

Github link - <https://github.com/andreimoldovan23/FLCD>

The chosen data structure for the Symbol Table was HashMap, using separate chaining for collision resolution.

There is an initial number of slots declared, a variable keeping track of the total number of occupied slots, a list of lists of strings which will represent the map and a threshold.

If the load factor of the map (number of occupied slots divided by number of total slots) exceeds the threshold an operation of ResizeAndRehash will take place. This exists in order to optimize the search operations, to avoid a scenario in which all symbols are hashed to the same position.

When the get method will be called for a symbol, a search will occur first.
If the symbol is not found then it will be added and its position returned.

All positions are pairs of two indices.