Compiler Construction (CS F363)

Assignment -1

Birla Institute of Technology and Science Pilani
Hyderabad Campus

Under the supervision of

Dr. Raghunath Reddy M

Submitted by

NAME	ID NUMBER	EMAIL
Kush Gadia	2021B4A72386H	f20212386@hyderabad.bits-pilani.ac.in
Arjun Tyagi	2021B3A72272H	f20212272@hyderabad.bits-pilani.ac.in
Anusha Makharia	2021B3A72681H	f20212681@hyderabad.bits-pilani.ac.in
Tanmay Agarwal	2021B3A73040H	f20213040@hyderabad.bits-pilani.ac.in

CONTEXT FREE GRAMMAR (CFG)

Language Specification

- <lowercase> → 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
- <digits> → 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
- <special> → + | - | % | / | * | < | > | = | _ | (|) | ; | , | : | { | }
- <alpha> → @
- <int_const> → (<decimal>, 10) | (<octal>, 8) | (<binary>, 2)
- <decimal> → <digits><decimal> | <digits>
- <octal> → <o> <octal> | <o>
- <o> → 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7
- <binary> → <b> <binary> | <b>
- <b> → 0 | 1
- <char> → '<c>'
- <c> → <lowercase> | <digits> | <nL> | <tab>
- <nL> → \n
- <tab> → \t
- <string> → "<s>"
- <s> → <s1> <s> | <s1>
- <s1> → <lowercase> | <digits> | <special> | <alpha>
- <keyword> → int | char | if | else | while | for | main | begin | end | print | scan | VarDecl | program | inc | dec
- <type> → int | char
- <var_name> → <lower_case> <rest_no_underscore>
- <rest_no_underscore> → <alpha_num> <rest_no_underscore> | _ <rest_with_underscore> | ε

- <rest_with_underscore> → <alpha_num> <rest_with_underscore> | ε
- <alpha_num> → <lower_case> | <digit>

Start

- <program> → begin program : <statements > end program

Statement Block

- <statements > → <statement> <statements > | <statement>
- <statement> → <assignment> | <if_statement> | <for_statement> | <while_statement> | <Var_Dec> | <blck_stmt> | <print_stmt> | <scan_stmt> | ε
- <blck_stmt> → begin <statements_1> end
- <statements_1> → <stmt_1> <statements_1> | <stmt_1>
- <stmt_1> → <assignment> | <print_stmt> | <scan_stmt>
- <Var_Dec> → begin VarDecl : <decl> end VarDecl
- <decl> → (<var_name>, <type>); <decl> | (<var_name>, <type>);

Conditional Statement: if

- <if_statement> → if <relational> begin <statements_1 > end; |
  if <relational> begin <statements_1> end else begin <statements_1 > end ;

For and While Loop

- <for_statement> → for <var_name> := <expr> <to_part> <update> <int_const> do begin <statements_1> end ;
- <to_part> → to <expr> | ε
- <update> → inc | dec
- <while_statementt> → while (<relational_x>) begin <statements_1> end ;

- <relational_x> → <relational> | <var_name> <rel_op> <omg> |
  <omg> <rel_op> <var_name>
- <omg> → <decimal> | <int_const>
- <rel_op> → = | > | < | <= | >= | <>

Print and Scan

- <print_stmt> → print (<string> , <additional>) ; | print (<string>) ;
- <additional> → <xyz>, <additional> | <xyz>
- <xyz> → <var_name> | <int_const> | <char>
- <scan_stmt> → scan (" <pqr> " , <additional_1>) ;
- <pqr> → <alpha> , <pqr> | <pqr>
- <additional_1> → < var_name>, <additional_1> | <var_name>

Arithmetic / Assignment and Relational Operators

- <assignment> → <var_name> <assignment_op> <expr>
- <expr> → <int_const> | <var_name> | <arithmetic>
- <arithmetic> → <arithmetic> + <arithmetic_1> | <arithmetic> - <arithmetic_1>|
  <arithmetic_1>
- <arithmetic_1> → <arithmetic_1> * <arithmetic_2> | <arithmetic_1> / <arithmetic_2> |
  <arithmetic_1> % <arithmetic_2> | <arithmetic_2>
- <arithmetic_2> → <int_const> | <var_name> | <decimal>
- <relational> → <relational> = <realtional_1> | <relational> <> <relational_1> |
  <relational_1>
- <relational_1> → <relational_1> < <relational_2> | <relational_1> <= <relational_2> |
  <relational_1> > <relational_2> | <relational_1> >= <relational_2> | <relational_2>
- <relational_2> → <int_const> | <var_name> | <decimal> | <arithmetic>