Context Free Grammar for Tiny Programming Language

**1: Program→ Function Program | MainFunction**

**2: Function→ FunctionDecl FunctionBody**

**3: FunctionDecl→ Decl1Var “(“ ParametersDecl “)“**

**4: Decl1Var→ DataType identifier**

**5: DataType→ int | float | string**

**6: ParametersDecl→** **Parameters | ε**

**7: Parameters→ Parameters “,” Decl1Var |Decl1Var**

**7.1: Parameters→ Decl1Var Parameters′**

**7.2: Parameters′→ “,” Decl1Var Parameters′ | ε**

**8: FunctionBody → “{“ ManyStatements ReturnStatement “}“**

**9: ManyStatements→ StatementForFunctionBody ManyStatements | ε**

**10: StatementForFunctionBody→ IfStatement | WriteStatement | ReadStatement   
| AssignmentStatement | FunctionCall  
| RepeatStatement | DeclarationStatement**

**11: IfStatement→ if ConditionStatement then StatementsForIf Ifff**

**12:ConditionStatement→ ConditionStatement BooleanOperator Condition| Condition**

**12.1:ConditionStatement→ Condition ConditionStatement′**

**12.2:ConditionStatement′→ BooleanOperator Condition ConditionStatement′| ε**

**13:Condition→ identifier ConditionOperator Term**

**14: ConditionOperator → < | > | <> | =**

**15: Term → FunctionCall | number | identifier**

**15: FunctionCall → identifier “(“ ManyIdentifiers “)“ “;”**

**16: ManyIdentifiers → Identifiers | ε**

**17: Identifiers → Identifiers “,” identifier | identifier**

**17.1: Identifiers → identifier Identifiers′**

**17.2: Identifiers′ → “,” identifier Identifiers′ | ε**

**18: BooleanOperator → && | ||**

**19: StatmentsForIf→ Statement StatmentsForIf | Statement**

**19.1: StatmentsForIf→ Statement StatmentsForIf′**

**19.2: StatmentsForIf′→ StatmentsForIf | ε**

**20:Statement → IfStatement | WriteStatement | ReadStatement   
| AssignmentStatement | FunctionCall  
| RepeatStatement | DeclarationStatement | ReturnStatement**

**20: Ifff → ElseIfStatement | ElseStatement | end**

**21: ElseIfStatement→ elseif ConditionStatement then StatmentsForIf Ifff**

**22: ElseStatement→ else StatmentsForIf end**

**23: WriteStatement→ write Something “;”**

**24: Something → Expression | endl**

**25: Expression → Equation | constantstring**

**26: Equation → Equation ArthimiticOperator HelperTerm | HelperTerm**

**26.1: Equation → HelperTerm Equation′**

**26.2: Equation′ → ArthimiticOperator HelperTerm Equation′ | ε**

**27: HelperTerm → Term | HelperEquation**

**28: HelperEquation → “(“ ManyTerms“)“**

**29: ManyTerms → ManyTerms ArthimiticOperator Term| Term**

**29.1: ManyTerms → Term ManyTerms′**

**29.2: ManyTerms′ → ArthimiticOperator Term ManyTerms′ | ε**

**30: ArthimiticOperator → + | - | \* | /**

**31: ReadStatement → read identifier “;”**

**32: ReturnStatement → return Expression “;”**

**33: AssignmentStatement → identifier “:=” Expression “;”**

**34: RepeatStatement → repeat StatmentsForIf until ConditionStatement**

**35:DeclarationStatement → DataType ManyIdentifiersDecl “;”**

**36: ManyIdentifiersDecl → ManyIdentifiersDecl “,” identifier AssignmentInDecl | identifier AssignmentInDecl**

**36.1: ManyIdentifiersDecl → identifier AssignmentInDecl ManyIdentifiersDecl′**

**36.2: ManyIdentifiersDecl′ → “,” identifier AssignmentInDecl ManyIdentifiersDecl′| ε**

**37: AssignmentInDecl → “:=” Expression| ε**

**38: MainFunction→ DataType main “(” “)” FunctionBody**