**- Invalid statement**

Program Example1; (\* Example 1 \*)

Begin

Integer

End. (\* Example 1 \*)

**- Invalid Type**

PROGRAM EXAMPLE4; (\* Example 4 \*)

CONST MAX = 10;

TYPE T = 8;

**- Invalid basic Type**

Program Example2; (\* Factorial \*)

Var n : Integer;

Function F(n : Integer1)

**- Invalid param**

Program Example2; (\* Factorial \*)

Var n : Integer;

Function F() : Integer;

Begin

If n = 0 Then F := 1 Else F := N \* F (N - 1);

End;

**- Invalid term**

Program Example2; (\* Factorial \*)

Var n : Integer;

Function F(n : Integer) : Integer;

Begin

If n = 0 F := 1 Else F := N \* F (N - 1);

End;

**- Invalid argument**

Program Example2; (\* Factorial \*)

Var n : Integer;

Function F(n : Integer) : Integer;

Begin

If n = 0 Then F := 1 Else F := N \* F (N - 1;

End;

**- Invalid expression**

Program Example2; (\* Factorial \*)

Var n : Integer;

Function F(n : Integer) : Integer;

Begin

If For For Then F := 1 Else F := N \* F (N - 1);

//If n = 0 Then F := 1 Else F := N \* F (N - 1);

End;

**- Invalid Factor**

Program Example2; (\* Factorial \*)

Var n : Integer;

Function F(n : Integer) : Integer;

Begin

If n =+= 0 Then F := 1 Else F := N \* F (N - 1);

End;

**-Invalid comparator**

Program Example2; (\* Factorial \*)

Var n : Integer;

Function F(n : Integer) : Integer;

Begin

If n )= 0 Then F := 1 Else F := N \* F (N - 1);

End;

**- Invalid constant**

Program Example2; (\* Factorial \*)

Const a = -;

Var n : Integer;