

## Scanner Doc

- Symbol Table structure:

I was assigned 1a and I implemented it using 2d - a Hash table

The hashing function was  $h(k) = (\text{val}(k) \bmod m) + 1$ ;

$\text{val}(k)$  represents the sum of the ASCII values of the characters composing the  $k$  value.

In case of conflict, when  $h(k) = h(x)$ ,  $k \neq x$ , we use a list where we keep the values with the same hashing.

The symbol table has one attribute (the hash table) and only one method "put" which assigns a value to the hash table if it hasn't been added before, and then returns the position of the value in the table: (x, y)

X - the position in the table

Y - the position in the list found at table(X)

- PIF structure:

The class has one attribute (the list) and only one method "put" with 2 parameters: token and ID, it adds the tuple of (token, ID) to the PIF table (the list)

- REGEX

**identifier regex:** `'^[a-zA-Z]([a-zA-Z][0-9]|_){0,10}$'`

- It starts with a character between [a-zA-Z], and then for 0 or more times, we have either a character between [a-zA-Z], or a digit [0-9], or the underline character

**const regex:** `'^(0|[-]?[1-9][0-9]*)$|'|\.'|\$|'|\\"'|\$'`

- (integer) - It's either 0, or it starts with one or none [-+], then we have one digit from [1-9], followed by zero or more digits from [0-9]
- (char) - it can be 1 character (.) between 2 of '
- (string) - zero or more characters (.\* ) between 2 of "

Class diagram:

