Lab 7 Andrei Candet and Radu Ceaca

Link to git: https://github.com/cinnamonbreakfast/flcd/tree/main/lab5\_final\_%40raduceaca Assignment for a team of 2 students!

***Statement: Implement a parser algorithm (cont)***

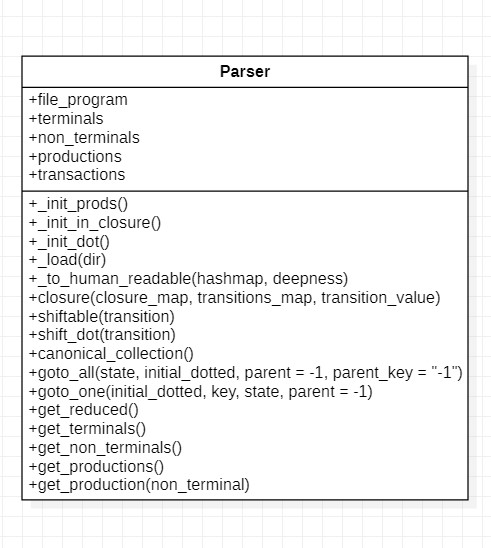
PART 2: Deliverables

1. Algorithm corresponding to parsing tables (if needed) and parsing strategy
2. Class ParserOutput - DS and operations corresponding to choice 2.a/2.b/2.c ([lab 5)](https://moodle.cs.ubbcluj.ro/mod/assign/view.php?id=2841) (required operations:

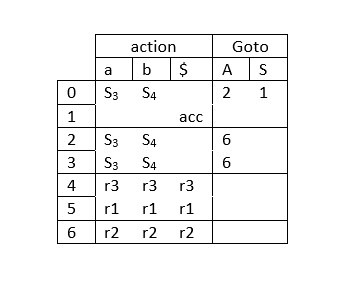
transform parsing tree into representation; print DS to screen and to file)

**Remark**:

- if the table contains conflicts, you will be helped to solve them. It is important to print message containing row (*symbol in LL(1), respectively state in LR(0)*) and column (symbol) where the conflict appears. For LL(1) values (αα,i) might also help



**+ parse\_string(string)**



\_init\_prods():

Initialize the production directory

\_init\_in\_closure()

Initialeze the closure map

\_init\_dot ()

Initialization method for closure

\_load(dir)

Load data from file closure(closure\_map, transitions\_map, transition\_value) Compute the closure\_map

shiftable(transition)

Check if the dot can be shifted

shift\_dot(transition)

Call the shiftable function and shift the dot

canonical\_collection()

Populate the canonical collection

goto\_all(state, initial\_dotted, parent=-1, parent\_key="-1")

Goes through every state

goto\_one(initial\_dotted, key, state, parent=-1)

Goes to a single state get\_reduced()

Returns the reduced map

get\_terminals()

Returns terminals

get\_non\_terminals()

Returns non terminals

get\_produtions()

Returns productions

get\_production(non\_terminal)

Returns the production of a non terminal

**parse\_string(string)**

**pre: string to parse**

**post: parsing table**