https://github.com/PopEmanuel/FLCD

I created a scanner for verifying the lexical correctness of my program. It uses regexPatterns to find what category each character belongs to.

I created a SymbolTable for holding the indentifiers using a custom HashTable

```
The hash algorithm :

 private final int hashCode = 4;
```

```
private int hash(String value) {
return value.hashCode() % hashCode;
}
```

We always work with strings so we use the hashCode of the string and use the modulo with a hashCode held in the class

The way the HashTable works is when you want to add a new value, it computes the hash value of the string, verifies if it is the first one with that index and if so adds it to index 1 otherwise it searches on the index to find the value, if it doesn't find it it will insert on the last index the value.

Getting the index of a value is easy, just searching on the index and returning the key value pair. Otherwise return (-1, -1)

I also created a PIF using an ArrayList from java. I add pifEntities to the List, in order to keep track of each identifier.
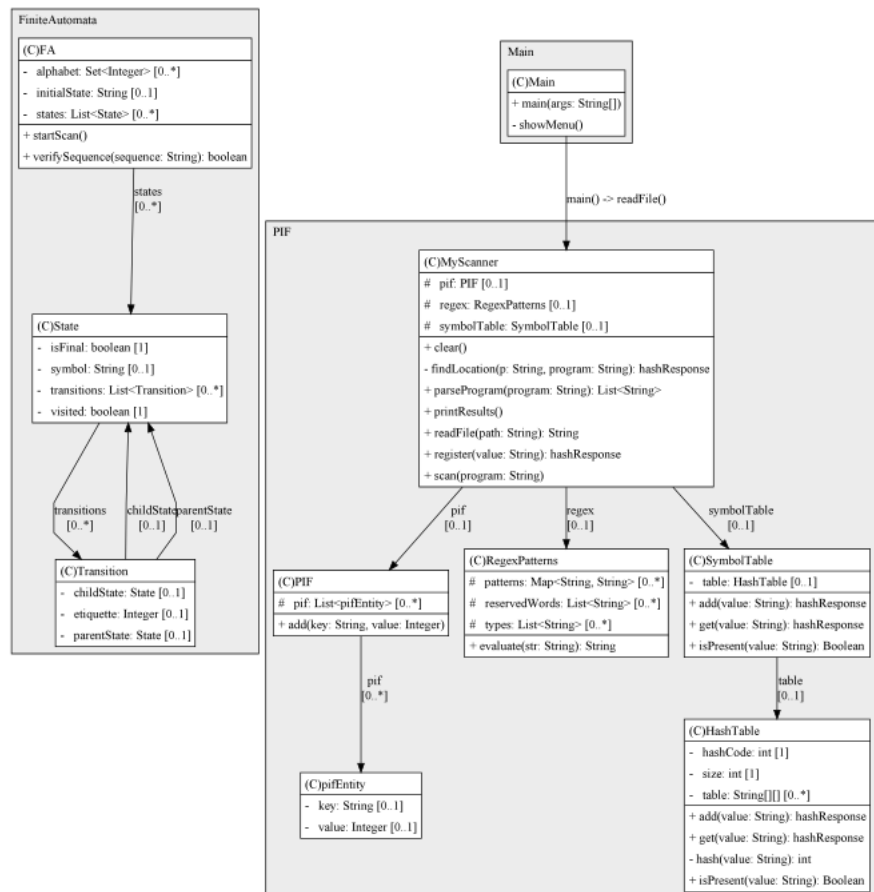
The scanner discovers the errors and provides the line and column that it happened.

The diagram class provides more info about the structure of the program

I created a Finite Automata using 3 classes, Transition, State and the Automata itself and the application now shows a menu with different options for it