Context-free Grammar Expression

BNF for LR(1) Parser

Compiler Theory - Programming Project 4 : LR(1) Parser
21400646 Lim Chae Eon

BNF Expression

Italic word : Non-Terminal, **blue colored word : Terminal**

```
1  # Script Definition part
2  Program              →    <script_start> stmt-sequence <script_end>
3  stmt-sequence        →    statement ; stmt-sequence | comment nextLine | ε
4  statement            →    if-stmt | declaration-stmt | loop-stmt | assign-stmt
                             | switch-stmt | function-stmt | increment-stmt  | break
5  simple-stmt          →    stmt-sequence | { stmt-sequence }
6
7  # If-Conditional Statement Part
8  if-stmt              →    if ( exp )simple-stmt
                             | if ( exp ) simple-stmt else simple-stmt
9
10 # Variable Declaration Statement Part
11 declaration-stmt     →    var id
12 id                   →    id , id | assign-stmt | IDENTIFIER
13
14 # Assignment Definition Statement Part
15 assign-stmt          →    IDENTIFIER assign-op exp
16 assign-op            →    = | -= | +=
17
18 # Operation Expression Part
```

```
19  exp          →   simple-exp logic-op simple-exp | simple-exp
20  logic-op     →   < | > | <= | >= | == | !=
21  simple-exp   →   simple-exp add-op term | term
22  add-op       →   + | −
23  term         →   term mul-op factor | factor
24  mul-op       →   * | /
25  factor       →   ( exp ) | NUMBER | IDENTIFIER
26
27  # Loop Statement Part
28  loop-stmt    →   for ( for-parameter ) simple-stmt
                     | while ( exp ) simple-stmt
29  for-parameter →  exp ; exp ; exp
30
31  # Function Statement Part
32  function-stmt →  function-keyword ( function-parameter )
33  function-keyword →  window.prompt | window | parseFloat
                        | document.writeln | document.write | document
34  function-parameter →  IDENTIFIER | LITERAL | NUMBER
35
36  # Increment/Decrement Operation Part
37  increment-stmt →  increment-op IDENTIFIER | IDENTIFIER increment-op
38  increment-op →  ++ | −−
39
40  # Switch-Conditional Statement Part
41  switch-stmt  →   switch ( IDENTIFIER ) { case-part default-block }
42  case-part    →   case-block case-part | ε
```

```
43  case-block          →    case-condition : stmt-sequence
44  case-condition      →    case case-parameter
45  case-parameter      →    NUMBER | LITERAL
46  default-block       →    default : stmt-sequence |  ε
47
48  # Comment Part
    comment             →    // anything
```