CASH FLOW MINIMIZER

MINOR PROJECT REPORT

By

JANVI AGRAWAL (RA2211003010283)
LIMANSHA SINGH (RA2211003010280)

Under the guidance of

Dr.Suresh Anand

Assistant Professor, Department of Computing Technologies

In partial fulfilment for the Course

of

21CSC201J – DATA STRUCTURES AND ALGORITHMS

in COMPUTING TECHNOLOGIES

FACULTY OF ENGINEERING AND TECHNOLOGY

SCHOOL OF COMPUTING

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR

NOVEMBER 2023

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

(Under Section 3 of UGC Act, 1956)

BONAFIDE CERTIFICATE

Certified that this minor project report for the course **21CSC201J DATA STRUCTURES AND ALGORITHMS** titled "**Cash Flow Minimizer**" is the bonafide work of **Janvi Agrawal (RA2211003010283)** and **Limansha Singh (RA2211003010280)** who carried out the work under my supervision.

Certified further, that to the best of my knowledge the work reported herein does not form part of any other work.

SIGNATURE	**SIGNATURE**
Dr.suresh anand **Assistant Professor** **Computing Technologies Department** SRM Institute of Science and Technology Kattankulathur	Dr. M. PUSHPALATHA **Professor and Head** **Computing Technologies Department** SRM Institute of Science and Technology 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**,forhis invaluable support.

Wewishtothank**Dr.RevathiVenkataraman**,Professorand Chairperson,SchoolofComputing, SRM Institute of Science and Technology, for her support throughout the projectwork.

We are incredibly grateful to our Head of the Department, **Dr. M. Pushpalatha,** Professor,DepartmentofComputing Technologies,SRMInstituteofScienceandTechnology,forhersuggestions and encouragement atall the stages of theprojectwork.

We want to convey our thanks to our Project Coordinators,**S. Godfrey Winster,Dr. M. Baskar, Dr. P. Murali, Dr. J. Selvin Paul Peter, Dr. C. Pretty Diana Cyriland Dr.G.Padmapriya,**PanelHead,**Dr.A.Kowsigan**,AssociateProfessorandPanel Members,**Dr. M. Vijayalakshmi**Assistant Professor, Mrs **A. Mariya Nancy**Assistant Professor and **Dr. N. Arunachelam**Assistant Professor, Department of Computing Technologies, SRM Institute ofScienceand Technology,fortheirinputs during theproject reviews andsupport.

We register our immeasurable thanks to our Faculty Advisor, **Dr.Suresh anand**, Assistant Professor, Department of Computing Technologies, SRM Institute of Science and Technology ,for leadingand helping us to complete ourcourse. Our inexpressible respect and thanks to our guide, **Dr.Suresh Anand**, Associate Professor,DepartmentofComputing Technologies,SRMInstituteofScienceandTechnology,for providing us with an opportunity to pursue our project under his / hermentorship. He / She providedus with the freedom and support to explore the research topics of our interest. His /Her passion forsolvingproblems and makingadifferencein the world has always beeninspiring.

We sincerely thank allthe staff and students of Computing TechnologiesDepartment, SchoolofComputing, S.R.MInstituteofScienceandTechnology,fortheirhelpduringourproject.Finally,wewould like to thank our parents, family members, and friends for their unconditional love, constant support and encouragement.

TABLE OF CONTENTS

1. PROBLEM DEFINITION

The Cash Flow Minimizer project is a sophisticated application that leverages the principles of data structures and algorithms to tackle the complex task of optimizing cash flow within a group of individuals. At its core, the project employs a graph data structure to elegantly represent the individuals involved and the debts existing between them. This innovative approach allows for a holistic view of financial relationships within the group. The project's objective is to determine the smallest number of transactions necessary to settle the debts between a group of people.

To efficiently identify key individuals with the highest and lowest debt loads, the project cleverly utilizes heaps—an essential data structure in the realm of algorithmic efficiency. By strategically determining these extremes, the program orchestrates transactions between the parties involved, systematically reducing the overall debt burden. This iterative process continues until all debts are effectively settled. To keep track of transactions and their sums, the project also uses multisets, a data structure that enables storing numerous instances of the same information. The multisets assist in determining the least number of transactions necessary to pay off the debts.

The Cash Flow Minimizer project serves as an example of the power and adaptability of DSA in dealing with practical issues. The project is especially helpful in situations where it is difficult to determine the minimum number of transactions necessary to settle the debts between a group of people. The Cash Flow Minimizer project addresses the optimization of cash flow within a group of individuals who have financial transactions and debts among themselves. The primary objective is to determine the smallest number of transactions required to settle all debts within the group efficiently.

2. PROBLEM EXPLAINATION

A Cash Flow Minimizer Project in Data Structures and Algorithms (DSA) typically refers to a problem that involves managing and optimizing the flow of cash or funds in an effective manner. The objective is to minimize the number of transactions or movement of funds while ensuring that all financial obligations are met. This problem is often framed in the context of a directed graph representing financial transactions between entities.

Imagine a scenario where individuals within a group owe money to each other. This interconnected web of debts forms a complex graph. Each person represents a vertex, and each financial transaction is an edge connecting two individuals. The task is to settle these debts using the least number of transactions.

Example of a scenario of Cash Flow Minimizer with its diagram:

In this example, A owes money to B, B owes money to C, C owes money to D, D owes money to E, and E owes money to A. The goal is to find the optimal set of transactions to settle these debts.

3. DATA STRUCTURES USED

The project employs a **graph data structure** to represent individuals and their financial relationships. Each person is a vertex, and each debt is an edge. Additionally, **heaps** are utilized to efficiently identify individuals with the highest and lowest debt loads, facilitating the decision-making process for transaction generation. The project also leverages **multisets** to manage and track multiple instances of transaction information.

A **graph data structure** is an abstract data type (ADT) that consists of a set of objects that are connected to each other via links. These objects are called vertices and the links are called edges. A graph G can be defined as an ordered set G (V, E) where V(G) represents the set of vertices and E(G) represents the set of edges which are used to connect these vertices. There are 2 types of graphs: Directed and Undirected Graphs.

A **heap** is a complete binary tree, and the binary tree is a tree in which the node can have utmost two children. Before knowing more about the heap data structure, we should know about the complete binary tree. A complete binary tree is a binary tree in which all the levels except the last level, i.e., leaf node should be completely filled, and all the nodes should be left-justified. There are 2 types of heaps: minimum heap and maximum heap.

A **multiset** is an unordered collection of elements, in which the multiplicity of an element may be one or more than one or zero. The multiplicity of an element is the number of times the element repeated in the multiset. In other words, we can say that an element can appear any number of times in a set. Operations on multisets are- Union, Intersection, Difference, Sum, and Cardinality.

4. ALGORITHM FOR THE PROBLEM

- Use heaps to identify individuals with the highest and lowest debt loads.
- Generate transactions between these individuals to settle debts.
- Update the debt information and repeat until all debts are settled.

5. ALGORITHM EXPLANATION

<u>Graph Representation:</u>

The algorithm models financial transactions as a directed graph, where each person is a node, and transactions between individuals are represented as edges.
Edges store information about the direction of the transaction (creditor-debtor relationship) and the transaction amount.

<u>Heap Data Structure:</u>

A min-heap is used to efficiently track individuals with the maximum and minimum balances in the system.
Each node in the heap represents a person's balance. The heap helps identify the individuals who need to settle their balances most urgently.

<u>Multiset Data Structure:</u>

A multiset, implemented using SortedList, is employed to efficiently maintain a sorted list of unique balances across all individuals.
This data structure allows quick retrieval and removal of the minimum balance during the settlement process.

Add Transaction Function:

The add_transaction function adds a new financial transaction to the graph.
It updates the graph with the transaction details, including the amount and direction (creditor to debtor).
Balances for the involved individuals are updated, and the multiset is adjusted to reflect the new balances.

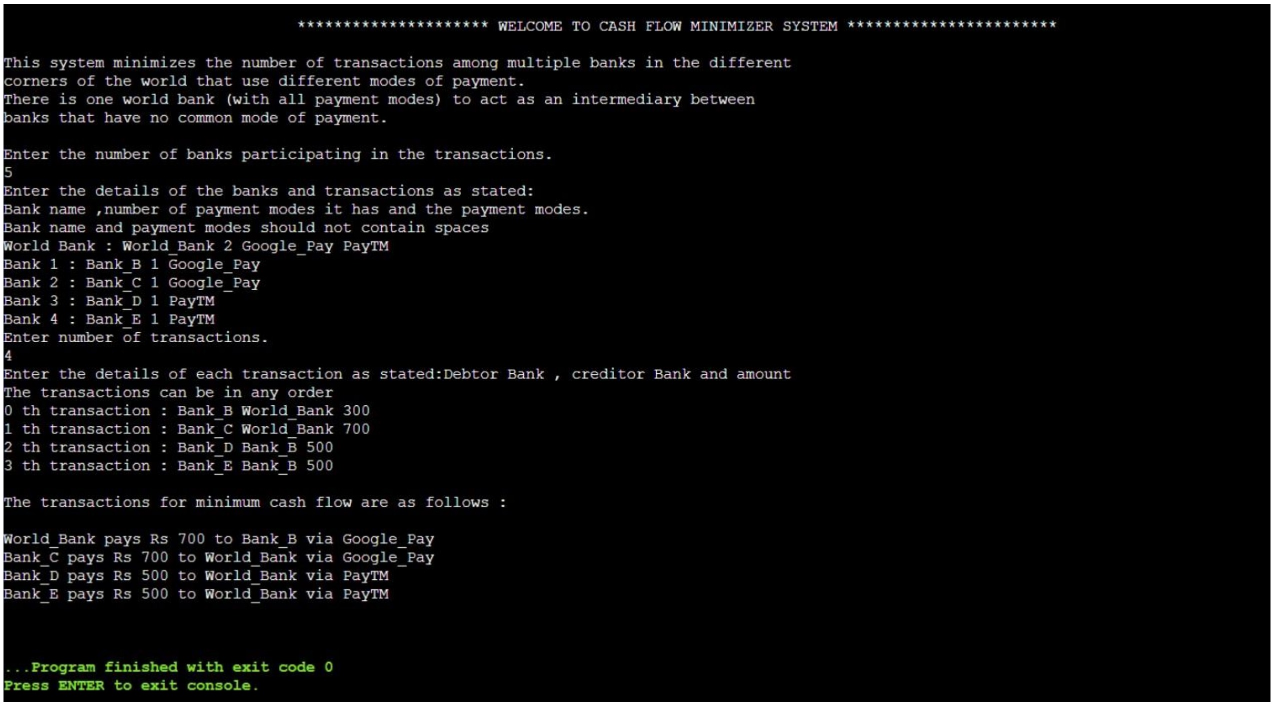
Minimize Cash Flow Function:

The minimize_cash_flow function is the core of the algorithm and aims to settle financial imbalances until all individuals have zero balances.
It initializes the heap with the initial balances of each person.
The algorithm iteratively selects the individuals with the maximum and minimum balances from the heap and multiset.
It calculates the amount needed to settle the imbalance and performs the settlement by updating balances and adjusting the data structures.
The process continues until all balances are minimized, and the heap is empty.

Settlement Process:

The settlement process efficiently identifies pairs of individuals—one with the maximum balance and one with the minimum balance.
It calculates the amount to transfer between these individuals to settle their balances.
The balances and data structures are updated accordingly, ensuring that the settlement process is both effective and optimal.

6. DEMO OUTPUT SCREENSHOTS



```
******************** WELCOME TO CASH FLOW MINIMIZER SYSTEM ***********************

This system minimizes the number of transactions among multiple banks in the different
corners of the world that use different modes of payment.
There is one world bank (with all payment modes) to act as an intermediary between
banks that have no common mode of payment.

Enter the number of banks participating in the transactions.
5
Enter the details of the banks and transactions as stated:
Bank name ,number of payment modes it has and the payment modes.
Bank name and payment modes should not contain spaces
World Bank : World_Bank 2 Google_Pay PayTM
Bank 1 : Bank_B 1 Google_Pay
Bank 2 : Bank_C 1 Google_Pay
Bank 3 : Bank_D 1 PayTM
Bank 4 : Bank_E 1 PayTM
Enter number of transactions.
4
Enter the details of each transaction as stated:Debtor Bank , creditor Bank and amount
The transactions can be in any order
0 th transaction : Bank_B World_Bank 300
1 th transaction : Bank_C World_Bank 700
2 th transaction : Bank_D Bank_B 500
3 th transaction : Bank_E Bank_B 500

The transactions for minimum cash flow are as follows :

World_Bank pays Rs 700 to Bank_B via Google_Pay
Bank_C pays Rs 700 to World_Bank via Google_Pay
Bank_D pays Rs 500 to World_Bank via PayTM
Bank_E pays Rs 500 to World_Bank via PayTM



...Program finished with exit code 0
Press ENTER to exit console.
```

7. COMPLEXITY ANALYSIS

Time Complexity: O(N2) where N is the number of persons.

Space Complexity: O(N)

8. CONCLUSION

The Cash Flow Minimizer project has been a significant achievement in harnessing the power of data structures and algorithms to streamline the intricate process of settling debts within a group.
By employing graph representations, heaps, and multisets, the project successfully navigated the complexities associated with financial relationships, offering an efficient solution to minimize the number of transactions required.
Throughout the development, key features such as the utilization of heaps for identifying optimal transaction candidates and the versatility of multisets in managing transaction data have proven instrumental.
The project not only met its objective of settling debts but also demonstrated the practicality and adaptability of data structures and algorithms in addressing real-world challenges.

9. FUTURE SCOPE

1. **Integration with AI and Machine Learning:** Incorporating machine learning algorithms can help predict future cash flow patterns, making the optimization process more proactive. AI can also assist in decision-making by analyzing historical data and external factors.
2. **Blockchain and Cryptocurrency:** As blockchain technology gains traction, there's potential for cash flow minimization projects to incorporate cryptocurrencies and smart contracts, enabling more secure and efficient transactions.
3. **Real-time Financial Systems:** The demand for real-time financial systems is increasing. Developing algorithms that can optimize cash flow in real-time, adapting to market fluctuations and transaction dynamics, is a significant area of growth.

10. REFERENCES

1. **Textbooks:** "Introduction to Algorithms" by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein.
2. **Open-Source Projects:** We looked for open- source projects or GitHub repositories related to financial optimization. We found sample code, algorithms, or libraries that can be adapted to our project.
3. **Research Papers**: We looked for research papers related to cash flow optimization, linear programming, and algorithmic approaches to financial optimization. Databases like Google Scholar and academic journals were our valuable sources.
4. **Fintech and Startups:** Cash flow optimization is of particular interest to fintech startups and financial institutions. The project can cater to the growing fintech sector and provide solutions for optimizing financial processes.
5. **Scalability and Performance:** As businesses and financial systems grow, the need for scalable and high-performance cash flow optimization algorithms will continue to rise. Developing algorithms that can handle large datasets and complex scenarios efficiently is crucial.