Transaction Processing System

Mini project for CS3251 Lab

0. Review the code at https://replit.com/@ashokb/Unit5Programs#trans.c
1. Compile, and run the executable code
2. Do your own testing of the program
3. What happens when you add a new account detail? And then generate accounts.txt file again. Inspect the contents. Is the contents as you expected them to be?
4. If there are some errors, fix the program logic.
5. Add some new functionality. How about listing the accounts information?
6. Can you make the program more efficient?

Evaluation Criteria

No	Dimension	Activity	Points	Remarks
1	General (25)	Self-effort	5	Student's own assessment on the number of hours spent in completing the project on their own (no copying, no referring of code from others)
		Turnaround time	10	Provide three time slots; the earlier you come, the more points you will get
		Demo-ing the project	10	Using repl.it
2	Comprehension (15)	Domain knowledge	5	general questions regarding the domain of the project (no-code involved)
		Added functionality	5	Student's own ideas on how the project can be improved (no-code involved)
		Code Comprehension	5	must be able to respond to questions regarding specific code snippets
3	Modification (35)	Code Improvement	5	coding convention/style change from `camelCase` to `snake_case`, formatting improvements, if any
		Functional Decomposition	10	Choose 1-2 functions and decompose them properly; separate the user input part and the actual functionality
		Refactoring for Memory usage	10	How did you improve memory efficiency?
		Refactoring for Speed	10	How did you improve processing efficiency?
4	Innovation (45)	New features / user stories	5	As specified by faculty (limit on negative balance, ATM debit card transaction, sorting of records using algorithm of choice). Proper documentation will be evaluated
		New features / user stories	10	Translation of requirements to codification
		Added functionality (simple)	10	Error handling, for example
		Added functionality (advanced)	20	What new functionality did you implement?
	Total		120	Reduced to total of 20

Rules of Mini-project implementation

- http://j.mp/transactionUnit5 - you will be given a maximum of 2 attempts at presenting your mini-project. If you are unhappy with your first score, you will be given a 2nd opportunity. The best of both attempts will be your final score.
- You are encouraged have discussion and clarification amongst yourselves. But do not copy or borrow code from each other, whatsoever.
- You are strongly discouraged from copying other people's code. If found violating this rule, you will be scored with zero (0, NULL).