RomanScriptLang - Compiler Project

# Grammar Rules

Variable Declaration:  
 Declaration -> ghoshit ID = Expression ;  
 Declaration -> ghoshit ID ;  
  
Operations (+, -, \*, /, >, <, ==, <=, >=, !=):  
 Expression -> Expression + Term  
 Expression -> Expression - Term  
 Expression -> Term  
 Term -> Term \* Factor  
 Term -> Term / Factor  
 Term -> Factor  
 Factor -> ( Expression )  
 Factor -> ID  
 Factor -> NUM  
 Condition -> Expression Relop Expression  
 Relop -> < | > | == | <= | >= | !=  
  
Condition Checking (If, Else, If-Else Ladder):  
 IfStatement -> yadi ( Condition ) { Statements }  
 IfElseStatement -> yadi ( Condition ) { Statements } anyatha { Statements }  
 IfElseLadder -> yadi ( Condition ) { Statements } ElseIfBlocks [anyatha { Statements }]  
 ElseIfBlocks -> ElseIfBlock ElseIfBlocks  
 ElseIfBlocks -> ε  
 ElseIfBlock -> anyatha\_yadi ( Condition ) { Statements }  
  
Return Statement:  
 ReturnStatement -> lautao Expression ;  
  
Break and Continue Statements:  
 BreakStatement -> todo ;  
 ContinueStatement -> aage\_bhado ;  
  
Loops (For, While):  
 WhileLoop -> jabtak ( Condition ) { Statements }  
 ForLoop -> ke\_liye ( DeclarationOrAssignment ; Condition ; Assignment ) { Statements }  
 DeclarationOrAssignment -> ghoshit ID = Expression  
 DeclarationOrAssignment -> Assignment  
 Assignment -> badlo ID = Expression  
  
Comments:  
 SingleLineComment -> ## TEXT  
 MultiLineComment -> ##\* TEXT \*##

# Example Programs

## Variable, Operations, Print

ghoshit a = 5;  
ghoshit b = 10;  
ghoshit sum;  
  
badlo sum = a + b;  
  
chhapo sum;

## Condition Checking (If-Else)

ghoshit num = 7;  
  
yadi (num % 2 == 0) {  
 chhapo num;  
} anyatha {  
 chhapo num + 1;  
}

## If-Else Ladder

ghoshit marks = 75;  
  
yadi (marks >= 90) {  
 chhapo "Grade A";  
} anyatha\_yadi (marks >= 75) {  
 chhapo "Grade B";  
} anyatha\_yadi (marks >= 50) {  
 chhapo "Grade C";  
} anyatha {  
 chhapo "Fail";  
}

## Loops (While + For)

ghoshit i = 0;  
  
jabtak (i < 5) {  
 chhapo i;  
 badlo i = i + 1;  
}  
  
ke\_liye (ghoshit j = 0; j < 5; badlo j = j + 1) {  
 chhapo j;  
}

## Break and Continue

ghoshit i = 0;  
  
jabtak (i < 10) {  
 badlo i = i + 1;  
  
 yadi (i == 3) {  
 aage\_bhado;  
 }  
  
 yadi (i == 7) {  
 todo;  
 }  
  
 chhapo i;  
}

## Return Statement

ghoshit addResult;  
  
badlo addResult = 5 + 10;  
  
lautao addResult;

## Comments

## This is a single-line comment  
  
##\*  
This is a   
multi-line comment  
\*##