

WHAT I DID

analysis1

analysis2

analysis3

A part of code transform to C

WHAT I DIDNT

analysis4

A part of code transform to C

HOW I DID

I created the correct forms of lex and yacc files

In analysis1

i count the ors(help of orcount) and i made a comparison with integer value of rule's nonterminals if they are not equal i gave a correct error message to terminal.

In analysis2

i kept b2(for rules) and b3(for conditions) arrays and i compared them if they are not includes correct forms i gave correct error message to terminal.

(For ex Lets say rule array contains extra nonterminal T and condition array contains extra A then i print Rule missing A and Condition missing T.)

In analysis3

i used my b2(for rules) array and i declared an array which name is "nonterminalfinder" with help of nonterminal array i find all elements after conditions arrow (E->) so that i am ready for comparing nonterminalfinder and b2 array and i compared them if i find a mistake i printed the correct error message

A part of code transform to C

i kept a CFlag to make sure there are any error if an error occurs it can not print the C code

In main function i started to write C output code i used terminal array to check typed enum part to write what i used

(arithmeticoperations,terminals) and i printed to terminal if it contains them.I used openparcount to check if i use open and close paranthesis.Finally i checked the b2 array(for rules) and i print nonterminals with help of it and i checked c array(nonterminal's integer counter) and again help of it i print the correct output to terminal

WHY I DIDNT

In analysis4

it seems like hard and it wants difficult algorithm to do i cant find the correct algorithm

A part of code transform to C

In main function i started to write C output code but i encounter with a hard problems to solve. I tried but i cant solve them

Mert UzgÃ¼l 20160702075