INSIGHT DATA CHALLENGE: Coding round solution documentation

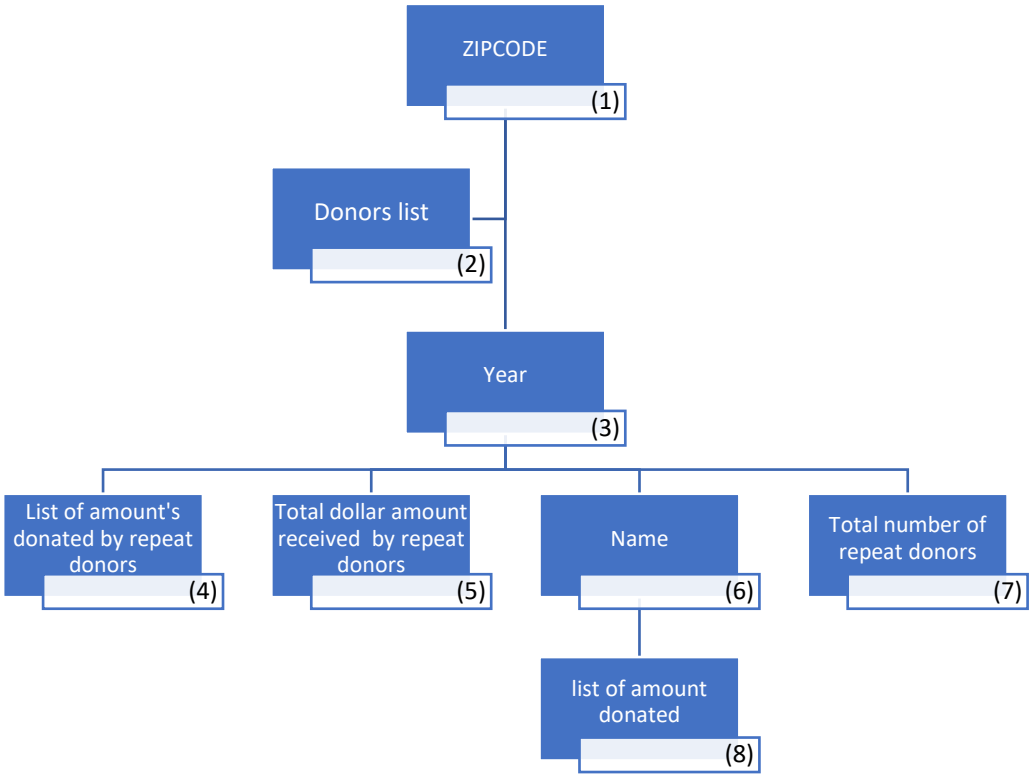
Coding challenge topic: Donation analytics.

In this document, I will only describe the solution. To know more about the coding challenge, look at the "readme.md" document in the challenge repo.

The programing language used to solve this challenge is python. My code will work with both version 2.7 and 3.6 of python. These are the following packages I have used.

- Numpy
- OS
- SYS

This is how the data is organized. I used a nested dictionary to organize my data. The names I used can be a bit different from what I used in the below figure. This is done only to give you a better idea of what is happening.



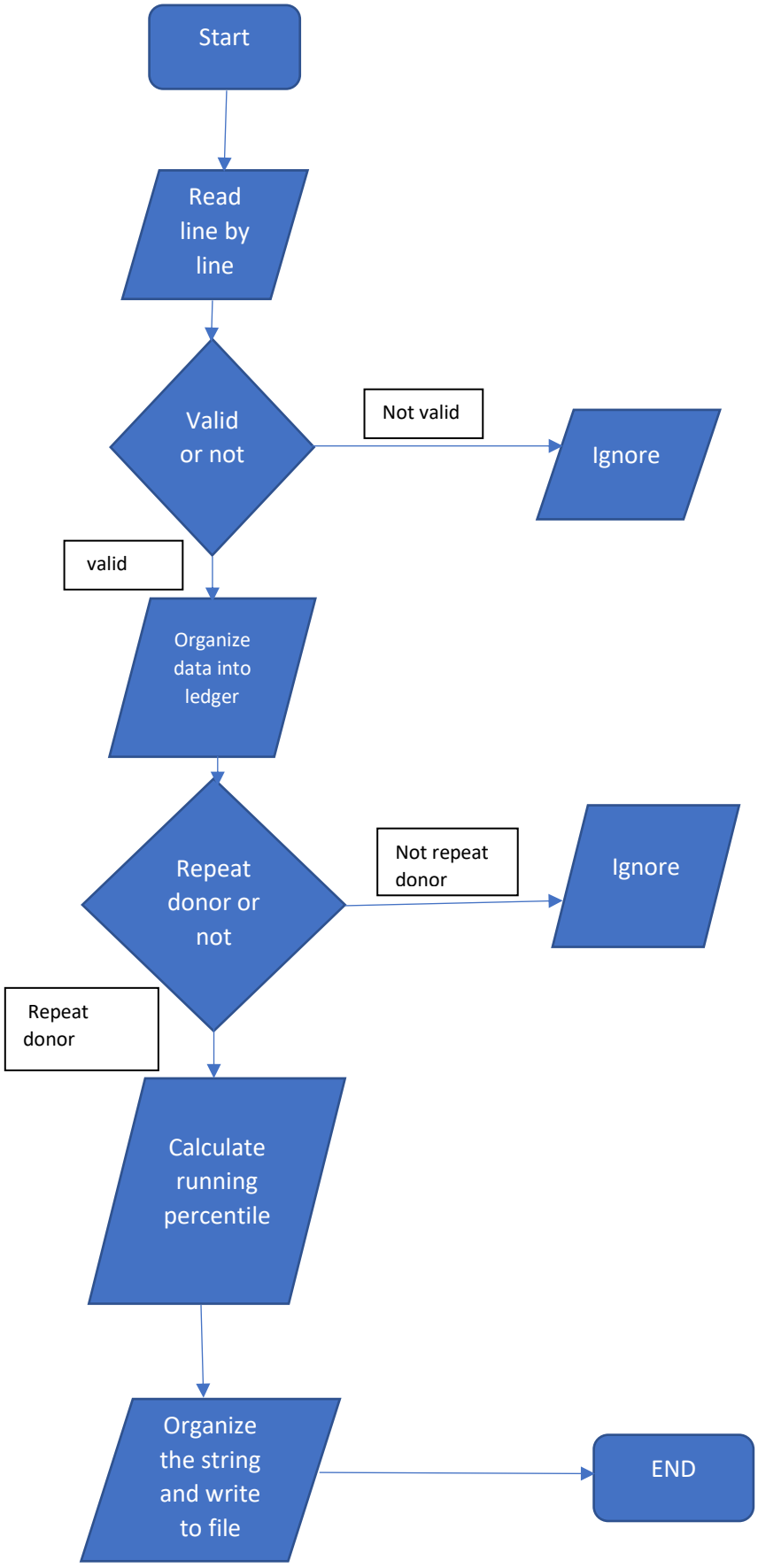
Below in detail, there is an explanation what each entity is about in the above chart

1. ZIPCODE: We are organizing our records based on the zip code that is extracted from each line of the record.
2. Donors List: This contains a list of all the donors living in an area with same zip code.
3. Year: We are organizing our records based on the year a person has donated.
4. List of amounts donated by repeat donors: Contains individual dollar amount received as donation by repeat donors
5. Total dollar amount received by repeat donors: Contain sum of donations received by repeat donors.
6. Name: Name of the donor
7. A total number of repeat donors: Number of repeat donors in that zip code.
8. List of the amount donated: We are using the list here as a person can donate multiple times in a year and we need to report the money donated by a person in a year.

Functions defined in the program:

- read_input_directory(): Access the record line by line and pass the information to check_valid_record(). If the record is valid to pass it functions process_data(). One of the inputs for this Stream. If the stream is set to False then execution will end after reaching end out "itcont.txt" file, if its set to True the program will not terminate but will handle data in real time.
- process_data(): This organizes the data into the ledger as per the conditions
- write_output(): Write our output to file mentioned
- running_percentile_calculator(): Function to calculate running percentile
- check_valid_record() : Checks if the record is valid or not.
- Main(): the function block from which execution starts.

Logic:


Start
Read line by line
Valid or not
Not valid
Ignore
valid
Organize data into ledger
Repeat donor or not
Not repeat donor
Ignore
Repeat donor
Calculate running percentile
Organize the string and write to file
END

Comments on the challenge:

it seems straight forward and easy. Much easier than previous challenge. Had difficulty in with running the run.tests.sh script but got that issue resolved.