**Program** -> **Heading Uses Declaration Execution**

**Heading** -> program identifier

**Uses** -> uses **Library** | ε

**Library** -> library **Library2** |

**Library2** -> , library Library2 | ε

**Declaration** -> **Const Type Var Function Procedure**

**Const** -> const \n identifier = value ; **Const2** | ε

**Const2** -> identifier = value ; **Const2** | ε

**Type** -> type \n identifier = datatype; **Type**2 |ε

**Type2** -> identifier = datatype; **Type2** | ε

**Var** -> var **Var2 |** ε

**Var2** -> , identifier **Var2**|: type ; **Var2** | identifier **Var2 |** ε

**Function** -> function identifier **Var** begin **Statements** end **Function**| ε

Statements -> **Statement** ; \n **Statements** | ε

**Statement** -> **Assign** | **IFClause** | **Forloop** | **Repeatloop** | read(identifier)| readln(identifier) | **WriteLn** | **Write** | **Whileloop**

**Whileloop ->** while ( **Condition** ) do begin **Statements** end ;

**WriteLn** -> writeln ( **Writelndash** | writeln (**Writelndash**

**Writelndash ->** identifier **WriteLn2 |** string **WriteLn2**

**WriteLn2** -> , identifier **WriteLn2**| , string **WriteLn2** | ) ;

**Write** -> write ( **Writedash** | write ( string **Writedash**

**Writedash ->** identifier **Write2 |** string **Write2**

**Write2 ->** , identifier **Write2**| , string **Write2** | ) ;

**Assign** -> identifier := **Expression**;

**Expression** -> **Term AddOp Expression** | **Term**

**Term** -> **Factor MultOp Term** | **Factor**

**Factor** -> identifier | constant

**AddOp** -> + | \_

**MultOp** -> \* | / | div | mod | and

**RelatOp** -> = | <> | > | < | <= | >=

**IFClause** -> if (**Condition)** then **Statements** else **Statements**

**Condition** -> **Expression RelatOp Expression**

**Forloop** -> for identifier := **Factor** to **Factor** do **Statements**

**RepeatLoop** -> repeat **Statements** until **Condition**;

**Procedure** -> procedure identifier **Var** begin **Statements** end **Procedure**| ε

**Execution** -> begin **Statements** end