Lab2

https://github.com/AnastasiaSusciuc/UBB/tree/main/Anul 3/Sem5/FLCD/Labs/Lab2
(both the ST and the scanner are here)

I opted for the option when the STs are different for identifier respectively constants

SymbolTable

- is based on a Hash Table
- add/ remove has O(1) amortized complexity
- uses simple chaining (each bucket has a list that solves the collisions)
- I used the hash function from https://cp-algorithms.com/string/string-hashing.html, where p and m are some constants

$$\begin{aligned} \text{hash}(s) &= s[0] + s[1] \cdot p + s[2] \cdot p^2 + \ldots + s[n-1] \cdot p^{n-1} \mod m \\ &= \sum_{i=0}^{n-1} s[i] \cdot p^i \mod m, \end{aligned}$$

`add(self, key):`

```
"""
adds an element into the hashtable
:param key: the value of the element
:return: the hash value and the position of the element in its list
"""
```

`remove(self, key):`

```
"""
removes the key from hashtable
```

```
:param key: the element to be removed
:return: the hash value
"""
```

`exists(self, key):`

```
"""
:param key: the element we are looking for
:return: True if key is inside the hash table
"""
```