Rule :

① `<program> ::= begin <stmt_list> end`

② `<stmt_list> ::= <stmt> | <stmt> ; <stmt_list>`

③ `<stmt> ::= <var> := <expression>`

④ `<var> ::= A | B | C`

⑤ `<expression> ::= <var> + <var> | <var> - <var> | <var>`

Simbol awal : `<program>`

`<program>` → `begin <stmt_list> end` (rule 1)

→ `begin <stmt> ; <stmt_list> end` (rule 2)

→ `begin <var> := <expression> ; <stmt_list> end` (rule 3)

→ `begin A := <expression> ; <stmt_list> end` (rule 4)

→ `begin A := <var> + <var> ; <stmt_list> end` (rule 5)

→ `begin A := B + <var> ; <stmt_list> end` (rule 4)

→ `begin A := B + C ; <stmt_list> end` (rule 4)

→ `begin A := B+C ; <stmt> end` (rule 2)

→ `begin A := B+C ; <var> := <expression> end` (rule 3)

→ `begin A := B+C ; <var> := <var> end` (rule 5)

→ `begin A = B+C ; B := <var> end` (rule 4)

→ `begin A = B+C ; B := C end` (rule 4)