CS323 Documentation

About 2 pages

1. Problem Statement

write a program that takes the output of the lexical analyzer and parses it, searching for errors in format

2. How to use your program

1. download zip folder on to desktop
2. unzip folder
3. open folder in editor
4. open terminal and run commands
5. “cd Desktop”
6. “cd CPSC323-project2-main/”
7. “clang++ -std=c++11 main.cpp lexer.cpp par.cpp -o main”
8. execute with ./main
9. enter the input file and the output will be printed in output.txt
10. add your own input files

3. Design of your program

top-down parser

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

<Function_Definitions> ::= <Function> <Function_Definitions_Prime>
<Function_Definitions_Prime> ::= <Function_Definitions> | <Empty>

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

<Parameter_List> ::= <Parameter> <Parameter_List_Prime>
<Parameter_List_Prime> ::= , <Parameter_List> | <Empty>

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

<Declaration_List> ::= <Declaration> ; <Declaration_List_Prime>
<Declaration_List_Prime> ::= <Declaration_List> | <Empty>

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

<IDs> ::= <Identifier> <IDs_Prime>
<IDs_Prime> ::= , <IDs> | <Empty>

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

<Statement_List> ::= <Statement> <Statement_List_Prime>
<Statement_List_Prime> ::= <Statement_List> | <Empty>

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

<If> ::= if (<Condition>) <Statement> <If_Prime>
<If_Prime> ::= fi | else <Statement> fi

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

<Return> ::= return <Return_Prime>
<Return_Prime> ::= ; | <Expression> ;

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

<Expression> ::= <Term> <Expression_Prime>

<Expression_Prime> ::= + <Term> <Expression_Prime> |
- <Term> <Expression_Prime> |
<Empty>

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

<Term> ::= <Factor> <Term_Prime>

<Term_Prime> ::= * <Factor> <Term_Prime> |
/ <Factor> <Term_Prime> |
<Empty>

4. Any Limitation

gets stuck on <RAT23S> and <Opt Function Definitions>

5. Any shortcomings

none