Project 2

Group members:
Drashti Mehta
Kunal Doshi

Introduction:
The Stock Maximization Problem involves determining the maximum number of stocks an investor can purchase given a limited amount of available financial resources. Two approaches, A and B, have been proposed to address this problem.

Approach A - Exhaustive Search:
This method employs an exhaustive search algorithm to evaluate all possible subsets of stocks and select the one with the highest total value within the budget constraint. The time complexity of this approach is exponential, specifically O(2^N), where N is the number of stocks. This is due to the need to consider all possible combinations.

Approach B - Dynamic Programming:
The dynamic programming approach employs a top-down strategy to handle overlapping subproblems, storing results in a two-dimensional array. The time complexity is more efficient, specifically O(N * Amount), where N is the number of stocks and Amount is the available investment sum. This approach optimally computes the maximum value obtainable within the given constraints.

Comparison and Recommendation:
While Approach A is straightforward, its exponential time complexity makes it less efficient, especially for larger input sizes. Approach B, with its polynomial time complexity, is more scalable and suitable for real-world scenarios with a considerable number of stocks. The dynamic programming approach, by avoiding redundant calculations, significantly outperforms the exhaustive search.

Conclusion:
In conclusion, Approach B, utilizing dynamic programming, is recommended for solving the Stock Maximization Problem. Its superior time complexity ensures efficient computations, making it more practical for handling larger datasets and real-world investment scenarios.

Scalability:
Approach A: Inefficient for larger datasets due to exponential growth.
Approach B: More efficient and scalable, especially for realistic scenarios with numerous stocks.

Code Logic:
Approach A: Directly explores all combinations, leading to redundant calculations.
Approach B: Utilizes memoization to avoid recomputing results, optimizing the overall process.