Original production

R1. <Rat23F> ::= <Opt Function Definitions> # <Opt Declaration List> <Statement List> #

R2. <Opt Function Definitions> ::= <Function Definitions> | <Empty>

R3. <Function Definitions> ::= <Function> | <Function> <Function Definitions>

R4. <Function> ::= function <Identifier> ( <Opt Parameter List> ) <Opt Declaration List> <Body>

R5. <Opt Parameter List> ::= <Parameter List> | <Empty>

R6. <Parameter List> ::= <Parameter> | <Parameter> , <Parameter List>

R7. <Parameter> ::= <IDs > <Qualifier>

R8. <Qualifier> ::= integer | bool | real

R9. <Body> ::= { < Statement List> }

R10. <Opt Declaration List> ::= <Declaration List> | <Empty>

R11. <Declaration List> := <Declaration> ; | <Declaration> ; <Declaration List>

R12. <Declaration> ::= <Qualifier > <IDs>

R13. <IDs> ::= <Identifier> | <Identifier>, <IDs>

R14. <Statement List> ::= <Statement> | <Statement> <Statement List>

R15. <Statement> ::= <Compound> | <Assign> | <If> | <Return> | <Print> | <Scan> | <While>

R16. <Compound> ::= { <Statement List> }

R17. <Assign> ::= <Identifier> = <Expression> ;

R18. <If> ::= if ( <Condition> ) <Statement> endif | if ( <Condition> ) <Statement> else <Statement> endif

R19. <Return> ::= ret ; | ret <Expression> ;

R20. <Print> ::= put ( <Expression>);

R21. <Scan> ::= get ( <IDs> );

R22. <While> ::= while ( <Condition> ) <Statement>

R23. <Condition> ::= <Expression> <Relop> <Expression>

R24. <Relop> ::= == | != | > | < | <= | =>

R25. <Expression> ::= <Expression> + <Term> | <Expression> - <Term> | <Term>

R26. <Term> ::= <Term> \* <Factor> | <Term> / <Factor> | <Factor>

R27. <Factor> ::= - <Primary> | <Primary>

R28. <Primary> ::= <Identifier> | <Integer> | <Identifier> ( <IDs> ) | ( <Expression> ) | <Real> | true | false

R29. <Empty> ::= ε

Note: <Identifier>, <Integer>, <Real> are token types as defined in section (1) of the project document on Canvas.

Modified production with left recursion and backtracking removed.

R1. <Rat23F> ::= <Opt Function Definitions> # <Opt Declaration List> <Statement List> #

R2. <Opt Function Definitions> ::= <Function Definitions> | <Empty>

R3. <Function Definitions> ::= <Function> <Function Definitions Continued>

R3.5. <Function Definitions Continued> ::= <Empty> | <Function Definitions>

R4. <Function> ::= function <Identifier> ( <Opt Parameter List> ) <Opt Declaration List> <Body>

R5. <Opt Parameter List> ::= <Parameter List> | <Empty>

R6. <Parameter List> ::= <Parameter> <Parameter List Continued>

R6.5. <Parameter List Continued> ::= <Empty> | ,<Parameter List>

R7. <Parameter> ::= <IDs > <Qualifier>

R8. <Qualifier> ::= integer | bool | real

R9. <Body> ::= { < Statement List> }

R10. <Opt Declaration List> ::= <Declaration List> | <Empty>

R11. <Declaration List> ::= <Declaration> ; <Declaration List Continued>

R11.5. <Declaration List Continued> ::= <Empty> | <Declaration List>

R12. <Declaration> ::= <Qualifier > <IDs>

R13. <IDs> ::= <Identifier> <IDs Continued>

R13.5. <IDs Continued> ::= <Empty> | , <IDs>

R14. <Statement List> ::= <Statement> <Statement List Continued>

R14.5. <Statement List Continued> ::= <Empty> | <Statement List>

R15. <Statement> ::= <Compound> | <Assign> | <If> | <Return> | <Print> | <Scan> | <While>

R16. <Compound> ::= { <Statement List> }

R17. <Assign> ::= <Identifier> = <Expression> ;

R18. <If> ::= if ( <Condition> ) <Statement> <If Continued>

R18.5. <If Continued> ::= endif | else <Statement> endif

R19. <Return> ::= ret <Return Continued>

R19.5 <Return Continued> ::= ; | <Expression> ;

R20. <Print> ::= put ( <Expression>);

R21. <Scan> ::= get ( <IDs> );

R22. <While> ::= while ( <Condition> ) <Statement>

R23. <Condition> ::= <Expression> <Relop> <Expression>

R24. <Relop> ::= == | != | > | < | <= | =>

R25. <Expression> ::= <Expression> <Expression Continued> | <Term>

R25.5. <Expression Continued> ::= + <Term> | - <Term>

R26. <Term> ::= <Term> <Term Continued | <Factor>

R26.5. <Term Continued> ::= \* <Factor> | / <Factor>

R27. <Factor> ::= - <Primary> | <Primary>

R28. <Primary> ::= <Identifier> <Primary Continued> | <Integer> | ( <Expression> ) | <Real> | true | false

R28.5. <Primary Continued> ::= <Empty> | ( <IDs> )

R29. <Empty> ::= ε

Note: <Identifier>, <Integer>, <Real> are token types as defined in section (1) of the project document on Canvas.