SUPPLY CHAIN MANAGEMENT SYSTEM

# A PROJECT REPORT

# *Submitted by*

**ADITII SHARMA [Reg No: RA2211027010093]**

**PARTH BHUNIA[Reg No: RA2211027010096]**

**ARNAV GUPTA[Reg No: RA2211027010125]**

*Under the Guidance of*

**DR. Rajkumar K**

Associate Professor, Department of Data Science and Business Systems *In partial fulfilment of the requirements for the degree of* **BACHELOR OF TECHNOLOGY**

**in**

**COMPUTER SCIENCE AND ENGINEERING**

**with a specialization in BIG DATA ANALYTICS**



**DEPARTMENT OF DATA SCIENCE AND BUSINESS SYSTEMS**

**COLLEGE OF ENGINEERING AND TECHNOLOGY SRM INSTITUTE OF SCIENCE AND TECHNOLOGY KATTANKULATHUR – 603 203**

**NOVEMBER 2023**

**SRM INSTITUTE OF SCIENCE AND TECHNOLOGY KATTANKULATHUR – 603 203**

**BONAFIDE CERTIFICATE**

Certified that this B.Tech project report titled “**Supply Chain Management System**” is the bonafide work of **Aditii Sharma[RA2211027010093]**, **Parth Bhunia[RA2211027010096]** and **Arnav Gupta[RA2211027010125]** who carried out the project work under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion for this or any other candidate.

**DR. RAJUKUMAR K**

Associate Professor

Department of Data Science and Business Systems

SRM Institute of Science and Technology

Kattankulathur

**DR. M. LAKSHMI**

**HEAD OF THE DEPARTMENT**

Department of Data Science and Business Systems

SRM Institute of Science and Technolgy

Kattankulathur

**ACKNOWLEDGEMENT**

We express our humble gratitude to Dr. C. Muthamizhchelvan, Vice-Chancellor, SRM Institute of Science and Technology, for the facilities extended for the project work and his continued support.

We extend our sincere thanks to Dean-CET, SRM Institute of Science and Technology,

Dr. T.V. Gopal, for his invaluable support.

We wish to thank Dr. Revathi Venkataraman, Professor & Chairperson, School of Computing, SRM Institute of Science and Technology, for her support throughout the project work.

We are incredibly grateful to our Head of the Department, Dr. M. Lakshmi Professor, Department of Data Science and Business Systems, SRM Institute of Science and Technology, for her suggestions and encouragement at all the stages of the project work.

We register our immeasurable thanks to our Faculty Advisor, Dr. Jayaraj R, Assistant Professor, Department of Data Science and Business Systems, SRM Institute of Science and Technology, for leading and helping us to complete our course.

Our inexpressible respect and thanks to my guide, DR. Rajkumar K, Assistant Professor, Department of Data Science and Business Systems, for providing me with an opportunity to pursue my project under his mentorship. He provided us with the freedom and support to explore the research topics of our interest. His passion for solving problems and making a difference in the world has always been inspiring.

We sincerely thank the Data Science and Business Systems staff and students, SRM Institute of Science and Technology, for their help during our project. Finally, we would like to thank parents, family members, and friends for their unconditional love, constant support, and encouragement.

Aditii Sharma, Parth Bhunia, Arnav Gupta

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| 1. | INTRODUCTION | 4 |
| 2 | ALGORITHMS USED AND TIME COMPLEXITY | 6 |
| 3 | COMPARATIVE STUDY | 8 |
| 4 | RESULT | 10 |
| 5 | IMPLEMENTATION/OUTPUT | 11 |
| 6 | REFERENCE | 13 |

**INTRODUCTION**

Supply chain management (SCM) plays a crucial role in the efficient functioning of any organization, and its significance in the healthcare sector, particularly in hospitals, cannot be overstated. In the context of hospitals, SCM refers to the oversight and coordination of various processes involved in the acquisition, storage, and distribution of medical supplies, pharmaceuticals, equipment, and other essential resources. A well-designed SCM system ensures that hospitals have timely access to the necessary materials, thereby enabling them to deliver high-quality patient care while optimizing operational efficiency and controlling costs. With the complexity of healthcare operations and the criticality of medical supplies, an effective SCM system in hospitals involves careful planning, inventory management, procurement strategies, supplier relationships, logistics coordination, and technology integration. By leveraging SCM principles, hospitals can streamline their supply chains, minimize stockouts and wastage, enhance resource utilization, maintain regulatory compliance, and ultimately, improve patient outcomes. In today's dynamic healthcare landscape marked by evolving patient needs, cost pressures, and supply chain disruptions, the adoption of advanced SCM practices becomes imperative for hospitals to sustainably meet the demands of modern healthcare delivery.

This project implements a supply chain management system for a hospital, focusing on forecasting demand, optimizing costs, and planning distribution routes for medical supplies. It allows users to input orders and supplier information, perform demand forecasting, optimize costs based on supplier pricing, and plan distribution routes within a specified budget and current stock level.

In the realm of hospital supply chain management (SCM), the principles of Design and Analysis of Algorithms (DAA) play a fundamental role in optimizing various processes and ensuring efficient resource utilization. DAA, a core field of computer science, provides methodologies for designing and evaluating algorithms to solve complex problems. When applied to hospital SCM systems, DAA methodologies contribute to enhancing the efficiency, reliability, and responsiveness of supply chain operations.

**ALGORITHMS TECHNIQUES USED**

**AND THEIR TIME COMPLEXITY**

**Greedy Algorithm**

The algorithm design techniques used in this code include: Greedy Algorithm: The ‘route\_planning’ method employs a greedy approach to select distribution routes. It iterates through sorted orders based on priority and selects orders that can be fulfilled within the maximum budget and current stock level, minimizing costs.

**Algorithm**

The algorithm follows these steps:

1) Sort orders based on priority in descending order.

2)Initialize an empty list for distribution routes.

3)Iterate through sorted orders.

4)If the cost of adding the order to the distribution route does not exceed the maximum budget and the order's quantity can be fulfilled with the current stock level, add the order to the distribution routes and update the current stock level.

5)Return the selected distribution routes.

**Time Complexity Analysis**

• Best Case**: O(n log n)** - In the best case scenario, the time complexity remains O(n log n) because the sorting of orders based on priority using the sorted function has a time complexity of O(n log n).

• Average Case: **O(n log n)** - Similarly, in the average case, the time complexity is O(n log n) due to the sorting operation.

• Worst Case: **O(n log n)** - Even in the worst case scenario, the time complexity

remains O(n log n) as the sorting operation dominates the overall time complexity.

The reason why the time complexity is O(n log n) in all cases is because the

algorithm primarily consists of sorting the orders based on priority. Sorting

algorithms like Python's built-in sorted function typically have a time complexity of O(n log n) for average and worst case scenarios. Therefore, regardless of the input data or specific circumstances, the time complexity for this operation remains consistent.

**COMPARATIVE STUDY**

To provide a comparative study with other algorithms, let's consider how the same supply chain management code would differ if implemented with different algorithmic approaches such as Dynamic Programming and Branch and Bound.

1. Dynamic Programming:

* Dynamic programming would involve breaking down the problem of selecting distribution routes into smaller subproblems and solving them recursively. However, implementing dynamic programming for this problem would be challenging as it requires defining subproblems, identifying overlapping subproblems, and storing intermediate results to avoid redundant computations.

Time Complexity-

The time complexity of dynamic programming depends on the specific implementation and problem structure but can range from O(n^2) to O(2^n) depending on the number of subproblems and the efficiency of memoization or tabulation.

1. Branch and Bound:

* Branch and bound is a technique used to solve optimization problems by systematically searching the solution space, pruning branches that cannot lead to better solutions.
* Implementing branch and bound for this problem would involve exploring different combinations of distribution routes and selecting the optimal combination that satisfies constraints while minimizing costs.

Time Complexity-

The time complexity of branch and bound can vary depending on the problem instance and the branching factor. It can range from exponential (O(b^d)) in the worst case to polynomial (O(n^k)) in certain cases where pruning is effective.

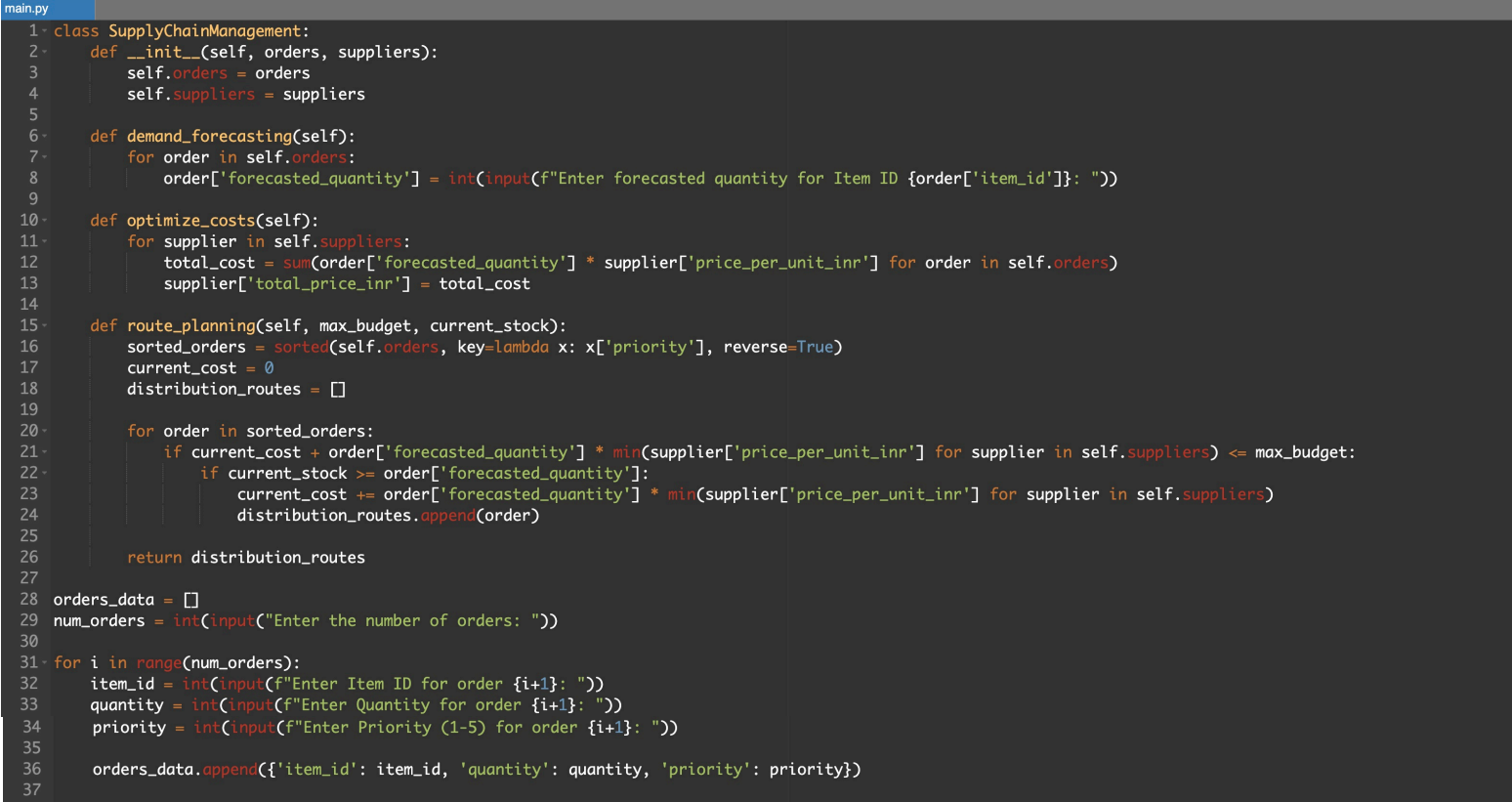
**RESULT**

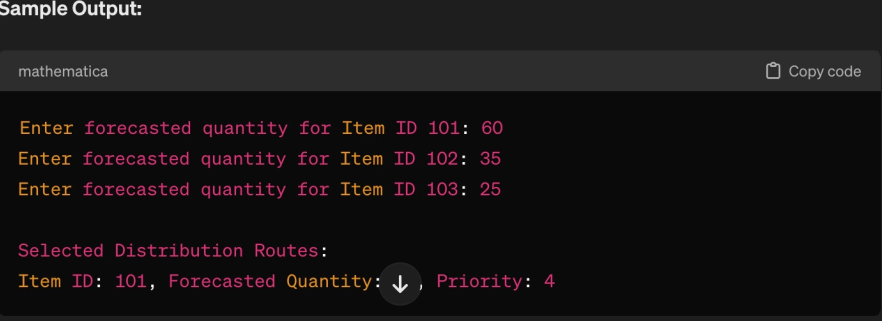
The Greedy Algorithm approach used in the provided code is efficient for small to medium-sized instances of the hospital supply chain management problem.

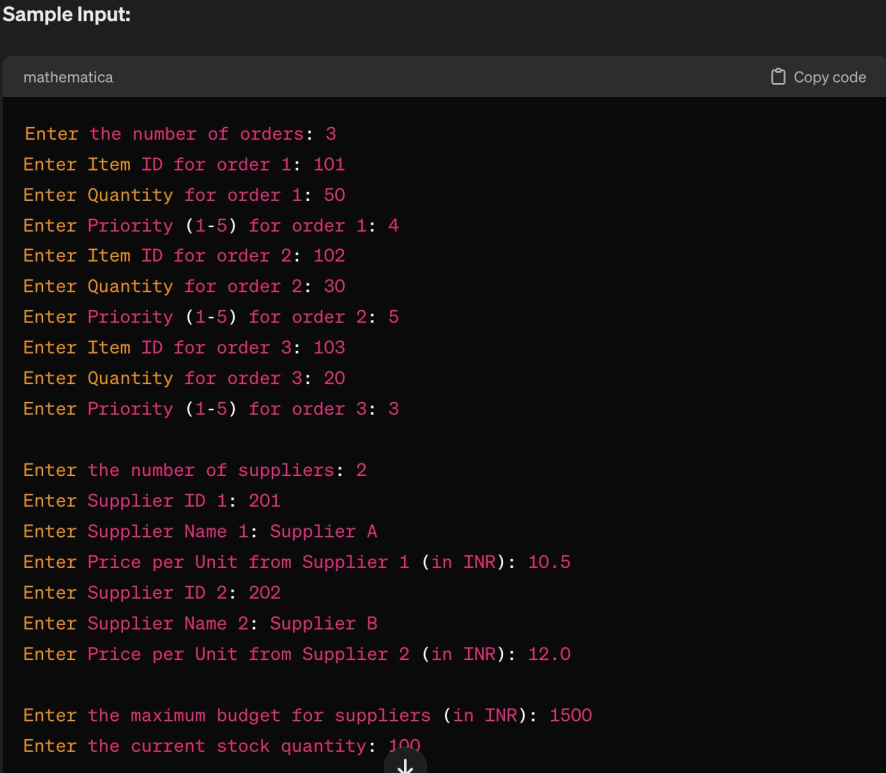
Dynamic Programming and Branch and Bound offer alternative approaches that can potentially provide optimal solutions but may be more complex to implement and have higher computational complexity. Depending on the specific requirements and constraints of the hospital supply chain management problem, the choice of algorithm may vary. Greedy Algorithm is suitable for quick, approximate solutions, while Dynamic Programming and Branch and Bound may be necessary for achieving optimal solutions at the cost of increased computational complexity.

**IMPLEMENTATION**





**OUTPUT**



**REFERENCES**

1. \*\*"Operating System Concepts" by Abraham Silberschatz, Peter B. Galvin, Greg Gagne\*\*

2. \*\*"Modern Operating Systems" by Andrew S. Tanenbaum\*\*

3. \*\*OSDev.org (Operating System Development) \*\*

- Website: [OSDev.org] (<https://wiki.osdev.org/Main_Page>)

4. \*\*Linux Kernel Archives\*\*

- Website: [Linux Kernel Archives] (https://www.kernel.org/)