Github repo: https://github.com/LolutaStefana/UBBComputerScience/tree/main/semester5/FLCD/lab2

I decided to create a single table that can be used for both the constants and identifiers tables, in addition to a single HashTable that can be utilized for both types of tables for my SymbolTable.

My hashtable has the form of lists inside lists, meaning that I essentially have a list for every position in the list.

The hashtable's size is returned by the getSize() method.

The position of the term is returned by the findPositionOfTerm(String term) method, which is calculated as follows: After computing the hash for that term, we verify that the list is not empty and if it is empty we return null. If it is not empty, we take the list from that position, parse it, and when we get to our terminal, we use the hash and the index of our terminal to build a pair from the list.

findByPos(Pair pos) method will return the terminal from the symbol table based on its position

The hash function I use is represented by hash(String key), and it computes the sum of the characters and applies the modulo size method.

add(String term) adds an element to the symbol table. containsTerm(String term) returns true or false depending on whether the element belongs to the symbol table or not.

The SymbolTable class implements every one of those methods, with the exception of the hash method.

I defined Pair as a data structure. It consists of two integer numbers.