Grammar:

```
E     -> 'let' D 'in' E                          => 'let'
      -> 'fn'  Vb+ '.' E                         => 'lambda'
      ->  Ew;
Ew    -> T  'where' Dr                           => 'where'
      -> T;

T     -> Ta ( ',' Ta )+                          => 'tau'
      -> Ta ;
Ta    -> Ta 'aug' Tc                             => 'aug'
      -> Tc ;
Tc    -> B '->' Tc '|' Tc                        => '->'
      -> B ;

B     -> B 'or' Bt                               => 'or'
      -> Bt ;
Bt    -> Bt '&' Bs                               => '&'
      -> Bs ;
Bs    -> 'not' Bp                                => 'not'
      -> Bp ;
Bp    -> A ('gr' | '>' ) A                       => 'gr'
      -> A ('ge' | '>=') A                       => 'ge'
      -> A ('ls' | '<' ) A                       => 'ls'
      -> A ('le' | '<=') A                       => 'le'
      -> A 'eq' A                                => 'eq'
      -> A 'ne' A                                => 'ne'
      -> A ;

A     -> A '+' At                                => '+'
      -> A '-' At                                => '-'
      ->   '+' At
      ->   '-' At                                => 'neg'
      -> At ;
At    -> At '*' Af                               => '*'
      -> At '/' Af                               => '/'
      -> Af ;
Af    -> Ap '**' Af                              => '**'
      -> Ap ;
Ap    -> Ap '@' '<IDENTIFIER>' R                 => '@'
      -> R ;

R     -> R Rn                                    => 'gamma'
      -> Rn ;
Rn    -> '<IDENTIFIER>'
      -> '<INTEGER>'
      -> '<STRING>'
      -> 'true'                                  => 'true'
      -> 'false'                                 => 'false'
      -> 'nil'                                   => 'nil'
      -> '(' E ')'
      -> 'dummy'                                 => 'dummy' ;

D     -> Da 'within' D                           => 'within'
      -> Da ;
Da    -> Dr ( 'and' Dr )+                        => 'and'
      -> Dr ;
Dr    -> 'rec' Db                                => 'rec'
      -> Db ;
Db    -> Vl '=' E                                => '='
      -> '<IDENTIFIER>' Vb+ '=' E                => 'fcn form'
      -> '(' D ')' ;


Vb    -> '<IDENTIFIER>'
      -> '(' Vl ')'
      -> '(' ')'                                  => '()';
Vl    -> '<IDENTIFIER>' list ','                  => ','?;
```