Transaction Processing System

1. Store formatted text file
2. Update an account: **Update the records for account numbers 4 and 6.**
3. Add a new account: **Add new records for account numbers 7 and 8."**
4. Delete an account: **Delete the record for account number 5."**
5. **Transfer Funds Logic (transferFunds):Transfer the amount between two accounts.**

5. Transfer Funds Logic (transferFunds)

This function is the most complex because it requires reading from two different locations in the file and writing back to those same two locations. It relies heavily on Random Access.

The Logic Breakdown:

1. **Input Collection:** The program asks for the Sender ID, Receiver ID, and the Amount.
2. **Locating the Sender:**
    - It calculates the byte offset using the formula: (accountNum - 1) * sizeof(struct clientData).
    - It uses fseek to jump the file pointer directly to that location.
    - It uses fread to pull the sender's data into memory.
3. **Locating the Receiver:**
    - It repeats the fseek and fread process for the receiver's account number.
4. **Validation (Crucial Step):**
    - It checks if both accounts actually exist (acctNum != 0).
    - It checks if the sender has enough money (sender.balance < amount).
5. **The Transaction:**
    - It updates the variables in memory: sender.balance -= amount and receiver.balance += amount.
6. **Writing Data Back (The "Save"):**
    - **Important:** You must move the file pointer *back* to the specific record before writing.
    - fseek to the Sender's position

- fwrite the updated sender struct.
- fseek to the Receiver's position
- fwrite the updated receiver struct.

Submitted By,

Anusha S AP/CSE(Cyber Security)