reference
  - Traversal: O(n) for matching
- Thread Safety: Atomic operations

4. Algorithm Implementation

4.1 Order Matching Algorithm

1. Check buy and sell order heads
2. Compare prices
3. If match found:
   - Calculate trade quantity
   - Update order quantities
   - Remove completed orders
   - Log trade
4. Repeat until no matches possible

4.2 Time Complexity Analysis
- Order Insertion: O(n)
- Order Matching: O(n)
- Ticker Lookup: O(n)
- Overall Performance: Linear scaling

5. Thread Safety Mechanisms

Atomic Operations
- Compare-and-Set (CAS) for order updates
- Volatile references for visibility
- Lock-free order book modifications

6. Testing and Validation

1. Simple match
2. Partial match
3. No match
4. Multiple matches
5. Cross-ticker validation

7. Conclusion
The implementation successfully meets all requirements while maintaining thread safety and performance goals.