Stage 1: First sets, Follow sets and Modified grammar submission

Group - 10

1) Akash S Revankar (2019A7PS0294P)
2) Harsh Butani (2019A7PS0022P)
3) Hemant Singh Sisodiya (2019A7PS0070P)
4) Mohit Sharma (2019A7PS0100P)
5) Siddharth Upadhyay (2019A7PS0033P)

Modified Grammar

1) <program> → <other Functions> <main Function>

2) <otherFunctions> → <function> <otherFunctions> | ε

3) <mainFunction> → TK_MAIN <stmts> TK_END

4) <function> → TK_FUNID <input_par> <output_par>
TK_SEM <stmts> TK_END

5) <stmts> → <definetype Stmts> <typeDefinitions> <declarations>
<otherStmts> <return Stmt>

6) <input_par> → TK_INPUT TK_PARAMETER TK_LIST
TK_SQL <parameter_list> TK_SQR

7) <output_par> → TK_OUTPUT TK_PARAMETER TK_LIST
TK_SQL <parameter_list> TK_SQR | ε

8) <definetypeStmts> → <definetypestmt> <definetypeStmts> | ε

9) <definetypestmt> → TK_DEFINETYPE <A> TK_RUID TK_AS
TK_RUID TK_SEM

10) <typeDefinitions> → <typeDefinition> <typeDefinitions> | ε

11) <typeDefinition> → TK_RECORD TK_RUID <recordDefinitions>
<fieldDefinitions> TK_ENDRECORD |
TK_UNION TK_RUID <fieldDefinitions>
TK_ENDUNION

12) <declarations> → <declaration> <declarations> | ε

13) <declaratin> → TK_TYPE <dataType> TK_COLON TK_ID
<global_or_not> TK_SEM

14) <otherStmts> → <stmt><otherStmts> | ε

15) <stmt> → <assignment Stmt> | <iterative Stmt> | <conditional Stmt> |
<io Stmt> | <funCall Stmt>

16) <returnStmt> → TK_RETURN <optionalReturn> TK_SEM

17) <parameter_list> → <dataType> TK_ID <remaining_list>

18) <dataType> → <primitive Datatype> | <Constructed Datatype>

19) <primitive Datatype> → TK_INT | TK_REAL

20) <constructed Datatype> → TK_RECORD TK_RUID | TK_UNION
TK_RUID

21) <remaining_list> → TK_COMMA <parameter_list> | ε

22) <A> → TK_RECORD | TK_UNION

23) <recordDefinitions> → <recordDefinition><recordDefinitions> | ε

24) <recordDefinition> → TK_RECORD TK_RUID <recordDefinitions>
<field Definitions> TK_ENDRECORD

25) <fieldDefinitions> → <fieldDefinition> <fieldDefinition> <moreFields>

26) <fieldDefinition> → TK_TYPE <dataType> TK_COLON TK_FIELDID
TK_SEM

27) <moreFields> → <fieldDefinition> <moreFields> | ε

28) <global_or_not> → TK_COLON TK_GLOBAL | ε

29) <assignmentStmt> → <singleOrRecId> TK_ASSIGNOP <arithmeticExpression>
TK_SEM

30) <singleOrRecId> → TK_ID <singleOrRecId'>

31) <singleOrRecId'> → TK_DOT <FieldIds> | ϵ

32) <FieldIds> → TK_FIELDID <FieldIds'>

33) <FieldIds'> → TK_DOT <FieldIds> | ϵ

34) <funCallStmt> → <outputParameters> TK_CALL TK_FUNID
TK_WITH TK_PARAMETERS <inputParameters>

35) <outputParameters> → TK_SQL <idList> TK_SQR TK_ASSIGNOP | ϵ

36) <inputParameters> → TK_SQL <idList> TK_SQR

37) <iterativeStmt> → TK_WHILE TK_OP <booleanExpression> TK_CL
<stmt> <otherStmts> TK_ENDWHILE

38) <conditionalStmt> → TK_IF TK_OP <booleanExpression> TK_CL
TK_THEN <stmt> <otherStmts> <conditionalStmt'>

39) <conditionalStmt'> → TK_ENDIF | TK_ELSE <otherStmts> TK_ENDIF

40) <ioStmt> → TK_READ TK_OP TK_CL TK_SEM | TK_WRITE
TK_OP TK_CL TK_SEM

41) <arithmeticExpression> → <expwithoutMINUS> <arithmeticExpression'>

42) <arithmeticExpression'> → TK_MINUS <expwithoutMINUS>
<arithmeticExpression'> | ϵ

43) <expwithoutMINUS> → <expwithoutPLUS> <expwithoutMINUS'>

44) <expwithoutMINUS'> → TK_PLUS <expwithoutPLUS>
<expwithoutMINUS'> | ϵ

45) <expwithoutPLUS> → <expwithout MUL> <expwithoutPLUS'>

46) <expwithout PLUS'> → TK_MUL <expwithout MUL>
<expwithout PLUS'> | ε

47) <expwithout MUL> → <expwithoutDIV> <expwithoutMUL'>

48) <expwithout MUL'> → TK_DIV <expwithoutDIV>
<exp without MUL'> | ε

49) <expwithoutDIV> → TK_OP <arithmetic Expression> TK_CL |
<var>

50) <boolean Expression> → TK_OP <booleanExpression> TK_CL
<logical op> TK_OP <boolean Expression>
TK_CL | <var> <relational op> <var> |
TK_NOT <boolean Expression>

51) <var> → TK_ID | TK_NUM | TK_RNUM

52) <logical op> → TK_AND | TK_OR

53) <relational op> → TK_LT | TK_LE | TK_EQ | TK_GT | TK_GE |
TK_NE

54) <optional Return> → TK_SQL <idlist> TK_SQR | ε

55) <idlist> → TK_ID <more_ids>

56) <more_ids> → TK_COMMA <idlist> | ε

First Sets

1) First(program) = {TK_FUNID, TK_MAIN}

2) First(otherFunctions) = {TK_FUNID, ∈}

3) First(mainFunction) = {TK_MAIN}

4) First(function) = {TK_FUNID}

5) First(stmts) = {TK_DEFINETYPE, TK_RECORD, TK_UNION, TK_TYPE, TK_ID, TK_WHILE, TK_IF, TK_SQL, TK_CALL, TK_READ, TK_WRITE, TK_SQL, TK_CALL, TK_RETURN}

6) First(input_par) = {TK_INPUT}

7) First(output_par) = {TK_OUTPUT, ∈}

8) First(definetypeStmts) = {TK_DEFINETYPE, ∈}

9) First(definetypestmt) = {TK_DEFINETYPE}

10) First(typeDefinitions) = {TK_RECORD, TK_UNION, ∈}

11) First(typeDefinition) = {TK_RECORD, TK_UNION}

12) First(declarations) = {TK_TYPE, ∈}

13) First(declaration) = {TK_TYPE}

14) First(otherStmts) = {TK_ID, TK_WHILE, TK_IF, TK_READ, TK_WRITE, TK_SQL, TK_CALL, ∈}

15) First(stmt) = {TK_ID, TK_WHILE, TK_IF, TK_READ, TK_WRITE, TK_SQL, TK_CALL}

16) First(returnStmt) = {TK_RETURN}

17) First(parameter_list) = {TK_INT, TK_REAL, TK_RECORD, TK_UNION}

18) First(data Type) = {TK_INT, TK_REAL, TK_RECORD, TK_UNION}

19) First(primitive Data type) = { TK_INT, TK_REAL}

20) First(constructed Data type) = { TK_RECORD, TK_UNION}

21) First(remaining_list) = { TK_COMMA, ∈}

22) First(A) = { TK_RECORD, TK_UNION}

23) First(recordDefinitions) = { TK_RECORD, ∈}

24) First(recordDefinition) = { TK_RECORD}

25) First(fieldDefinitions) = { TK_TYPE}

26) First(field Definition) = { TK_TYPE}

27) First(moreFields) = { TK_TYPE, ∈}

28) First(global_or_not) = { TK_COLON, ∈}

29) First(assignment-Stmt) = { TK_ID}

30) First(single Or Rec Id) = { TK_ID}

31) First(single Or RecId') = { TK_DOT, ∈}

32) First(FieldIds) = { TK_FIELDID}

33) First(FieldIds') = { TK_DOT, ∈}

34) First(funCallStmt) = { TK_SQL, TK_CALL}

35) First(outputParameters) = { TK_SQL, ∈}

36) First(inputParameters) = { TK_SQL}

37) First(iterative Stmt) = { TK_WHILE}

38) First(conditional Stmt) = { TK_IF}

39) First(conditionalStmt') = { TK_ENDIF, TK_ELSE}

40) First(ioStmt) = { TK_READ, TK_WRITE}

41) First(arithmetic Expression) = { TK-OP, TK-ID, TK-NUM, TK-RNUM }

42) First(arithmeticExpression') = { TK-MINUS, ε }

43) First(expwithoutMINUS) = { TK-OP, TK-ID, TK-NUM, TK-RNUM }

44) First(expwithoutMINUS') = { TK-PLUS, ε }

45) First(expwithoutPLUS) = { TK-OP, TK-ID, TK-NUM, TK-RNUM }

46) First(expwithout PLUS') = { TK_MUL, ε }

47) First (expwithoutMUL) = { TK-OP, TK-ID, TK-NUM, TK-RNUM }

48) First(expwithoutMUL') = { TK-DIV, ε }

49) First (exp without DIV) = { TK-OP , TK-ID, TK- NUM, TK-RNUM }

50) First(booleanExpression) = { TK-OP, TK-NOT , TK-ID, TK-NUM ,
TK-RNUM }

51) First (var) = { TK-ID, TK- NUM, TK-RNUM }

52) First (logical op) = { TK-AND, TK-OR }

53) First (relational op) = { TK-LT, TK-LE, TK-EQ , TK-GT,
TK-GE , TK-NE }

54) First(optionalReturn) = { TK-SQL, ε }

55) First (idList) = { TK-ID }

56) First (more_ids) = { TK_COMMA, ε }

Follow Sets

1) Follow (program) = { $ }

2) Follow (other Functions) = { TK_MAIN }

3) Follow (mainFunction) = { $ }

4) Follow (function) = { TK_FUNID , TK_MAIN }

5) Follow (stmts) = { TK_END }

6) Follow (input_par) = { TK_OUTPUT , TK_SEM }

7) Follow (output_par) = { TK_SEM }

8) Follow (definetypeStmts) = { TK_RECORD , TK_UNION, TK_TYPE,
TK_ID, TK_WHILE, TK_IF, TK_READ,
TK_WRITE, TK_SQL , TK_CALL ,
TK_RETURN }

9) Follow (define type stmt) = { TK_RECORD, TK_UNION , TK_TYPE,
TK_ID, TK_WHILE, TK_IF, TK_READ,
TK_WRITE, TK_SQL, TK_CALL,
TK_RETURN, TK_DEFINETYPE }

10) Follow(typeDefinitions) = { TK_TYPE, TK_ID, TK_WHILE, TK_IF,
TK_READ, TK_WRITE, TK_SQL, TK_CALL,
TK_RETURN }

11) Follow (typeDefinition) = { TK_RECORD, TK_UNION, TK_TYPE, TK_ID,
TK_WHILE, TK_IF, TK_READ, TK_WRITE,
TK_SQL, TK_CALL , TK_RETURN }

12) Follow (declarations) = { TK_ID, TK_WHILE , TK_IF, TK_READ,
TK_WRITE, TK_SQL, TK_CALL , TK_RETURN }

13) Follow(declaration) = { TK_TYPE, TK_ID, TK_WHILE , TK_IF, TK_READ,
TK_WRITE, TK_SQL, TK_CALL, TK_RETURN }

14) Follow (otherStmts) = { TK_RETURN, TK_ENDIF, TK_ENDWHILE, TK_ELSE }

15) Follow (stmt) = { TK_ID, TK_WHILE, TK_IF, TK_READ, TK_WRITE,
TK_SQL, TK_CALL, TK_RETURN, TK_ENDIF,
TK_ENDWHILE, TK_ELSE }

16) Follow (returnStmt) = { TK_END }

17) Follow (parameter_list) = { TK_SQR }

18) Follow (dataType) = { TK_ID, TK_COLON }

19) Follow (primitive Datatype) = { TK_ID, TK_COLON }

20) Follow (constructed Datatype) = { TK_ID, TK_COLON }

21) Follow (remaining_list) = { TK_SQR }

22) Follow (A) = { TK_RUID }

23) Follow (record Definitions) = { TK_TYPE }

24) Follow (recordDefinition) = { TK_RECORD, TK_TYPE }

25) Follow (fieldDefinitions) = (TK_ENDRECORD, TK_ENDUNION)

26) Follow (fieldDefinition) = { TK_TYPE, TK_ENDRECORD, TK_ENDUNION }

27) Follow (moreFields) = { TK_ENDRECORD, TK_ENDUNION }

28) Follow(global_or_not) = { TK_SEM }

29) Follow(assignmentStmt) = { TK_ID, TK_WHILE, TK_IF, TK_READ,
TK_WRITE, TK_SQL, TK_CALL, TK_RETURN,
TK_ENDIF, TK_ENDWHILE, TK_ELSE }

30) Follow(singleOrRecId) = { TK_ASSIGNOP }

31) Follow (singleOrRecId') = { TK_ASSIGNOP }

32) Follow (fieldIds) = { TK_ASSIGNOP }

33) Follow (fieldIds') = { TK_ASSIGNOP }

34) Follow(funCallStmt) = { TK-ID, TK-WHILE, TK-IF, TK-READ,
TK-WRITE, TK-SQL, TK-CALL, TK-RETURN,
TK-ENDIF, TK-ENDWHILE, TK-ELSE}

35) Follow(outputParameters) = { TK-CALL}

36) Follow (inputParameters) = { TK-ID, TK-WHILE, TK-IF,
TK-READ, TK-WRITE, TK-SQL,
TK-CALL, TK-RETURN, TK-ENDIF,
TK-ENDWHILE, TK-ELSE}

37) Follow(iterative Stmt) = { TK_ID, TK_WHILE, TK_IF, TK_READ,
TK_WRITE, TK_SQL, TK_CALL, TK_RETURN,
TK_ENDIF, TK_ENDWHILE, TK_ELSE}

38) Follow (conditionalStmt) = { TK_ID, TK_WHILE, TK_IF, TK_READ,
TK_WRITE, TK_SQL, TK_CALL, TK_RETURN,
TK_ENDIF, TK_ENDWHILE, TK_ELSE}

39) Follow(conditional Stmt') = { TK_ID, TK_WHILE, TK_IF, TK_READ,
TK_WRITE, TK_SQL, TK_CALL, TK_RETURN,
TK_ENDIF, TK_ENDWHILE, TK_ELSE}

40) Follow (ioStmt) = { TK_ID, TK_WHILE, TK_IF, TK_READ,
TK_WRITE, TK_SQL, TK_CALL, TK_RETURN,
TK_ENDIF, TK_ENDWHILE, TK_ELSE}

41) Follow (arithmetic Expression) = { TK-SEM, TK-CL}

42) Follow(arithmetic Expression') = { TK-SEM, TK-CL}

43) Follow (expwithout MINUS) = { TK_MINUS, TK-SEM, TK-CL}

44) Follow (expwithoutMINUS') = { TK-MINUS, TK-SEM, TK-CL}

45) Follow (expwithout PLUS) = { TK_PLUS, TK_MINUS, TK_SEM, TK_CL}

46) Follow(expwithout PLUS') = { TK_PLUS, TK_MINUS, TK_SEM, TK_CL}

47) Follow (expwithout MUL) = { TK_MUL, TK_PLUS, TK_MINUS,
TK_SEM, TK_CL }

48) Follow (expwithout MUL') = { TK_MUL, TK_PLUS, TK_MINUS,
TK_SEM, TK_CL }

49) Follow (expwithout DIV) = { TK_DIV, TK_MUL, TK_PLUS, TK_MINUS,
TK_SEM, TK_CL}

50) Follow (booleanExpression) = { TK_CL}

51) Follow (var) = { TK_LT, TK_LE, TK_EQ, TK_GT, TK_GE,
TK_NE, TK_CL, TK_DIV, TK_MUL, TK_PLUS,
TK_MINUS, TK_SEM}

52) Follow(logical op) = { TK_OP}

53) Follow (relational op) = { TK_ID , TK_NUM, TK_RNUM }

54) Follow (optional Return) = { TK_SEM}

55) Follow (idList) = { TK_SQR}

56) Follow (more_ids) = { TK_SQR}

Assumptions

1) All definetype statements are at the starting of functions

2) If there exists a nested record, it is defined at the starting of parent record