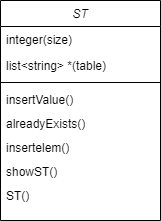
https://github.com/danielgabor99/FLCD/tree/master/Lab4

My ST is represented by a hash table with chaining. The hash table is an array of linked lists, each index has its own linked list.

void insertItem(string value):

Inserts a non existent element in the st

Pre: the element to be inserted

Post: the st with the inserted item

bool alreadyexists(string value):

Boolean that returns if a element is present already in the st

Pre: the element to be inserted

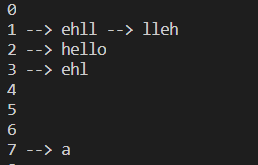
Post: true or false

int hashFun(string value):

The hash function that returns the index in st of the value

Pre: the value

Post: the index in the st

void showST():

The print of the st

Pre:

Post: the st

For the scanner I made all the operations in main():

if(find(tokenArray.begin(),tokenArray.end(),(\*it)[0])!=tokenArray.end())

            {

//if exists in token.txt

Pre:the actual token and the array from the token.txt

Post:inserts into pif all the tokens that exists in token.txt

ELSE

//if does not exist in token.txt

Pre:the actual token and the array from the token.txt

Post:inserts into pif and in the ST with the value from the Hash, if it is a constant, with a value 0;

For the lexicals erros I used regex((?<!,)\b([\d\.]+)\b(?!,))

