As part of the interview process, the prospect candidate is asked to solve the following task:

* Money laundering (ML) is the process of transforming the profits of crime and corruption into legitimate assets. To this end, criminal organizations and entities use the legal banking system to ingest money acquired from illegal activities, which is later withdrawn as legitimate payments. One ML technique used by criminals is the so-called "bridging", where an entity A sends money to an entity B via an entity C (i.e., A->C->B), creating a layer of dissociation between entities A and B. In one instance, a drug dealer 'John Smith' may want to pay his immediate drug distributor 'Mary Jerry'. They may then agree to use Mary's brother-in-law 'Joseph Louis' to intermediate the transaction, serving as a "bridge" between John and Mary. Joseph therefore will start forwarding every payment made by John to Mary and, depending on the agreement, may take a percentage of the payments as fees for his "services". According to financial crime investigators, bridging transactions usually occur within the same day with fees payed to the "bridging entity" going up to 10% of the original payment. Your goal is to write computer code that scans financial records in search of bridging transactions and entities suspected of bridging. You will use the attached zipped file "data/transactions.csv" containing financial transactions separated by pipes ('|'). Each row in the file is the record for 1 transaction, which is represented by a unique identifier, a timestamp (yyyy-mm-dd), the amount transferred (in US$), the sender's account, and the receiver's account. Feel free to use your preferred programming language and libraries. Keep in mind the scale of the real problem, focusing in low computational time and low memory utilization. The deliverable should contain your source code, a file with all suspicious transactions (1 per line), and a file with all suspicious entities (1 per line) ranked by number of suspicious transactions (high to low). Bonus points for parallel/multi-threaded solutions.

The transaction data file to test you code is available here: http://www.pvmiii.net/transactions.zip

You will be evaluated based on cleanness, clarity, performance, and innovation of your solution.

Should you have questions, do not hesitate to contact us.

Please email us your code when you are finished.

Good luck!

Phil removed

Leandro removed