For my SymbolTable I chose to implement one HashTable which can be used for both the identifiers table and constants table, as well as one single table which contains both of them.

My approach is to use an array of arrays. When hashing we get the position of the list wehere we want to insert the value, then we iterate to the first free position.

findPosOfValue(String value) method returns position of the givern value, which we find by hashing the value, then simply iterating through the found array (in O(1) time). We return a Pair of position where the first integer is the position of the list and the second is the position of the value inside that list

add(String value) – adds an element to the symbol table - Checks if the element exists in the table. If so returns false. Else hashes that value and finds the position of the element.

getHashTable() method returns a string representation of our table: ({postion_of_list}, {position_of_value_in_that_list}) -> {value}

All those methods, are implemented in the SymbolTable class.

I have also defined the Pair as a generic data structure. It is composed of an K key and a V value.