Milestone 2
Tiny Language CFG
Compiler Theory

Name	Section	ID
عبدالرحمن مصطفي حسن احمد	3	2022170242
عبدالله شريف سمير محمود	3	2022170222
عبدالرحمن تامر محمد عبدالفتاح	3	2022170230
زياد اشرف رشاد شرف	2	2022170165
صفوة ابراهيم صلاح ابراهيم	3	2022170214
عبدالرحمن احمد صابر حسن	3	2022170226

❖ Terminals

- Reserved Words
  - int float string read write repeat until if elseif else end then return endl main
- Arithmetic Operators
  - + - * /
- Condition Operators
  - < > <> =
- Boolean Operators
  - && ||
- Operators
  - ( ) { } , := [ ] ;
- Others
  - Identifier String number

❖ Production Rules

1. Program → Function_Statements Main_Function
2. Function_Statements → Function_Statement Function_Statements | ε
3. Function_Statement → Function_Declaration Function_Body
4. Function_Declaration → Datatype Function_Name (Function_Declaration'
5. Function_Declaration' → Parameters) |)
6. Main_Function → Datatype main ( ) Function_Body
7. Function_Body → { Statements Return_Statement }
8. Function_Name → identifier
9. Parameters → Parameter Parameters'
10. Parameter → Datatype identifier
11. Parameters' → , Parameter Parameters' | ε
12. Statements → Statement Statements | ε
13. Statement → Assignment_Statement ; | Read_Statement | Write_Statement | If_Statement | Repeat_Statement | Return_Statement | Decleration_Statement | ε
14. Term → number | identifier | Function_Call
15. Datatype → int | float | string
16. Function_Call → identifier (Function_Call'
17. Function_Call'→ IdList) |)
18. IdList → Term IdList'
19. IdList' → , Term IdList' | ε
20. Declaration_Statement → Datatype Ids ;
21. Ids → IdDecl Ids'
22. IdDecl → Assignment_Statement | identifier
23. Ids' → , IdDecl Ids' | ε
24. Assignment_Statement → identifier := Expression

25. Expression → Term | Equations | String
26. Equations → Term_Eq Equation
27. Term_Eq → Term | (Equations)
28. Equation → Arithmatic_Operator Equations | ε
29. Write_Statement → write Write_Statement’
30. Write_Statement’ → Expression ; | endl ;
31. Read_Statement → read identifier ;
32. Return_Statement → return Expression ;
33. If_Statement → if Condition_Statement then Statements Else
34. Else → Else_If_Statment | Else_Statment | end
35. Else_If_Statment → elseif Condition_Statement then Statements Else
36. Else_Statement → else Statements end
37. Repeat_Statement → repeat Statements until Condition_Statement
38. Condition_Statement → Condition Condition_Statement'
39. Condition → identifier Condition_Operator Term
40. Condition_Statement' → Boolean_Operator Condition Condition_Statement' | ε
41. Arithmatic_Operator → + | - | * | /
42. Boolean_Operator → && | ||
43. Condition_Operator → <> | < | > | =

❖ Elimination

• Left Recursion

1. Function_Statements → Function_Statements Function_Statement | ε
   - Function_Statements → Function_Statements'
   - Function_Statements' → Function_Statement Function_Statements' | ε
2. Statements → Statements Statement | Statement
   - Statements → Statement Statements'
   - Statements' → Statement Statements' | ε
3. Parameters → Parameters , Parameter | Parameter
   - Parameters → Parameter Parameters'
   - Parameters' → , Parameter Parameters' | ε
4. IdList → IdList , Term | Term
   - IdList → Term IdList'
   - IdList' → , Term IdList' | ε
5. Ids → Ids , IdDecl | IdDecl
   - Ids → IdDecl Ids'
   - Ids' → , IdDecl Ids' | ε
6. Condition_Statement → Condition_Statement Boolean_Operator Condition | Condition
   - Condition_Statement → Condition Condition_Statement'
   - Condition_Statement' → Boolean_Operator Condition Condition_Statement' | ε

• Left Factoring

1. Function_Declaration → Datatype Function_Name (Parameters) | Datatype Function_Name ( )
   - Function_Declaration → Datatype Function_Name (Function_Declaration'
   - Function_Declaration'→ Parameters) |)
2. Function_Call → identifier (IdList) | identifier ( )
   - Function_Call → identifier (Function_Call'
   - Function_Call'→ IdList) |)

3. Write_Statement → write Expression ; | write endl ;
   - Write_Statement → write Write_Statement'
   - Write_Statement' → Expression ; | endl ;