

<https://github.com/Szabo-Mark/FLCD>

The symbol table is just a wrapper over the HashTable:

HashTable implementation:

The class Position will hold the position of the bucket array and the position in the linked list of the token.

The class Node is a node in the linked list. It holds the value(the actual token) and the next node in the list.

The class HashTable has the public method add(String token).

It works in the following way:

-if the given token is already in the symbol table it will return the position of it(a Position object)

-if not, it firstly adds the token into the table and then returns its new position.

The collision strategy is using linked lists. Each slot in the table is the head of a linked list.

First we hash the given token and get the head of the linked list on that position.

We parse that linked list searching for the token. If it's not there, we add it at the end.

