```
program tail> ::= <statement> program tail> | epsilon
<statement> ::= != !<for loop> |
<input operation> | <send command>
definition> ::= "on" "(" <string> "," <lambda expr> "," <lambda expr> ")"
<trigger call> ::= <string> "->" <expression>
<if statement> ::= "if" <expression> ":" <block> "end"
<for loop> ::= "for" <identifier> "in" <expression> ":" <block> "end"
<blook> ::= <statement> <block tail>
<blook tail> ::= <statement> <block tail> | epsilon
<lambda expr> ::= "lambda" <identifier> ":" <lambda body> "end"
<lambda body> ::= <expression> | <statement> <block tail>
<input operation> ::= <identifier> "=" "input" "(" ")"
<send command> ::= "send command" "(" <string> "," <string> ")"
<expression> ::= <boolean expr> | <arithmetic expr> | <function_call> | <string> | <identifier> |
<number>
<boolean expr> ::= <expression> <boolean op> <expression> | "True" | "False"
<boolean op> ::= "and" | "or" | "not" | "==" | "!=" | "<" | ">" | "<=" | ">="
<arithmetic expr> ::= <expression> <arith op> <expression>
<arith op> ::= "+" | "-" | "*" | "/" | "%"
<function call> ::= <identifier> "(" <argument list> ")"
<argument list> ::= <expression> <argument list tail> | epsilon
<argument_list_tail> ::= "," <expression> <argument_list_tail> | epsilon
<string> ::= "" <string characters> ""
<string characters> ::= <string character> <string characters> | epsilon
<string character> ::= <letter> | <digit>
<identifier> ::= <letter> <identifier tail>
<identifier tail> ::= <letter or digit or underscore> <identifier tail> | epsilon
<letter or digit or underscore> ::= <letter> | <digit> | " "
<letter> ::= "a" | ... | "z" | "A" | ... | "Z"
<digit> ::= "0" | ... | "9"
<number> ::= <digit> <number tail>
<number tail> ::= <digit> <number tail> | epsilon
```