/**

This EBNF follows the format from lecture notes
This EBNF contains custom rule to reduce repetition.
The custom rule is located at the bottom of the page.

**/

<basic_program> ::=

<declaration_unit>
|
<implementation_unit>

<declaration_unit> ::=

"DECLARATION"
"OF"
<ident>
["CONST" <constant_declaration>]
["VAR" <variable_declaration>]
[<type_declaration>]
[<procedure_interface>]
[<function_interface>]
"DECLARATION"
"END"

<procedure_interface> ::=

"PROCEDURE"
<ident>
[<formal_parameters>]

<function_interface> ::=

"FUNCTION"
<ident>
[<formal_parameters>]

<type_declaration> ::=

"TYPE"
<ident>
":"
<type>
";"

<formal_parameters> ::=

"("
<ident>
{ ";" <ident> }
")"

<constant_declaration> ::=

<constant_declaration_part>
{ "," <constant_declaration_part> }
";"

<variable_declaration> ::=

<variable_declaration_part>
{ "," <variable_declaration_part> }
";"

<type> ::=

<basic_type>
|
<array_type>

<basic_type> ::=

<ident>
|
<enumerated_type>
|
<range_type>

<enumerated_type> ::=

"{"
<ident>
{ "," <ident> }
"}"

<range_type> ::=

"["
<range>
"]"

<array_type> ::=

"ARRAY"
<ident>
"["
<range>
"]"
"OF"
<type>

<range> ::=

<number>
".."
<number>

<implementation_unit> ::=

"IMPLEMENTATION"
"OF"
<ident>
<block>
"."

<block> ::=

<specification_part>
<implementation_part>

<specification_part> ::=

[
("CONST" <constant_declaration>)
|
("VAR" <variable_declaration>)
|
<procedure_declaration>
|
<function_declaration>
]

<procedure_declaration> ::=

"PROCEDURE"
<ident>
";"
<block>
";"

<function_declaration> ::=

"FUNCTION"
<ident>
";"
<block>
";"

<implementation_part> ::=

<compound_statement>

<statement> ::=

<assignment>
|
<procedure_call>
|
<if_statement>
|
<while_statement>
|
<do_statement>
|
<for_statement>
|
<compound_statement>
;

<assignment> ::=

<ident>
"::="
<expression>

<procedure_call> ::=

"CALL"
<ident>

<if_statement> ::=

"IF"
<expression>
"THEN"
<compound_statement>
"END"
"IF"

<while_statement> ::=

"WHILE"
<expression>
"DO"
<compound_statement>
"END"
"WHILE"

<do_statement> ::=

"DO"
<compound_statement>
"WHILE"
<expression>
"END"
"DO"

<for_statement> ::=

"FOR"
"EACH"
<ident>
"IN"
<ident>
"DO"
<compound_statement>
"END"
"FOR"

<compound_statement> ::=

"BEGIN"
<semi_colon_and_statement>
"END"

<expression> ::=

<term>
[{ <add_subtract_term> }]

<term> ::=

<id_num>
[{ <multiply_divide_id_num> }]

<id_num> ::=

<ident>
|
<number>

<number> ::=

<digit>
{ <digit> }

<ident> ::=

<alphabet>
{ <alphabet> }

/**

Custom rules begin here.
7 custom rules are used in this EBNF.

**/

<constant_declaration_part> ::=

<ident>
"="
<number>

<variable_declaration_part> ::=

<ident>
":"
<ident>

<semi_colon_and_statement> ::=

<statement>
{ ";" <statement> }

<add_subtract_term> ::=

("+" | "-")
<term>

<multiply_divide_id_num> ::=

("*" | "/")
<id_num>

<digit> ::=

"0" | "1" | "2" | "3" | "4" |
"5" | "6" | "7" | "8" | "9"

<alphabet> ::=

"a" | "b" | "c" | "d" | "e" | "f" | "g" | "h" |
"i" | "j" | "k" | "l" | "m" | "n" | "o" | "p" |
"q" | "r" | "s" | "t" | "u" | "v" | "w" | "x" |
"y" | "z"