

<https://github.com/GamaCatalin/Parser>

Gama Marius + Ghetina Vlad

Grammar representation:

line 1: non-terminals separated by space

line 2: terminals separated by space

line 3: start symbol

line 4+: productions on each line following this rule:

non-terminal separated by '->' from symbols

and the symbols separated by " "

ex:  $S \rightarrow a b S$

CFG check:

1. We take the start symbol and check if it is a non-terminal
  - a. If it isn't a non-terminal we throw an error
  - b. If it is a non-terminal it continues
2. We take each key from the production dictionary
  - a. If any key isn't a non-terminal it throws an error
  - b. Then we check every move for each key
3. We take each move from the productions and check its steps
4. If any step can't be found in non-terminal and terminal symbols it throws an error
5. It returns true

Parser – recursive descendant with parsing tree –

<https://github.com/GamaCatalin/Parser/blob/main/RDParser.py>