PROGRAM = {INCLUDE}, TYPE, "main", "(" , {IDENTIFIER}, ")", "{",

        {STATEMENT}, ["return", NUMBER| "0"| IDENTIFIER| "" | {ASCII
CODE},";"] "}" .

LETTER = "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" | "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" .

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

NONZERONUMBER = "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9",

           {DIGIT} .

IDENTIFIER = LETTER{ LETTER | DIGIT | "_"} ;

NUMBER = ["-"] , "0" | NONZERONUMBER;

INCLUDE = "#include", ("<", IDENTIFIER , ">" , ";") | (" " "
      , IDENTIFIER, " " ", ";") .

TYPE = "int" | "bool" | "std::string" | "double" | "float" .

USERTYPE = LETTER, {LETTER | DIGIT | "_"}.

ARRAY = TYPE | USERTYPE, IDENTIFIER, "[", NONZERONUMBER, {NUMER}, "]",
    ";" .

STATEMENT = VARIABLEDECLARATION | ASSIGMENT | **ARITHMETICASSIGMENT** |

      (IDENTIFIER, "(",{IDENTIFIER}, ")") .

VARIABLEDECLARATION = (TYPE | USERTYPE, IDENTIFIER) | ARRAY, ";" .

ASSIGMENT = IDENTIFIER, "=", ("new", USERTYPE | TYPE)

      | VARIABLE | NUMBER , ";" .

BREAK = "break", ";" .

ARITHMETICASSIGMENT = (IDENTIFIER, "++" | "—") | ("++"
                | " -", IDENTIFIER) | (["+" | "-"], IDENTIFIER,
            {"+" | "-" | "*" | "/" , IDENTIFIER}, ";" .
CONDITION = IDENTIFIER | NUMBER, ("="|"<"|"<="|">"|">="

```
                    | "!="), IDENTIFIER | NUMBER | "true"| "false" .

IF = "if" , "(" , CONDITION, ["&&" | "||"] {CONDITION}, ")" ,
     STATEMENT | ("{" , STATEMENT, {STATEMENT} , "}" )
     [("else",STATEMENT) | "elseif", "(" , CONDITION, ["&&" | "||"]
     {CONDITION}, ")" ,"{", {STATEMENT},"}" .
```

```
OUTPUT = "std::cout<<", IDENTIFIER | NUMBER | ("" , {ASCII CODE},
         ""), ";" .
INPUT = "std::cin>>", IDENTIFER, ";" .
LOOP = "while", "(", ("true" | "false") | ( CONDITION, {"&&" |
       "||", CONDITION},")", "{", STATEMENT, {STATEMENT}, "}".
```