Part B: Finding fingerprint groups

Design: by Yann Youbi

Spec: read stdin

given a set of names with fingerprints identify groups of names with a common fingerprint.

What data structure to use?

The structure that we would use is hash maps as it would allow us to store a mapping of keys of type k to values of type v.

It also allows us to look up data not linearly with indexing but by using a key that can be of any type.

What are the types of variables?

For our first hashmap, we would use the name for the key of type string and the fingerprint for value of type string.

We would make groups by using a second hash map, with a key of type string for a fingerprint and a value of type vector of strings for all the names associated with the fingerprint key.

What are my invariants? (Relate input to data structure...)

- For every name read, there is a key and its fingerprint as a string value in the hash map

- For every name string key read, there is our second hashmap containing its fingerprint as the key and the read name key as a string element of the vector value.

- For every fingerprint key in our second hashmap associated with more than one name (vector length > 1), there is a group containing all the names present in the vector value printed a name per line.

- For every group there is an empty new line separating each group printed

How do I solve the problem?

By making groups using that second hash map. with each fingerprint associated with a list of names.

Then printing only the names when the fingerprint has more than one name associated with it.

One name is printed per line

After the last name, the list of names constitutes a group.

After the first group the next groups are printed only after a blank new line is printed.