|  |  |
| --- | --- |
| Successful SA Code Block | Log Results |
| {}$ | Beginning Lexing Session... \*Stings Treated As CharList\*  LEXER --> | T\_OPENING\_BRACE [ { ] on line 2...  LEXER --> | T\_CLOSING\_BRACE [ } ] on line 2...  LEXER --> | T\_EOPS [ $ ] on line 2...  Lex Completed With 0 WARNING(S) and 0 ERROR(S)...  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Beginning Parsing Session...  PARSER --> | PASSED! Expecting [ T\_OPENING\_BRACE ] found [ T\_OPENING\_BRACE ] on line 2...  PARSER --> | PASSED! λ production on line 2...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_BRACE ] found [ T\_CLOSING\_BRACE ] on line 2...  PARSER --> | PASSED! Expecting [ T\_EOPS ] found [ T\_EOPS ] on line 2...  Parse Completed With 0 WARNING(S) and 0 ERROR(S)...  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Beginning Semantic Analysis Session...  Semantic Analysis Completed With 0 WARNING(S) and 0 ERROR(S)...  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| {  print("there is no spoon")  }$ | Beginning Lexing Session... \*Stings Treated As CharList\*  LEXER --> | T\_OPENING\_BRACE [ { ] on line 2...  LEXER --> | T\_PRINT [ print ] on line 3...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 3...  LEXER --> | T\_QUOTE [ " ] on line 3...  LEXER --> | T\_CHAR [ t ] on line 3...  LEXER --> | T\_CHAR [ h ] on line 3...  LEXER --> | T\_CHAR [ e ] on line 3...  LEXER --> | T\_CHAR [ r ] on line 3...  LEXER --> | T\_CHAR [ e ] on line 3...  LEXER --> | T\_CHAR [ ] on line 3...  LEXER --> | T\_CHAR [ i ] on line 3...  LEXER --> | T\_CHAR [ s ] on line 3...  LEXER --> | T\_CHAR [ ] on line 3...  LEXER --> | T\_CHAR [ n ] on line 3...  LEXER --> | T\_CHAR [ o ] on line 3...  LEXER --> | T\_CHAR [ ] on line 3...  LEXER --> | T\_CHAR [ s ] on line 3...  LEXER --> | T\_CHAR [ p ] on line 3...  LEXER --> | T\_CHAR [ o ] on line 3...  LEXER --> | T\_CHAR [ o ] on line 3...  LEXER --> | T\_CHAR [ n ] on line 3...  LEXER --> | T\_QUOTE [ " ] on line 3...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 3...  LEXER --> | T\_CLOSING\_BRACE [ } ] on line 4...  LEXER --> | T\_EOPS [ $ ] on line 4...  Lex Completed With 0 WARNING(S) and 0 ERROR(S)...  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Beginning Parsing Session...  PARSER --> | PASSED! Expecting [ T\_OPENING\_BRACE ] found [ T\_OPENING\_BRACE ] on line 2...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 3...  PARSER --> | PASSED! Expecting [ PrintStatement ] found [ T\_PRINT ] on line 3...  PARSER --> | PASSED! Expecting [ T\_PRINT ] found [ T\_PRINT ] on line 3...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ T\_OPENING\_PARENTHESIS ] on line 3...  PARSER --> | PASSED! Expecting [ Expr ] found [ StringExpr ] on line 3...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ t ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ h ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ e ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ r ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ e ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ i ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ s ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ n ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ o ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ s ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ p ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ o ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ o ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ n ] on line 3...  PARSER --> | PASSED! λ production on line 3...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ T\_CLOSING\_PARENTHESIS ] on line 3...  PARSER --> | PASSED! λ production on line 4...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_BRACE ] found [ T\_CLOSING\_BRACE ] on line 4...  PARSER --> | PASSED! Expecting [ T\_EOPS ] found [ T\_EOPS ] on line 4...  Parse Completed With 0 WARNING(S) and 0 ERROR(S)...  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Beginning Semantic Analysis Session...  Semantic Analysis Completed With 0 WARNING(S) and 0 ERROR(S)...  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| {  print((false == true))  print((true != true))  print((false != false))  print((false != true))  } | Beginning Lexing Session... \*Stings Treated As CharList\*  LEXER --> | T\_OPENING\_BRACE [ { ] on line 1...  LEXER --> | T\_PRINT [ print ] on line 2...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 2...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 2...  LEXER --> | T\_BOOLEAN\_VALUE [ false ] on line 2...  LEXER --> | T\_EQUALITY\_OP [ == ] on line 2...  LEXER --> | T\_BOOLEAN\_VALUE [ true ] on line 2...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 2...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 2...  LEXER --> | T\_PRINT [ print ] on line 3...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 3...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 3...  LEXER --> | T\_BOOLEAN\_VALUE [ true ] on line 3...  LEXER --> | T\_INEQUALITY\_OP [ != ] on line 3...  LEXER --> | T\_BOOLEAN\_VALUE [ true ] on line 3...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 3...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 3...  LEXER --> | T\_PRINT [ print ] on line 4...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 4...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 4...  LEXER --> | T\_BOOLEAN\_VALUE [ false ] on line 4...  LEXER --> | T\_INEQUALITY\_OP [ != ] on line 4...  LEXER --> | T\_BOOLEAN\_VALUE [ false ] on line 4...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 4...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 4...  LEXER --> | T\_PRINT [ print ] on line 5...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 5...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 5...  LEXER --> | T\_BOOLEAN\_VALUE [ false ] on line 5...  LEXER --> | T\_INEQUALITY\_OP [ != ] on line 5...  LEXER --> | T\_BOOLEAN\_VALUE [ true ] on line 5...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 5...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 5...  LEXER --> | T\_CLOSING\_BRACE [ } ] on line 6...  LEXER --> | T\_EOPS [ $ ] on line 6...  Lex Completed With 0 WARNING(S) and 0 ERROR(S)...  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Beginning Parsing Session...  PARSER --> | PASSED! Expecting [ T\_OPENING\_BRACE ] found [ T\_OPENING\_BRACE ] on line 1...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 2...  PARSER --> | PASSED! Expecting [ PrintStatement ] found [ T\_PRINT ] on line 2...  PARSER --> | PASSED! Expecting [ T\_PRINT ] found [ T\_PRINT ] on line 2...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ T\_OPENING\_PARENTHESIS ] on line 2...  PARSER --> | PASSED! Expecting [ Expr ] found [ BooleanExpr ] on line 2...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ ( ] on line 2...  PARSER --> | PASSED! Expecting [ Expr ] found [ BooleanExpr ] on line 2...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ T\_BOOLEAN\_VALUE ] on line 2...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ false ] on line 2...  PARSER --> | PASSED! Expecting [ T\_EQUALITY\_OP ] found [ == ] on line 2...  PARSER --> | PASSED! Expecting [ Expr ] found [ BooleanExpr ] on line 2...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ T\_BOOLEAN\_VALUE ] on line 2...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ true ] on line 2...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ ) ] on line 2...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ T\_CLOSING\_PARENTHESIS ] on line 2...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 3...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 3...  PARSER --> | PASSED! Expecting [ PrintStatement ] found [ T\_PRINT ] on line 3...  PARSER --> | PASSED! Expecting [ T\_PRINT ] found [ T\_PRINT ] on line 3...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ T\_OPENING\_PARENTHESIS ] on line 3...  PARSER --> | PASSED! Expecting [ Expr ] found [ BooleanExpr ] on line 3...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ ( ] on line 3...  PARSER --> | PASSED! Expecting [ Expr ] found [ BooleanExpr ] on line 3...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ T\_BOOLEAN\_VALUE ] on line 3...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ true ] on line 3...  PARSER --> | PASSED! Expecting [ T\_INEQUALITY\_OP ] found [ != ] on line 3...  PARSER --> | PASSED! Expecting [ Expr ] found [ BooleanExpr ] on line 3...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ T\_BOOLEAN\_VALUE ] on line 3...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ true ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ ) ] on line 3...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ T\_CLOSING\_PARENTHESIS ] on line 3...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 4...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 4...  PARSER --> | PASSED! Expecting [ PrintStatement ] found [ T\_PRINT ] on line 4...  PARSER --> | PASSED! Expecting [ T\_PRINT ] found [ T\_PRINT ] on line 4...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ T\_OPENING\_PARENTHESIS ] on line 4...  PARSER --> | PASSED! Expecting [ Expr ] found [ BooleanExpr ] on line 4...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ ( ] on line 4...  PARSER --> | PASSED! Expecting [ Expr ] found [ BooleanExpr ] on line 4...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ T\_BOOLEAN\_VALUE ] on line 4...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ false ] on line 4...  PARSER --> | PASSED! Expecting [ T\_INEQUALITY\_OP ] found [ != ] on line 4...  PARSER --> | PASSED! Expecting [ Expr ] found [ BooleanExpr ] on line 4...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ T\_BOOLEAN\_VALUE ] on line 4...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ false ] on line 4...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ ) ] on line 4...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ T\_CLOSING\_PARENTHESIS ] on line 4...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 5...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 5...  PARSER --> | PASSED! Expecting [ PrintStatement ] found [ T\_PRINT ] on line 5...  PARSER --> | PASSED! Expecting [ T\_PRINT ] found [ T\_PRINT ] on line 5...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ T\_OPENING\_PARENTHESIS ] on line 5...  PARSER --> | PASSED! Expecting [ Expr ] found [ BooleanExpr ] on line 5...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ ( ] on line 5...  PARSER --> | PASSED! Expecting [ Expr ] found [ BooleanExpr ] on line 5...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ T\_BOOLEAN\_VALUE ] on line 5...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ false ] on line 5...  PARSER --> | PASSED! Expecting [ T\_INEQUALITY\_OP ] found [ != ] on line 5...  PARSER --> | PASSED! Expecting [ Expr ] found [ BooleanExpr ] on line 5...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ T\_BOOLEAN\_VALUE ] on line 5...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ true ] on line 5...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ ) ] on line 5...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ T\_CLOSING\_PARENTHESIS ] on line 5...  PARSER --> | PASSED! λ production on line 6...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_BRACE ] found [ T\_CLOSING\_BRACE ] on line 6...  PARSER --> | PASSED! Expecting [ T\_EOPS ] found [ T\_EOPS ] on line 6...  Parse Completed With 0 WARNING(S) and 0 ERROR(S)...  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Beginning Semantic Analysis Session...  Semantic Analysis Completed With 0 WARNING(S) and 0 ERROR(S)...  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| {  int a  int b  a = 0  b = 0  while (a != 3) {  print(a)  while (b != 3) {  print(b)  b = 1 + b  if (b == 2) {  print("there is no spoon")  }  }  b = 0  a = 1 + a  }  }$ | Beginning Lexing Session... \*Stings Treated As CharList\*  LEXER --> | T\_OPENING\_BRACE [ { ] on line 2...  LEXER --> | T\_VARIABLE\_TYPE [ int ] on line 4...  LEXER --> | T\_ID [ a ] on line 4...  LEXER --> | T\_VARIABLE\_TYPE [ int ] on line 5...  LEXER --> | T\_ID [ b ] on line 5...  LEXER --> | T\_ID [ a ] on line 7...  LEXER --> | T\_ASSIGNMENT\_OP [ = ] on line 7...  LEXER --> | T\_DIGIT [ 0 ] on line 7...  LEXER --> | T\_ID [ b ] on line 8...  LEXER --> | T\_ASSIGNMENT\_OP [ = ] on line 8...  LEXER --> | T\_DIGIT [ 0 ] on line 8...  LEXER --> | T\_WHILE [ while ] on line 11...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 11...  LEXER --> | T\_ID [ a ] on line 11...  LEXER --> | T\_INEQUALITY\_OP [ != ] on line 11...  LEXER --> | T\_DIGIT [ 3 ] on line 11...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 11...  LEXER --> | T\_OPENING\_BRACE [ { ] on line 11...  LEXER --> | T\_PRINT [ print ] on line 12...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 12...  LEXER --> | T\_ID [ a ] on line 12...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 12...  LEXER --> | T\_WHILE [ while ] on line 13...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 13...  LEXER --> | T\_ID [ b ] on line 13...  LEXER --> | T\_INEQUALITY\_OP [ != ] on line 13...  LEXER --> | T\_DIGIT [ 3 ] on line 13...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 13...  LEXER --> | T\_OPENING\_BRACE [ { ] on line 13...  LEXER --> | T\_PRINT [ print ] on line 14...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 14...  LEXER --> | T\_ID [ b ] on line 14...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 14...  LEXER --> | T\_ID [ b ] on line 15...  LEXER --> | T\_ASSIGNMENT\_OP [ = ] on line 15...  LEXER --> | T\_DIGIT [ 1 ] on line 15...  LEXER --> | T\_ADDITION\_OP [ + ] on line 15...  LEXER --> | T\_ID [ b ] on line 15...  LEXER --> | T\_IF [ if ] on line 16...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 16...  LEXER --> | T\_ID [ b ] on line 16...  LEXER --> | T\_EQUALITY\_OP [ == ] on line 16...  LEXER --> | T\_DIGIT [ 2 ] on line 16...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 16...  LEXER --> | T\_OPENING\_BRACE [ { ] on line 16...  LEXER --> | T\_PRINT [ print ] on line 18...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 18...  LEXER --> | T\_QUOTE [ " ] on line 18...  LEXER --> | T\_CHAR [ t ] on line 18...  LEXER --> | T\_CHAR [ h ] on line 18...  LEXER --> | T\_CHAR [ e ] on line 18...  LEXER --> | T\_CHAR [ r ] on line 18...  LEXER --> | T\_CHAR [ e ] on line 18...  LEXER --> | T\_CHAR [ ] on line 18...  LEXER --> | T\_CHAR [ i ] on line 18...  LEXER --> | T\_CHAR [ s ] on line 18...  LEXER --> | T\_CHAR [ ] on line 18...  LEXER --> | T\_CHAR [ n ] on line 18...  LEXER --> | T\_CHAR [ o ] on line 18...  LEXER --> | T\_CHAR [ ] on line 18...  LEXER --> | T\_CHAR [ s ] on line 18...  LEXER --> | T\_CHAR [ p ] on line 18...  LEXER --> | T\_CHAR [ o ] on line 18...  LEXER --> | T\_CHAR [ o ] on line 18...  LEXER --> | T\_CHAR [ n ] on line 18...  LEXER --> | T\_QUOTE [ " ] on line 18...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 18...  LEXER --> | T\_CLOSING\_BRACE [ } ] on line 19...  LEXER --> | T\_CLOSING\_BRACE [ } ] on line 20...  LEXER --> | T\_ID [ b ] on line 22...  LEXER --> | T\_ASSIGNMENT\_OP [ = ] on line 22...  LEXER --> | T\_DIGIT [ 0 ] on line 22...  LEXER --> | T\_ID [ a ] on line 23...  LEXER --> | T\_ASSIGNMENT\_OP [ = ] on line 23...  LEXER --> | T\_DIGIT [ 1 ] on line 23...  LEXER --> | T\_ADDITION\_OP [ + ] on line 23...  LEXER --> | T\_ID [ a ] on line 23...  LEXER --> | T\_CLOSING\_BRACE [ } ] on line 24...  LEXER --> | T\_CLOSING\_BRACE [ } ] on line 25...  LEXER --> | T\_EOPS [ $ ] on line 25...  Lex Completed With 0 WARNING(S) and 0 ERROR(S)...  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Beginning Parsing Session...  PARSER --> | PASSED! Expecting [ T\_OPENING\_BRACE ] found [ T\_OPENING\_BRACE ] on line 2...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 4...  PARSER --> | PASSED! Expecting [ VarDecl ] found [ T\_VARIABLE\_TYPE ] on line 4...  PARSER --> | PASSED! Expecting [ T\_VARIABLE\_TYPE ] found [ T\_VARIABLE\_TYPE ] on line 4...  PARSER --> | PASSED! Expecting [ T\_VARIABLE\_TYPE ] found [ int ] on line 4...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 4...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 4...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 5...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 5...  PARSER --> | PASSED! Expecting [ VarDecl ] found [ T\_VARIABLE\_TYPE ] on line 5...  PARSER --> | PASSED! Expecting [ T\_VARIABLE\_TYPE ] found [ T\_VARIABLE\_TYPE ] on line 5...  PARSER --> | PASSED! Expecting [ T\_VARIABLE\_TYPE ] found [ int ] on line 5...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 5...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ b ] on line 5...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 7...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 7...  PARSER --> | PASSED! Expecting [ AssignmentStatement ] found [ T\_ID ] on line 7...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 7...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 7...  PARSER --> | PASSED! Expecting [ T\_ASSIGNMENT\_OP ] found [ = ] on line 7...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 7...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ IntExpr ] on line 7...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 0 ] on line 7...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 8...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 8...  PARSER --> | PASSED! Expecting [ AssignmentStatement ] found [ T\_ID ] on line 8...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 8...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ b ] on line 8...  PARSER --> | PASSED! Expecting [ T\_ASSIGNMENT\_OP ] found [ = ] on line 8...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 8...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ IntExpr ] on line 8...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 0 ] on line 8...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 11...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 11...  PARSER --> | PASSED! Expecting [ WhileStatement ] found [ T\_WHILE ] on line 11...  PARSER --> | PASSED! Expecting [ T\_WHILE ] found [ while ] on line 11...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ ( ] on line 11...  PARSER --> | PASSED! Expecting [ Expr ] found [ Id ] on line 11...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 11...  PARSER --> | PASSED! Expecting [ T\_INEQUALITY\_OP ] found [ != ] on line 11...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 11...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ IntExpr ] on line 11...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 3 ] on line 11...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ ) ] on line 11...  PARSER --> | PASSED! Expecting [ T\_OPENING\_BRACE ] found [ T\_OPENING\_BRACE ] on line 11...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 12...  PARSER --> | PASSED! Expecting [ PrintStatement ] found [ T\_PRINT ] on line 12...  PARSER --> | PASSED! Expecting [ T\_PRINT ] found [ T\_PRINT ] on line 12...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ T\_OPENING\_PARENTHESIS ] on line 12...  PARSER --> | PASSED! Expecting [ Expr ] found [ Id ] on line 12...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 12...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ T\_CLOSING\_PARENTHESIS ] on line 12...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 13...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 13...  PARSER --> | PASSED! Expecting [ WhileStatement ] found [ T\_WHILE ] on line 13...  PARSER --> | PASSED! Expecting [ T\_WHILE ] found [ while ] on line 13...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ ( ] on line 13...  PARSER --> | PASSED! Expecting [ Expr ] found [ Id ] on line 13...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ b ] on line 13...  PARSER --> | PASSED! Expecting [ T\_INEQUALITY\_OP ] found [ != ] on line 13...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 13...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ IntExpr ] on line 13...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 3 ] on line 13...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ ) ] on line 13...  PARSER --> | PASSED! Expecting [ T\_OPENING\_BRACE ] found [ T\_OPENING\_BRACE ] on line 13...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 14...  PARSER --> | PASSED! Expecting [ PrintStatement ] found [ T\_PRINT ] on line 14...  PARSER --> | PASSED! Expecting [ T\_PRINT ] found [ T\_PRINT ] on line 14...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ T\_OPENING\_PARENTHESIS ] on line 14...  PARSER --> | PASSED! Expecting [ Expr ] found [ Id ] on line 14...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ b ] on line 14...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ T\_CLOSING\_PARENTHESIS ] on line 14...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 15...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 15...  PARSER --> | PASSED! Expecting [ AssignmentStatement ] found [ T\_ID ] on line 15...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 15...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ b ] on line 15...  PARSER --> | PASSED! Expecting [ T\_ASSIGNMENT\_OP ] found [ = ] on line 15...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 15...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ T\_DIGIT ] on line 15...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 1 ] on line 15...  PARSER --> | PASSED! Expecting [ T\_ADDITION\_OP ] found [ + ] on line 15...  PARSER --> | PASSED! Expecting [ Expr ] found [ Id ] on line 15...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ b ] on line 15...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 16...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 16...  PARSER --> | PASSED! Expecting [ IfStatement ] found [ T\_IF ] on line 16...  PARSER --> | PASSED! Expecting [ T\_IF ] found [ if ] on line 16...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ ( ] on line 16...  PARSER --> | PASSED! Expecting [ Expr ] found [ Id ] on line 16...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ b ] on line 16...  PARSER --> | PASSED! Expecting [ T\_EQUALITY\_OP ] found [ == ] on line 16...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 16...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ IntExpr ] on line 16...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 2 ] on line 16...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ ) ] on line 16...  PARSER --> | PASSED! Expecting [ T\_OPENING\_BRACE ] found [ T\_OPENING\_BRACE ] on line 16...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 18...  PARSER --> | PASSED! Expecting [ PrintStatement ] found [ T\_PRINT ] on line 18...  PARSER --> | PASSED! Expecting [ T\_PRINT ] found [ T\_PRINT ] on line 18...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ T\_OPENING\_PARENTHESIS ] on line 18...  PARSER --> | PASSED! Expecting [ Expr ] found [ StringExpr ] on line 18...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ t ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ h ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ e ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ r ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ e ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ i ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ s ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ n ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ o ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ s ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ p ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ o ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ o ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ n ] on line 18...  PARSER --> | PASSED! λ production on line 18...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 18...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ T\_CLOSING\_PARENTHESIS ] on line 18...  PARSER --> | PASSED! λ production on line 19...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_BRACE ] found [ T\_CLOSING\_BRACE ] on line 19...  PARSER --> | PASSED! λ production on line 20...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_BRACE ] found [ T\_CLOSING\_BRACE ] on line 20...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 22...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 22...  PARSER --> | PASSED! Expecting [ AssignmentStatement ] found [ T\_ID ] on line 22...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 22...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ b ] on line 22...  PARSER --> | PASSED! Expecting [ T\_ASSIGNMENT\_OP ] found [ = ] on line 22...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 22...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ IntExpr ] on line 22...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 0 ] on line 22...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 23...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 23...  PARSER --> | PASSED! Expecting [ AssignmentStatement ] found [ T\_ID ] on line 23...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 23...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 23...  PARSER --> | PASSED! Expecting [ T\_ASSIGNMENT\_OP ] found [ = ] on line 23...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 23...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ T\_DIGIT ] on line 23...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 1 ] on line 23...  PARSER --> | PASSED! Expecting [ T\_ADDITION\_OP ] found [ + ] on line 23...  PARSER --> | PASSED! Expecting [ Expr ] found [ Id ] on line 23...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 23...  PARSER --> | PASSED! λ production on line 24...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_BRACE ] found [ T\_CLOSING\_BRACE ] on line 24...  PARSER --> | PASSED! λ production on line 25...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_BRACE ] found [ T\_CLOSING\_BRACE ] on line 25...  PARSER --> | PASSED! Expecting [ T\_EOPS ] found [ T\_EOPS ] on line 25...  Parse Completed With 0 WARNING(S) and 0 ERROR(S)...  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Beginning Semantic Analysis Session...  S.ANALYZE --> | PASSED! Variable [ a ] on line 7 has been declared...  S.ANALYZE --> | PASSED! Variable [ a ] on line 7 has [ int ] type...  S.ANALYZE --> | PASSED! Variable [ a ] on line 7 has been used...  S.ANALYZE --> | PASSED! Variable [ a ] on line 7 has type int and is assigned the correct type: int...  S.ANALYZE --> | PASSED! Variable [ b ] on line 8 has been declared...  S.ANALYZE --> | PASSED! Variable [ b ] on line 8 has [ int ] type...  S.ANALYZE --> | PASSED! Variable [ b ] on line 8 has been used...  S.ANALYZE --> | PASSED! Variable [ b ] on line 8 has type int and is assigned the correct type: int...  S.ANALYZE --> | PASSED! Variable [ a ] on line 11 has been declared...  S.ANALYZE --> | PASSED! Variable [ a ] on line 11 has been used...  S.ANALYZE --> | PASSED! Variable [ a ] on line 11 has [ int ] type...  S.ANALYZE --> | PASSED! Variable [ a ] on line 12 has been declared...  S.ANALYZE --> | PASSED! Variable [ a ] on line 12 has been used...  S.ANALYZE --> | PASSED! Variable [ a ] on line 12 has [ int ] type...  S.ANALYZE --> | PASSED! Variable [ b ] on line 13 has been declared...  S.ANALYZE --> | PASSED! Variable [ b ] on line 13 has been used...  S.ANALYZE --> | PASSED! Variable [ b ] on line 13 has [ int ] type...  S.ANALYZE --> | PASSED! Variable [ b ] on line 14 has been declared...  S.ANALYZE --> | PASSED! Variable [ b ] on line 14 has been used...  S.ANALYZE --> | PASSED! Variable [ b ] on line 14 has [ int ] type...  S.ANALYZE --> | PASSED! Variable [ b ] on line 15 has been declared...  S.ANALYZE --> | PASSED! Variable [ b ] on line 15 has [ int ] type...  S.ANALYZE --> | PASSED! Variable [ b ] on line 15 has been used...  S.ANALYZE --> | PASSED! Variable [ b ] on line 15 has been declared...  S.ANALYZE --> | PASSED! Variable [ b ] on line 15 has been used...  S.ANALYZE --> | PASSED! Variable [ b ] on line 15 has [ int ] type...  S.ANALYZE --> | PASSED! Variable [ b ] on line 15 has type int and is assigned the correct type: int...  S.ANALYZE --> | PASSED! Variable [ b ] on line 16 has been declared...  S.ANALYZE --> | PASSED! Variable [ b ] on line 16 has been used...  S.ANALYZE --> | PASSED! Variable [ b ] on line 16 has [ int ] type...  S.ANALYZE --> | PASSED! Variable [ b ] on line 22 has been declared...  S.ANALYZE --> | PASSED! Variable [ b ] on line 22 has [ int ] type...  S.ANALYZE --> | PASSED! Variable [ b ] on line 22 has been used...  S.ANALYZE --> | PASSED! Variable [ b ] on line 22 has type int and is assigned the correct type: int...  S.ANALYZE --> | PASSED! Variable [ a ] on line 23 has been declared...  S.ANALYZE --> | PASSED! Variable [ a ] on line 23 has [ int ] type...  S.ANALYZE --> | PASSED! Variable [ a ] on line 23 has been used...  S.ANALYZE --> | PASSED! Variable [ a ] on line 23 has been declared...  S.ANALYZE --> | PASSED! Variable [ a ] on line 23 has been used...  S.ANALYZE --> | PASSED! Variable [ a ] on line 23 has [ int ] type...  S.ANALYZE --> | PASSED! Variable [ a ] on line 23 has type int and is assigned the correct type: int...  Semantic Analysis Completed With 0 WARNING(S) and 0 ERROR(S)...  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

|  |  |
| --- | --- |
| Failed Lex Code Block | Log Results |
| {  string a  a = true  }$ | Beginning Lexing Session... \*Stings Treated As CharList\*  LEXER --> | T\_OPENING\_BRACE [ { ] on line 4...  LEXER --> | T\_VARIABLE\_TYPE [ string ] on line 5...  LEXER --> | T\_ID [ a ] on line 5...  LEXER --> | T\_ID [ a ] on line 6...  LEXER --> | T\_ASSIGNMENT\_OP [ = ] on line 6...  LEXER --> | T\_BOOLEAN\_VALUE [ true ] on line 6...  LEXER --> | T\_CLOSING\_BRACE [ } ] on line 7...  LEXER --> | T\_EOPS [ $ ] on line 7...  Lex Completed With 0 WARNING(S) and 0 ERROR(S)...  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Beginning Parsing Session...  PARSER --> | PASSED! Expecting [ T\_OPENING\_BRACE ] found [ T\_OPENING\_BRACE ] on line 4...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 5...  PARSER --> | PASSED! Expecting [ VarDecl ] found [ T\_VARIABLE\_TYPE ] on line 5...  PARSER --> | PASSED! Expecting [ T\_VARIABLE\_TYPE ] found [ T\_VARIABLE\_TYPE ] on line 5...  PARSER --> | PASSED! Expecting [ T\_VARIABLE\_TYPE ] found [ string ] on line 5...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 5...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 5...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 6...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 6...  PARSER --> | PASSED! Expecting [ AssignmentStatement ] found [ T\_ID ] on line 6...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 6...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 6...  PARSER --> | PASSED! Expecting [ T\_ASSIGNMENT\_OP ] found [ = ] on line 6...  PARSER --> | PASSED! Expecting [ Expr ] found [ BooleanExpr ] on line 6...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ T\_BOOLEAN\_VALUE ] on line 6...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ true ] on line 6...  PARSER --> | PASSED! λ production on line 7...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_BRACE ] found [ T\_CLOSING\_BRACE ] on line 7...  PARSER --> | PASSED! Expecting [ T\_EOPS ] found [ T\_EOPS ] on line 7...  Parse Completed With 0 WARNING(S) and 0 ERROR(S)...  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Beginning Semantic Analysis Session...  S.ANALYZE --> | PASSED! Variable [ a ] on line 6 has been declared...  S.ANALYZE --> | PASSED! Variable [ a ] on line 6 has [ string ] type...  S.ANALYZE --> | PASSED! Variable [ a ] on line 6 has been used...  S.ANALYZE --> | ERROR! Variable [ a ] on line 6 has type string and is assigned the wrong type [ boolean ]...  Semantic Analysis Failed With 0 WARNING(S) and 1 ERROR(S)... |
| {  int a  a = 1  if("a" == 3) {  a = 2  }  if(a != 1) {  a = 3  }  if(a == 1) {  a = 3  }  }$ | Beginning Lexing Session... \*Stings Treated As CharList\*  LEXER --> | T\_OPENING\_BRACE [ { ] on line 4...  LEXER --> | T\_VARIABLE\_TYPE [ int ] on line 5...  LEXER --> | T\_ID [ a ] on line 5...  LEXER --> | T\_ID [ a ] on line 6...  LEXER --> | T\_ASSIGNMENT\_OP [ = ] on line 6...  LEXER --> | T\_DIGIT [ 1 ] on line 6...  LEXER --> | T\_IF [ if ] on line 8...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 8...  LEXER --> | T\_QUOTE [ " ] on line 8...  LEXER --> | T\_CHAR [ a ] on line 8...  LEXER --> | T\_QUOTE [ " ] on line 8...  LEXER --> | T\_EQUALITY\_OP [ == ] on line 8...  LEXER --> | T\_DIGIT [ 3 ] on line 8...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 8...  LEXER --> | T\_OPENING\_BRACE [ { ] on line 8...  LEXER --> | T\_ID [ a ] on line 9...  LEXER --> | T\_ASSIGNMENT\_OP [ = ] on line 9...  LEXER --> | T\_DIGIT [ 2 ] on line 9...  LEXER --> | T\_CLOSING\_BRACE [ } ] on line 10...  LEXER --> | T\_IF [ if ] on line 12...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 12...  LEXER --> | T\_ID [ a ] on line 12...  LEXER --> | T\_INEQUALITY\_OP [ != ] on line 12...  LEXER --> | T\_DIGIT [ 1 ] on line 12...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 12...  LEXER --> | T\_OPENING\_BRACE [ { ] on line 12...  LEXER --> | T\_ID [ a ] on line 13...  LEXER --> | T\_ASSIGNMENT\_OP [ = ] on line 13...  LEXER --> | T\_DIGIT [ 3 ] on line 13...  LEXER --> | T\_CLOSING\_BRACE [ } ] on line 14...  LEXER --> | T\_IF [ if ] on line 16...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 16...  LEXER --> | T\_ID [ a ] on line 16...  LEXER --> | T\_EQUALITY\_OP [ == ] on line 16...  LEXER --> | T\_DIGIT [ 1 ] on line 16...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 16...  LEXER --> | T\_OPENING\_BRACE [ { ] on line 16...  LEXER --> | T\_ID [ a ] on line 17...  LEXER --> | T\_ASSIGNMENT\_OP [ = ] on line 17...  LEXER --> | T\_DIGIT [ 3 ] on line 17...  LEXER --> | T\_CLOSING\_BRACE [ } ] on line 18...  LEXER --> | T\_CLOSING\_BRACE [ } ] on line 19...  LEXER --> | T\_EOPS [ $ ] on line 19...  Lex Completed With 0 WARNING(S) and 0 ERROR(S)...  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Beginning Parsing Session...  PARSER --> | PASSED! Expecting [ T\_OPENING\_BRACE ] found [ T\_OPENING\_BRACE ] on line 4...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 5...  PARSER --> | PASSED! Expecting [ VarDecl ] found [ T\_VARIABLE\_TYPE ] on line 5...  PARSER --> | PASSED! Expecting [ T\_VARIABLE\_TYPE ] found [ T\_VARIABLE\_TYPE ] on line 5...  PARSER --> | PASSED! Expecting [ T\_VARIABLE\_TYPE ] found [ int ] on line 5...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 5...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 5...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 6...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 6...  PARSER --> | PASSED! Expecting [ AssignmentStatement ] found [ T\_ID ] on line 6...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 6...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 6...  PARSER --> | PASSED! Expecting [ T\_ASSIGNMENT\_OP ] found [ = ] on line 6...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 6...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ IntExpr ] on line 6...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 1 ] on line 6...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 8...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 8...  PARSER --> | PASSED! Expecting [ IfStatement ] found [ T\_IF ] on line 8...  PARSER --> | PASSED! Expecting [ T\_IF ] found [ if ] on line 8...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ ( ] on line 8...  PARSER --> | PASSED! Expecting [ Expr ] found [ StringExpr ] on line 8...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 8...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 8...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ a ] on line 8...  PARSER --> | PASSED! λ production on line 8...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 8...  PARSER --> | PASSED! Expecting [ T\_EQUALITY\_OP ] found [ == ] on line 8...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 8...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ IntExpr ] on line 8...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 3 ] on line 8...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ ) ] on line 8...  PARSER --> | PASSED! Expecting [ T\_OPENING\_BRACE ] found [ T\_OPENING\_BRACE ] on line 8...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 9...  PARSER --> | PASSED! Expecting [ AssignmentStatement ] found [ T\_ID ] on line 9...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 9...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 9...  PARSER --> | PASSED! Expecting [ T\_ASSIGNMENT\_OP ] found [ = ] on line 9...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 9...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ IntExpr ] on line 9...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 2 ] on line 9...  PARSER --> | PASSED! λ production on line 10...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_BRACE ] found [ T\_CLOSING\_BRACE ] on line 10...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 12...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 12...  PARSER --> | PASSED! Expecting [ IfStatement ] found [ T\_IF ] on line 12...  PARSER --> | PASSED! Expecting [ T\_IF ] found [ if ] on line 12...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ ( ] on line 12...  PARSER --> | PASSED! Expecting [ Expr ] found [ Id ] on line 12...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 12...  PARSER --> | PASSED! Expecting [ T\_INEQUALITY\_OP ] found [ != ] on line 12...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 12...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ IntExpr ] on line 12...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 1 ] on line 12...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ ) ] on line 12...  PARSER --> | PASSED! Expecting [ T\_OPENING\_BRACE ] found [ T\_OPENING\_BRACE ] on line 12...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 13...  PARSER --> | PASSED! Expecting [ AssignmentStatement ] found [ T\_ID ] on line 13...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 13...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 13...  PARSER --> | PASSED! Expecting [ T\_ASSIGNMENT\_OP ] found [ = ] on line 13...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 13...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ IntExpr ] on line 13...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 3 ] on line 13...  PARSER --> | PASSED! λ production on line 14...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_BRACE ] found [ T\_CLOSING\_BRACE ] on line 14...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 16...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 16...  PARSER --> | PASSED! Expecting [ IfStatement ] found [ T\_IF ] on line 16...  PARSER --> | PASSED! Expecting [ T\_IF ] found [ if ] on line 16...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ ( ] on line 16...  PARSER --> | PASSED! Expecting [ Expr ] found [ Id ] on line 16...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 16...  PARSER --> | PASSED! Expecting [ T\_EQUALITY\_OP ] found [ == ] on line 16...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 16...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ IntExpr ] on line 16...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 1 ] on line 16...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ ) ] on line 16...  PARSER --> | PASSED! Expecting [ T\_OPENING\_BRACE ] found [ T\_OPENING\_BRACE ] on line 16...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 17...  PARSER --> | PASSED! Expecting [ AssignmentStatement ] found [ T\_ID ] on line 17...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 17...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 17...  PARSER --> | PASSED! Expecting [ T\_ASSIGNMENT\_OP ] found [ = ] on line 17...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 17...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ IntExpr ] on line 17...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 3 ] on line 17...  PARSER --> | PASSED! λ production on line 18...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_BRACE ] found [ T\_CLOSING\_BRACE ] on line 18...  PARSER --> | PASSED! λ production on line 19...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_BRACE ] found [ T\_CLOSING\_BRACE ] on line 19...  PARSER --> | PASSED! Expecting [ T\_EOPS ] found [ T\_EOPS ] on line 19...  Parse Completed With 0 WARNING(S) and 0 ERROR(S)...  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Beginning Semantic Analysis Session...  S.ANALYZE --> | PASSED! Variable [ a ] on line 6 has been declared...  S.ANALYZE --> | PASSED! Variable [ a ] on line 6 has [ int ] type...  S.ANALYZE --> | PASSED! Variable [ a ] on line 6 has been used...  S.ANALYZE --> | PASSED! Variable [ a ] on line 6 has type int and is assigned the correct type: int...  S.ANALYZE --> | ERROR! Expr on line 8 has type [ string ] and is compared to the wrong type [ int ]...  Semantic Analysis Failed With 0 WARNING(S) and 1 ERROR(S)... |
| {  int a  a = 4  a = 1 + "there"  }$ | Beginning Lexing Session... \*Stings Treated As CharList\*  LEXER --> | T\_OPENING\_BRACE [ { ] on line 4...  LEXER --> | T\_VARIABLE\_TYPE [ int ] on line 5...  LEXER --> | T\_ID [ a ] on line 5...  LEXER --> | T\_ID [ a ] on line 6...  LEXER --> | T\_ASSIGNMENT\_OP [ = ] on line 6...  LEXER --> | T\_DIGIT [ 4 ] on line 6...  LEXER --> | T\_ID [ a ] on line 7...  LEXER --> | T\_ASSIGNMENT\_OP [ = ] on line 7...  LEXER --> | T\_DIGIT [ 1 ] on line 7...  LEXER --> | T\_ADDITION\_OP [ + ] on line 7...  LEXER --> | T\_QUOTE [ " ] on line 7...  LEXER --> | T\_CHAR [ t ] on line 7...  LEXER --> | T\_CHAR [ h ] on line 7...  LEXER --> | T\_CHAR [ e ] on line 7...  LEXER --> | T\_CHAR [ r ] on line 7...  LEXER --> | T\_CHAR [ e ] on line 7...  LEXER --> | T\_QUOTE [ " ] on line 7...  LEXER --> | T\_CLOSING\_BRACE [ } ] on line 8...  LEXER --> | T\_EOPS [ $ ] on line 8...  Lex Completed With 0 WARNING(S) and 0 ERROR(S)...  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Beginning Parsing Session...  PARSER --> | PASSED! Expecting [ T\_OPENING\_BRACE ] found [ T\_OPENING\_BRACE ] on line 4...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 5...  PARSER --> | PASSED! Expecting [ VarDecl ] found [ T\_VARIABLE\_TYPE ] on line 5...  PARSER --> | PASSED! Expecting [ T\_VARIABLE\_TYPE ] found [ T\_VARIABLE\_TYPE ] on line 5...  PARSER --> | PASSED! Expecting [ T\_VARIABLE\_TYPE ] found [ int ] on line 5...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 5...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 5...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 6...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 6...  PARSER --> | PASSED! Expecting [ AssignmentStatement ] found [ T\_ID ] on line 6...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 6...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 6...  PARSER --> | PASSED! Expecting [ T\_ASSIGNMENT\_OP ] found [ = ] on line 6...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 6...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ IntExpr ] on line 6...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 4 ] on line 6...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 7...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 7...  PARSER --> | PASSED! Expecting [ AssignmentStatement ] found [ T\_ID ] on line 7...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 7...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 7...  PARSER --> | PASSED! Expecting [ T\_ASSIGNMENT\_OP ] found [ = ] on line 7...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 7...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ T\_DIGIT ] on line 7...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 1 ] on line 7...  PARSER --> | PASSED! Expecting [ T\_ADDITION\_OP ] found [ + ] on line 7...  PARSER --> | PASSED! Expecting [ Expr ] found [ StringExpr ] on line 7...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 7...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 7...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ t ] on line 7...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 7...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ h ] on line 7...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 7...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ e ] on line 7...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 7...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ r ] on line 7...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 7...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ e ] on line 7...  PARSER --> | PASSED! λ production on line 7...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 7...  PARSER --> | PASSED! λ production on line 8...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_BRACE ] found [ T\_CLOSING\_BRACE ] on line 8...  PARSER --> | PASSED! Expecting [ T\_EOPS ] found [ T\_EOPS ] on line 8...  Parse Completed With 0 WARNING(S) and 0 ERROR(S)...  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Beginning Semantic Analysis Session...  S.ANALYZE --> | PASSED! Variable [ a ] on line 6 has been declared...  S.ANALYZE --> | PASSED! Variable [ a ] on line 6 has [ int ] type...  S.ANALYZE --> | PASSED! Variable [ a ] on line 6 has been used...  S.ANALYZE --> | PASSED! Variable [ a ] on line 6 has type int and is assigned the correct type: int...  S.ANALYZE --> | PASSED! Variable [ a ] on line 7 has been declared...  S.ANALYZE --> | PASSED! Variable [ a ] on line 7 has [ int ] type...  S.ANALYZE --> | PASSED! Variable [ a ] on line 7 has been used...  S.ANALYZE --> | ERROR! Expr on line 7 has type [ int ] and is added the wrong type [ string ]...  Semantic Analysis Failed With 0 WARNING(S) and 1 ERROR(S)... |
| {  int a  a = 9  boolean a  }$ | Beginning Lexing Session... \*Stings Treated As CharList\*  LEXER --> | T\_OPENING\_BRACE [ { ] on line 4...  LEXER --> | T\_VARIABLE\_TYPE [ int ] on line 5...  LEXER --> | T\_ID [ a ] on line 5...  LEXER --> | T\_ID [ a ] on line 6...  LEXER --> | T\_ASSIGNMENT\_OP [ = ] on line 6...  LEXER --> | T\_DIGIT [ 9 ] on line 6...  LEXER --> | T\_VARIABLE\_TYPE [ boolean ] on line 7...  LEXER --> | T\_ID [ a ] on line 7...  LEXER --> | T\_CLOSING\_BRACE [ } ] on line 8...  LEXER --> | T\_EOPS [ $ ] on line 8...  Lex Completed With 0 WARNING(S) and 0 ERROR(S)...  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Beginning Parsing Session...  PARSER --> | PASSED! Expecting [ T\_OPENING\_BRACE ] found [ T\_OPENING\_BRACE ] on line 4...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 5...  PARSER --> | PASSED! Expecting [ VarDecl ] found [ T\_VARIABLE\_TYPE ] on line 5...  PARSER --> | PASSED! Expecting [ T\_VARIABLE\_TYPE ] found [ T\_VARIABLE\_TYPE ] on line 5...  PARSER --> | PASSED! Expecting [ T\_VARIABLE\_TYPE ] found [ int ] on line 5...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 5...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 5...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 6...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 6...  PARSER --> | PASSED! Expecting [ AssignmentStatement ] found [ T\_ID ] on line 6...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 6...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 6...  PARSER --> | PASSED! Expecting [ T\_ASSIGNMENT\_OP ] found [ = ] on line 6...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 6...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ IntExpr ] on line 6...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 9 ] on line 6...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 7...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 7...  PARSER --> | PASSED! Expecting [ VarDecl ] found [ T\_VARIABLE\_TYPE ] on line 7...  PARSER --> | PASSED! Expecting [ T\_VARIABLE\_TYPE ] found [ T\_VARIABLE\_TYPE ] on line 7...  PARSER --> | PASSED! Expecting [ T\_VARIABLE\_TYPE ] found [ boolean ] on line 7...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 7...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 7...  PARSER --> | PASSED! λ production on line 8...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_BRACE ] found [ T\_CLOSING\_BRACE ] on line 8...  PARSER --> | PASSED! Expecting [ T\_EOPS ] found [ T\_EOPS ] on line 8...  Parse Completed With 0 WARNING(S) and 0 ERROR(S)...  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Beginning Semantic Analysis Session...  S.ANALYZE --> | PASSED! Variable [ a ] on line 6 has been declared...  S.ANALYZE --> | PASSED! Variable [ a ] on line 6 has [ int ] type...  S.ANALYZE --> | PASSED! Variable [ a ] on line 6 has been used...  S.ANALYZE --> | PASSED! Variable [ a ] on line 6 has type int and is assigned the correct type: int...  S.ANALYZE --> | ERROR! Variable [ a ] on line 7 has already been declared in current scope...  Semantic Analysis Failed With 0 WARNING(S) and 1 ERROR(S)... |
| {  int a  a = 0  string z  z = "bond"  while (a != 9) {  if (a != 5) {  print("bond")  }  {  a = 1 + a  string b  b = "james bond"  print(b)  }  }  {/\*Holy Hell This is Disgusting\*/}  boolean c  c = true  boolean d  d = (true == (true == false))  d = (a == b)  d = (1 == a)  d = (1 != 1)  d = ("string" == 1)  d = (a != "string")  d = ("string" != "string")  if (d == true) {  int c  c = 1 + d  if (c == 1) {  print("ugh")  }  }  while ("string" == a) {  while (1 == true) {  a = 1 + "string"  }  }  }$ | Beginning Lexing Session... \*Stings Treated As CharList\*  LEXER --> | T\_OPENING\_BRACE [ { ] on line 1...  LEXER --> | T\_VARIABLE\_TYPE [ int ] on line 2...  LEXER --> | T\_ID [ a ] on line 2...  LEXER --> | T\_ID [ a ] on line 3...  LEXER --> | T\_ASSIGNMENT\_OP [ = ] on line 3...  LEXER --> | T\_DIGIT [ 0 ] on line 3...  LEXER --> | T\_VARIABLE\_TYPE [ string ] on line 4...  LEXER --> | T\_ID [ z ] on line 4...  LEXER --> | T\_ID [ z ] on line 5...  LEXER --> | T\_ASSIGNMENT\_OP [ = ] on line 5...  LEXER --> | T\_QUOTE [ " ] on line 5...  LEXER --> | T\_CHAR [ b ] on line 5...  LEXER --> | T\_CHAR [ o ] on line 5...  LEXER --> | T\_CHAR [ n ] on line 5...  LEXER --> | T\_CHAR [ d ] on line 5...  LEXER --> | T\_QUOTE [ " ] on line 5...  LEXER --> | T\_WHILE [ while ] on line 6...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 6...  LEXER --> | T\_ID [ a ] on line 6...  LEXER --> | T\_INEQUALITY\_OP [ != ] on line 6...  LEXER --> | T\_DIGIT [ 9 ] on line 6...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 6...  LEXER --> | T\_OPENING\_BRACE [ { ] on line 6...  LEXER --> | T\_IF [ if ] on line 7...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 7...  LEXER --> | T\_ID [ a ] on line 7...  LEXER --> | T\_INEQUALITY\_OP [ != ] on line 7...  LEXER --> | T\_DIGIT [ 5 ] on line 7...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 7...  LEXER --> | T\_OPENING\_BRACE [ { ] on line 7...  LEXER --> | T\_PRINT [ print ] on line 8...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 8...  LEXER --> | T\_QUOTE [ " ] on line 8...  LEXER --> | T\_CHAR [ b ] on line 8...  LEXER --> | T\_CHAR [ o ] on line 8...  LEXER --> | T\_CHAR [ n ] on line 8...  LEXER --> | T\_CHAR [ d ] on line 8...  LEXER --> | T\_QUOTE [ " ] on line 8...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 8...  LEXER --> | T\_CLOSING\_BRACE [ } ] on line 9...  LEXER --> | T\_OPENING\_BRACE [ { ] on line 10...  LEXER --> | T\_ID [ a ] on line 11...  LEXER --> | T\_ASSIGNMENT\_OP [ = ] on line 11...  LEXER --> | T\_DIGIT [ 1 ] on line 11...  LEXER --> | T\_ADDITION\_OP [ + ] on line 11...  LEXER --> | T\_ID [ a ] on line 11...  LEXER --> | T\_VARIABLE\_TYPE [ string ] on line 12...  LEXER --> | T\_ID [ b ] on line 12...  LEXER --> | T\_ID [ b ] on line 13...  LEXER --> | T\_ASSIGNMENT\_OP [ = ] on line 13...  LEXER --> | T\_QUOTE [ " ] on line 13...  LEXER --> | T\_CHAR [ j ] on line 13...  LEXER --> | T\_CHAR [ a ] on line 13...  LEXER --> | T\_CHAR [ m ] on line 13...  LEXER --> | T\_CHAR [ e ] on line 13...  LEXER --> | T\_CHAR [ s ] on line 13...  LEXER --> | T\_CHAR [ ] on line 13...  LEXER --> | T\_CHAR [ b ] on line 13...  LEXER --> | T\_CHAR [ o ] on line 13...  LEXER --> | T\_CHAR [ n ] on line 13...  LEXER --> | T\_CHAR [ d ] on line 13...  LEXER --> | T\_QUOTE [ " ] on line 13...  LEXER --> | T\_PRINT [ print ] on line 14...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 14...  LEXER --> | T\_ID [ b ] on line 14...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 14...  LEXER --> | T\_CLOSING\_BRACE [ } ] on line 15...  LEXER --> | T\_CLOSING\_BRACE [ } ] on line 16...  LEXER --> | T\_OPENING\_BRACE [ { ] on line 17...  LEXER --> | T\_CLOSING\_BRACE [ } ] on line 17...  LEXER --> | T\_VARIABLE\_TYPE [ boolean ] on line 18...  LEXER --> | T\_ID [ c ] on line 18...  LEXER --> | T\_ID [ c ] on line 19...  LEXER --> | T\_ASSIGNMENT\_OP [ = ] on line 19...  LEXER --> | T\_BOOLEAN\_VALUE [ true ] on line 19...  LEXER --> | T\_VARIABLE\_TYPE [ boolean ] on line 20...  LEXER --> | T\_ID [ d ] on line 20...  LEXER --> | T\_ID [ d ] on line 21...  LEXER --> | T\_ASSIGNMENT\_OP [ = ] on line 21...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 21...  LEXER --> | T\_BOOLEAN\_VALUE [ true ] on line 21...  LEXER --> | T\_EQUALITY\_OP [ == ] on line 21...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 21...  LEXER --> | T\_BOOLEAN\_VALUE [ true ] on line 21...  LEXER --> | T\_EQUALITY\_OP [ == ] on line 21...  LEXER --> | T\_BOOLEAN\_VALUE [ false ] on line 21...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 21...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 21...  LEXER --> | T\_ID [ d ] on line 22...  LEXER --> | T\_ASSIGNMENT\_OP [ = ] on line 22...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 22...  LEXER --> | T\_ID [ a ] on line 22...  LEXER --> | T\_EQUALITY\_OP [ == ] on line 22...  LEXER --> | T\_ID [ b ] on line 22...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 22...  LEXER --> | T\_ID [ d ] on line 23...  LEXER --> | T\_ASSIGNMENT\_OP [ = ] on line 23...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 23...  LEXER --> | T\_DIGIT [ 1 ] on line 23...  LEXER --> | T\_EQUALITY\_OP [ == ] on line 23...  LEXER --> | T\_ID [ a ] on line 23...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 23...  LEXER --> | T\_ID [ d ] on line 24...  LEXER --> | T\_ASSIGNMENT\_OP [ = ] on line 24...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 24...  LEXER --> | T\_DIGIT [ 1 ] on line 24...  LEXER --> | T\_INEQUALITY\_OP [ != ] on line 24...  LEXER --> | T\_DIGIT [ 1 ] on line 24...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 24...  LEXER --> | T\_ID [ d ] on line 25...  LEXER --> | T\_ASSIGNMENT\_OP [ = ] on line 25...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 25...  LEXER --> | T\_QUOTE [ " ] on line 25...  LEXER --> | T\_CHAR [ s ] on line 25...  LEXER --> | T\_CHAR [ t ] on line 25...  LEXER --> | T\_CHAR [ r ] on line 25...  LEXER --> | T\_CHAR [ i ] on line 25...  LEXER --> | T\_CHAR [ n ] on line 25...  LEXER --> | T\_CHAR [ g ] on line 25...  LEXER --> | T\_QUOTE [ " ] on line 25...  LEXER --> | T\_EQUALITY\_OP [ == ] on line 25...  LEXER --> | T\_DIGIT [ 1 ] on line 25...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 25...  LEXER --> | T\_ID [ d ] on line 26...  LEXER --> | T\_ASSIGNMENT\_OP [ = ] on line 26...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 26...  LEXER --> | T\_ID [ a ] on line 26...  LEXER --> | T\_INEQUALITY\_OP [ != ] on line 26...  LEXER --> | T\_QUOTE [ " ] on line 26...  LEXER --> | T\_CHAR [ s ] on line 26...  LEXER --> | T\_CHAR [ t ] on line 26...  LEXER --> | T\_CHAR [ r ] on line 26...  LEXER --> | T\_CHAR [ i ] on line 26...  LEXER --> | T\_CHAR [ n ] on line 26...  LEXER --> | T\_CHAR [ g ] on line 26...  LEXER --> | T\_QUOTE [ " ] on line 26...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 26...  LEXER --> | T\_ID [ d ] on line 27...  LEXER --> | T\_ASSIGNMENT\_OP [ = ] on line 27...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 27...  LEXER --> | T\_QUOTE [ " ] on line 27...  LEXER --> | T\_CHAR [ s ] on line 27...  LEXER --> | T\_CHAR [ t ] on line 27...  LEXER --> | T\_CHAR [ r ] on line 27...  LEXER --> | T\_CHAR [ i ] on line 27...  LEXER --> | T\_CHAR [ n ] on line 27...  LEXER --> | T\_CHAR [ g ] on line 27...  LEXER --> | T\_QUOTE [ " ] on line 27...  LEXER --> | T\_INEQUALITY\_OP [ != ] on line 27...  LEXER --> | T\_QUOTE [ " ] on line 27...  LEXER --> | T\_CHAR [ s ] on line 27...  LEXER --> | T\_CHAR [ t ] on line 27...  LEXER --> | T\_CHAR [ r ] on line 27...  LEXER --> | T\_CHAR [ i ] on line 27...  LEXER --> | T\_CHAR [ n ] on line 27...  LEXER --> | T\_CHAR [ g ] on line 27...  LEXER --> | T\_QUOTE [ " ] on line 27...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 27...  LEXER --> | T\_IF [ if ] on line 28...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 28...  LEXER --> | T\_ID [ d ] on line 28...  LEXER --> | T\_EQUALITY\_OP [ == ] on line 28...  LEXER --> | T\_BOOLEAN\_VALUE [ true ] on line 28...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 28...  LEXER --> | T\_OPENING\_BRACE [ { ] on line 28...  LEXER --> | T\_VARIABLE\_TYPE [ int ] on line 29...  LEXER --> | T\_ID [ c ] on line 29...  LEXER --> | T\_ID [ c ] on line 30...  LEXER --> | T\_ASSIGNMENT\_OP [ = ] on line 30...  LEXER --> | T\_DIGIT [ 1 ] on line 30...  LEXER --> | T\_ADDITION\_OP [ + ] on line 30...  LEXER --> | T\_ID [ d ] on line 30...  LEXER --> | T\_IF [ if ] on line 31...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 31...  LEXER --> | T\_ID [ c ] on line 31...  LEXER --> | T\_EQUALITY\_OP [ == ] on line 31...  LEXER --> | T\_DIGIT [ 1 ] on line 31...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 31...  LEXER --> | T\_OPENING\_BRACE [ { ] on line 31...  LEXER --> | T\_PRINT [ print ] on line 32...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 32...  LEXER --> | T\_QUOTE [ " ] on line 32...  LEXER --> | T\_CHAR [ u ] on line 32...  LEXER --> | T\_CHAR [ g ] on line 32...  LEXER --> | T\_CHAR [ h ] on line 32...  LEXER --> | T\_QUOTE [ " ] on line 32...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 32...  LEXER --> | T\_CLOSING\_BRACE [ } ] on line 33...  LEXER --> | T\_CLOSING\_BRACE [ } ] on line 34...  LEXER --> | T\_WHILE [ while ] on line 35...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 35...  LEXER --> | T\_QUOTE [ " ] on line 35...  LEXER --> | T\_CHAR [ s ] on line 35...  LEXER --> | T\_CHAR [ t ] on line 35...  LEXER --> | T\_CHAR [ r ] on line 35...  LEXER --> | T\_CHAR [ i ] on line 35...  LEXER --> | T\_CHAR [ n ] on line 35...  LEXER --> | T\_CHAR [ g ] on line 35...  LEXER --> | T\_QUOTE [ " ] on line 35...  LEXER --> | T\_EQUALITY\_OP [ == ] on line 35...  LEXER --> | T\_ID [ a ] on line 35...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 35...  LEXER --> | T\_OPENING\_BRACE [ { ] on line 35...  LEXER --> | T\_WHILE [ while ] on line 36...  LEXER --> | T\_OPENING\_PARENTHESIS [ ( ] on line 36...  LEXER --> | T\_DIGIT [ 1 ] on line 36...  LEXER --> | T\_EQUALITY\_OP [ == ] on line 36...  LEXER --> | T\_BOOLEAN\_VALUE [ true ] on line 36...  LEXER --> | T\_CLOSING\_PARENTHESIS [ ) ] on line 36...  LEXER --> | T\_OPENING\_BRACE [ { ] on line 36...  LEXER --> | T\_ID [ a ] on line 37...  LEXER --> | T\_ASSIGNMENT\_OP [ = ] on line 37...  LEXER --> | T\_DIGIT [ 1 ] on line 37...  LEXER --> | T\_ADDITION\_OP [ + ] on line 37...  LEXER --> | T\_QUOTE [ " ] on line 37...  LEXER --> | T\_CHAR [ s ] on line 37...  LEXER --> | T\_CHAR [ t ] on line 37...  LEXER --> | T\_CHAR [ r ] on line 37...  LEXER --> | T\_CHAR [ i ] on line 37...  LEXER --> | T\_CHAR [ n ] on line 37...  LEXER --> | T\_CHAR [ g ] on line 37...  LEXER --> | T\_QUOTE [ " ] on line 37...  LEXER --> | T\_CLOSING\_BRACE [ } ] on line 38...  LEXER --> | T\_CLOSING\_BRACE [ } ] on line 39...  LEXER --> | T\_CLOSING\_BRACE [ } ] on line 40...  LEXER --> | T\_EOPS [ $ ] on line 40...  Lex Completed With 0 WARNING(S) and 0 ERROR(S)...  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Beginning Parsing Session...  PARSER --> | PASSED! Expecting [ T\_OPENING\_BRACE ] found [ T\_OPENING\_BRACE ] on line 1...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 2...  PARSER --> | PASSED! Expecting [ VarDecl ] found [ T\_VARIABLE\_TYPE ] on line 2...  PARSER --> | PASSED! Expecting [ T\_VARIABLE\_TYPE ] found [ T\_VARIABLE\_TYPE ] on line 2...  PARSER --> | PASSED! Expecting [ T\_VARIABLE\_TYPE ] found [ int ] on line 2...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 2...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 2...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 3...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 3...  PARSER --> | PASSED! Expecting [ AssignmentStatement ] found [ T\_ID ] on line 3...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 3...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 3...  PARSER --> | PASSED! Expecting [ T\_ASSIGNMENT\_OP ] found [ = ] on line 3...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 3...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ IntExpr ] on line 3...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 0 ] on line 3...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 4...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 4...  PARSER --> | PASSED! Expecting [ VarDecl ] found [ T\_VARIABLE\_TYPE ] on line 4...  PARSER --> | PASSED! Expecting [ T\_VARIABLE\_TYPE ] found [ T\_VARIABLE\_TYPE ] on line 4...  PARSER --> | PASSED! Expecting [ T\_VARIABLE\_TYPE ] found [ string ] on line 4...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 4...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ z ] on line 4...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 5...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 5...  PARSER --> | PASSED! Expecting [ AssignmentStatement ] found [ T\_ID ] on line 5...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 5...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ z ] on line 5...  PARSER --> | PASSED! Expecting [ T\_ASSIGNMENT\_OP ] found [ = ] on line 5...  PARSER --> | PASSED! Expecting [ Expr ] found [ StringExpr ] on line 5...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 5...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 5...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ b ] on line 5...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 5...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ o ] on line 5...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 5...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ n ] on line 5...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 5...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ d ] on line 5...  PARSER --> | PASSED! λ production on line 5...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 5...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 6...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 6...  PARSER --> | PASSED! Expecting [ WhileStatement ] found [ T\_WHILE ] on line 6...  PARSER --> | PASSED! Expecting [ T\_WHILE ] found [ while ] on line 6...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ ( ] on line 6...  PARSER --> | PASSED! Expecting [ Expr ] found [ Id ] on line 6...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 6...  PARSER --> | PASSED! Expecting [ T\_INEQUALITY\_OP ] found [ != ] on line 6...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 6...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ IntExpr ] on line 6...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 9 ] on line 6...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ ) ] on line 6...  PARSER --> | PASSED! Expecting [ T\_OPENING\_BRACE ] found [ T\_OPENING\_BRACE ] on line 6...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 7...  PARSER --> | PASSED! Expecting [ IfStatement ] found [ T\_IF ] on line 7...  PARSER --> | PASSED! Expecting [ T\_IF ] found [ if ] on line 7...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ ( ] on line 7...  PARSER --> | PASSED! Expecting [ Expr ] found [ Id ] on line 7...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 7...  PARSER --> | PASSED! Expecting [ T\_INEQUALITY\_OP ] found [ != ] on line 7...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 7...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ IntExpr ] on line 7...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 5 ] on line 7...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ ) ] on line 7...  PARSER --> | PASSED! Expecting [ T\_OPENING\_BRACE ] found [ T\_OPENING\_BRACE ] on line 7...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 8...  PARSER --> | PASSED! Expecting [ PrintStatement ] found [ T\_PRINT ] on line 8...  PARSER --> | PASSED! Expecting [ T\_PRINT ] found [ T\_PRINT ] on line 8...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ T\_OPENING\_PARENTHESIS ] on line 8...  PARSER --> | PASSED! Expecting [ Expr ] found [ StringExpr ] on line 8...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 8...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 8...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ b ] on line 8...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 8...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ o ] on line 8...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 8...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ n ] on line 8...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 8...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ d ] on line 8...  PARSER --> | PASSED! λ production on line 8...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 8...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ T\_CLOSING\_PARENTHESIS ] on line 8...  PARSER --> | PASSED! λ production on line 9...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_BRACE ] found [ T\_CLOSING\_BRACE ] on line 9...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 10...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 10...  PARSER --> | PASSED! Expecting [ T\_OPENING\_BRACE ] found [ T\_OPENING\_BRACE ] on line 10...  PARSER --> | PASSED! Expecting [ T\_OPENING\_BRACE ] found [ T\_OPENING\_BRACE ] on line 10...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 11...  PARSER --> | PASSED! Expecting [ AssignmentStatement ] found [ T\_ID ] on line 11...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 11...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 11...  PARSER --> | PASSED! Expecting [ T\_ASSIGNMENT\_OP ] found [ = ] on line 11...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 11...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ T\_DIGIT ] on line 11...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 1 ] on line 11...  PARSER --> | PASSED! Expecting [ T\_ADDITION\_OP ] found [ + ] on line 11...  PARSER --> | PASSED! Expecting [ Expr ] found [ Id ] on line 11...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 11...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 12...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 12...  PARSER --> | PASSED! Expecting [ VarDecl ] found [ T\_VARIABLE\_TYPE ] on line 12...  PARSER --> | PASSED! Expecting [ T\_VARIABLE\_TYPE ] found [ T\_VARIABLE\_TYPE ] on line 12...  PARSER --> | PASSED! Expecting [ T\_VARIABLE\_TYPE ] found [ string ] on line 12...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 12...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ b ] on line 12...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 13...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 13...  PARSER --> | PASSED! Expecting [ AssignmentStatement ] found [ T\_ID ] on line 13...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 13...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ b ] on line 13...  PARSER --> | PASSED! Expecting [ T\_ASSIGNMENT\_OP ] found [ = ] on line 13...  PARSER --> | PASSED! Expecting [ Expr ] found [ StringExpr ] on line 13...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 13...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 13...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ j ] on line 13...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 13...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ a ] on line 13...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 13...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ m ] on line 13...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 13...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ e ] on line 13...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 13...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ s ] on line 13...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 13...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ ] on line 13...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 13...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ b ] on line 13...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 13...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ o ] on line 13...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 13...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ n ] on line 13...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 13...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ d ] on line 13...  PARSER --> | PASSED! λ production on line 13...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 13...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 14...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 14...  PARSER --> | PASSED! Expecting [ PrintStatement ] found [ T\_PRINT ] on line 14...  PARSER --> | PASSED! Expecting [ T\_PRINT ] found [ T\_PRINT ] on line 14...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ T\_OPENING\_PARENTHESIS ] on line 14...  PARSER --> | PASSED! Expecting [ Expr ] found [ Id ] on line 14...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ b ] on line 14...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ T\_CLOSING\_PARENTHESIS ] on line 14...  PARSER --> | PASSED! λ production on line 15...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_BRACE ] found [ T\_CLOSING\_BRACE ] on line 15...  PARSER --> | PASSED! λ production on line 16...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_BRACE ] found [ T\_CLOSING\_BRACE ] on line 16...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 17...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 17...  PARSER --> | PASSED! Expecting [ T\_OPENING\_BRACE ] found [ T\_OPENING\_BRACE ] on line 17...  PARSER --> | PASSED! Expecting [ T\_OPENING\_BRACE ] found [ T\_OPENING\_BRACE ] on line 17...  PARSER --> | PASSED! λ production on line 17...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_BRACE ] found [ T\_CLOSING\_BRACE ] on line 17...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 18...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 18...  PARSER --> | PASSED! Expecting [ VarDecl ] found [ T\_VARIABLE\_TYPE ] on line 18...  PARSER --> | PASSED! Expecting [ T\_VARIABLE\_TYPE ] found [ T\_VARIABLE\_TYPE ] on line 18...  PARSER --> | PASSED! Expecting [ T\_VARIABLE\_TYPE ] found [ boolean ] on line 18...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 18...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ c ] on line 18...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 19...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 19...  PARSER --> | PASSED! Expecting [ AssignmentStatement ] found [ T\_ID ] on line 19...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 19...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ c ] on line 19...  PARSER --> | PASSED! Expecting [ T\_ASSIGNMENT\_OP ] found [ = ] on line 19...  PARSER --> | PASSED! Expecting [ Expr ] found [ BooleanExpr ] on line 19...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ T\_BOOLEAN\_VALUE ] on line 19...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ true ] on line 19...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 20...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 20...  PARSER --> | PASSED! Expecting [ VarDecl ] found [ T\_VARIABLE\_TYPE ] on line 20...  PARSER --> | PASSED! Expecting [ T\_VARIABLE\_TYPE ] found [ T\_VARIABLE\_TYPE ] on line 20...  PARSER --> | PASSED! Expecting [ T\_VARIABLE\_TYPE ] found [ boolean ] on line 20...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 20...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ d ] on line 20...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 21...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 21...  PARSER --> | PASSED! Expecting [ AssignmentStatement ] found [ T\_ID ] on line 21...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 21...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ d ] on line 21...  PARSER --> | PASSED! Expecting [ T\_ASSIGNMENT\_OP ] found [ = ] on line 21...  PARSER --> | PASSED! Expecting [ Expr ] found [ BooleanExpr ] on line 21...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ ( ] on line 21...  PARSER --> | PASSED! Expecting [ Expr ] found [ BooleanExpr ] on line 21...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ T\_BOOLEAN\_VALUE ] on line 21...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ true ] on line 21...  PARSER --> | PASSED! Expecting [ T\_EQUALITY\_OP ] found [ == ] on line 21...  PARSER --> | PASSED! Expecting [ Expr ] found [ BooleanExpr ] on line 21...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ ( ] on line 21...  PARSER --> | PASSED! Expecting [ Expr ] found [ BooleanExpr ] on line 21...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ T\_BOOLEAN\_VALUE ] on line 21...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ true ] on line 21...  PARSER --> | PASSED! Expecting [ T\_EQUALITY\_OP ] found [ == ] on line 21...  PARSER --> | PASSED! Expecting [ Expr ] found [ BooleanExpr ] on line 21...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ T\_BOOLEAN\_VALUE ] on line 21...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ false ] on line 21...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ ) ] on line 21...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ ) ] on line 21...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 22...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 22...  PARSER --> | PASSED! Expecting [ AssignmentStatement ] found [ T\_ID ] on line 22...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 22...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ d ] on line 22...  PARSER --> | PASSED! Expecting [ T\_ASSIGNMENT\_OP ] found [ = ] on line 22...  PARSER --> | PASSED! Expecting [ Expr ] found [ BooleanExpr ] on line 22...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ ( ] on line 22...  PARSER --> | PASSED! Expecting [ Expr ] found [ Id ] on line 22...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 22...  PARSER --> | PASSED! Expecting [ T\_EQUALITY\_OP ] found [ == ] on line 22...  PARSER --> | PASSED! Expecting [ Expr ] found [ Id ] on line 22...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ b ] on line 22...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ ) ] on line 22...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 23...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 23...  PARSER --> | PASSED! Expecting [ AssignmentStatement ] found [ T\_ID ] on line 23...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 23...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ d ] on line 23...  PARSER --> | PASSED! Expecting [ T\_ASSIGNMENT\_OP ] found [ = ] on line 23...  PARSER --> | PASSED! Expecting [ Expr ] found [ BooleanExpr ] on line 23...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ ( ] on line 23...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 23...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ IntExpr ] on line 23...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 1 ] on line 23...  PARSER --> | PASSED! Expecting [ T\_EQUALITY\_OP ] found [ == ] on line 23...  PARSER --> | PASSED! Expecting [ Expr ] found [ Id ] on line 23...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 23...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ ) ] on line 23...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 24...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 24...  PARSER --> | PASSED! Expecting [ AssignmentStatement ] found [ T\_ID ] on line 24...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 24...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ d ] on line 24...  PARSER --> | PASSED! Expecting [ T\_ASSIGNMENT\_OP ] found [ = ] on line 24...  PARSER --> | PASSED! Expecting [ Expr ] found [ BooleanExpr ] on line 24...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ ( ] on line 24...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 24...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ IntExpr ] on line 24...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 1 ] on line 24...  PARSER --> | PASSED! Expecting [ T\_INEQUALITY\_OP ] found [ != ] on line 24...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 24...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ IntExpr ] on line 24...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 1 ] on line 24...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ ) ] on line 24...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 25...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 25...  PARSER --> | PASSED! Expecting [ AssignmentStatement ] found [ T\_ID ] on line 25...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 25...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ d ] on line 25...  PARSER --> | PASSED! Expecting [ T\_ASSIGNMENT\_OP ] found [ = ] on line 25...  PARSER --> | PASSED! Expecting [ Expr ] found [ BooleanExpr ] on line 25...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ ( ] on line 25...  PARSER --> | PASSED! Expecting [ Expr ] found [ StringExpr ] on line 25...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 25...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 25...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ s ] on line 25...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 25...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ t ] on line 25...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 25...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ r ] on line 25...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 25...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ i ] on line 25...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 25...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ n ] on line 25...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 25...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ g ] on line 25...  PARSER --> | PASSED! λ production on line 25...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 25...  PARSER --> | PASSED! Expecting [ T\_EQUALITY\_OP ] found [ == ] on line 25...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 25...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ IntExpr ] on line 25...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 1 ] on line 25...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ ) ] on line 25...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 26...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 26...  PARSER --> | PASSED! Expecting [ AssignmentStatement ] found [ T\_ID ] on line 26...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 26...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ d ] on line 26...  PARSER --> | PASSED! Expecting [ T\_ASSIGNMENT\_OP ] found [ = ] on line 26...  PARSER --> | PASSED! Expecting [ Expr ] found [ BooleanExpr ] on line 26...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ ( ] on line 26...  PARSER --> | PASSED! Expecting [ Expr ] found [ Id ] on line 26...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 26...  PARSER --> | PASSED! Expecting [ T\_INEQUALITY\_OP ] found [ != ] on line 26...  PARSER --> | PASSED! Expecting [ Expr ] found [ StringExpr ] on line 26...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 26...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 26...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ s ] on line 26...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 26...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ t ] on line 26...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 26...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ r ] on line 26...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 26...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ i ] on line 26...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 26...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ n ] on line 26...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 26...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ g ] on line 26...  PARSER --> | PASSED! λ production on line 26...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 26...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ ) ] on line 26...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 27...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 27...  PARSER --> | PASSED! Expecting [ AssignmentStatement ] found [ T\_ID ] on line 27...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 27...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ d ] on line 27...  PARSER --> | PASSED! Expecting [ T\_ASSIGNMENT\_OP ] found [ = ] on line 27...  PARSER --> | PASSED! Expecting [ Expr ] found [ BooleanExpr ] on line 27...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ ( ] on line 27...  PARSER --> | PASSED! Expecting [ Expr ] found [ StringExpr ] on line 27...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 27...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 27...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ s ] on line 27...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 27...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ t ] on line 27...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 27...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ r ] on line 27...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 27...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ i ] on line 27...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 27...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ n ] on line 27...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 27...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ g ] on line 27...  PARSER --> | PASSED! λ production on line 27...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 27...  PARSER --> | PASSED! Expecting [ T\_INEQUALITY\_OP ] found [ != ] on line 27...  PARSER --> | PASSED! Expecting [ Expr ] found [ StringExpr ] on line 27...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 27...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 27...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ s ] on line 27...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 27...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ t ] on line 27...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 27...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ r ] on line 27...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 27...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ i ] on line 27...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 27...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ n ] on line 27...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 27...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ g ] on line 27...  PARSER --> | PASSED! λ production on line 27...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 27...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ ) ] on line 27...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 28...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 28...  PARSER --> | PASSED! Expecting [ IfStatement ] found [ T\_IF ] on line 28...  PARSER --> | PASSED! Expecting [ T\_IF ] found [ if ] on line 28...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ ( ] on line 28...  PARSER --> | PASSED! Expecting [ Expr ] found [ Id ] on line 28...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ d ] on line 28...  PARSER --> | PASSED! Expecting [ T\_EQUALITY\_OP ] found [ == ] on line 28...  PARSER --> | PASSED! Expecting [ Expr ] found [ BooleanExpr ] on line 28...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ T\_BOOLEAN\_VALUE ] on line 28...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ true ] on line 28...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ ) ] on line 28...  PARSER --> | PASSED! Expecting [ T\_OPENING\_BRACE ] found [ T\_OPENING\_BRACE ] on line 28...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 29...  PARSER --> | PASSED! Expecting [ VarDecl ] found [ T\_VARIABLE\_TYPE ] on line 29...  PARSER --> | PASSED! Expecting [ T\_VARIABLE\_TYPE ] found [ T\_VARIABLE\_TYPE ] on line 29...  PARSER --> | PASSED! Expecting [ T\_VARIABLE\_TYPE ] found [ int ] on line 29...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 29...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ c ] on line 29...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 30...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 30...  PARSER --> | PASSED! Expecting [ AssignmentStatement ] found [ T\_ID ] on line 30...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 30...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ c ] on line 30...  PARSER --> | PASSED! Expecting [ T\_ASSIGNMENT\_OP ] found [ = ] on line 30...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 30...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ T\_DIGIT ] on line 30...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 1 ] on line 30...  PARSER --> | PASSED! Expecting [ T\_ADDITION\_OP ] found [ + ] on line 30...  PARSER --> | PASSED! Expecting [ Expr ] found [ Id ] on line 30...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ d ] on line 30...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 31...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 31...  PARSER --> | PASSED! Expecting [ IfStatement ] found [ T\_IF ] on line 31...  PARSER --> | PASSED! Expecting [ T\_IF ] found [ if ] on line 31...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ ( ] on line 31...  PARSER --> | PASSED! Expecting [ Expr ] found [ Id ] on line 31...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ c ] on line 31...  PARSER --> | PASSED! Expecting [ T\_EQUALITY\_OP ] found [ == ] on line 31...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 31...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ IntExpr ] on line 31...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 1 ] on line 31...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ ) ] on line 31...  PARSER --> | PASSED! Expecting [ T\_OPENING\_BRACE ] found [ T\_OPENING\_BRACE ] on line 31...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 32...  PARSER --> | PASSED! Expecting [ PrintStatement ] found [ T\_PRINT ] on line 32...  PARSER --> | PASSED! Expecting [ T\_PRINT ] found [ T\_PRINT ] on line 32...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ T\_OPENING\_PARENTHESIS ] on line 32...  PARSER --> | PASSED! Expecting [ Expr ] found [ StringExpr ] on line 32...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 32...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 32...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ u ] on line 32...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 32...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ g ] on line 32...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 32...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ h ] on line 32...  PARSER --> | PASSED! λ production on line 32...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 32...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ T\_CLOSING\_PARENTHESIS ] on line 32...  PARSER --> | PASSED! λ production on line 33...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_BRACE ] found [ T\_CLOSING\_BRACE ] on line 33...  PARSER --> | PASSED! λ production on line 34...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_BRACE ] found [ T\_CLOSING\_BRACE ] on line 34...  PARSER --> | PASSED! Expecting [ StatementList ] found [ Statement ] on line 35...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 35...  PARSER --> | PASSED! Expecting [ WhileStatement ] found [ T\_WHILE ] on line 35...  PARSER --> | PASSED! Expecting [ T\_WHILE ] found [ while ] on line 35...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ ( ] on line 35...  PARSER --> | PASSED! Expecting [ Expr ] found [ StringExpr ] on line 35...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 35...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 35...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ s ] on line 35...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 35...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ t ] on line 35...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 35...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ r ] on line 35...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 35...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ i ] on line 35...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 35...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ n ] on line 35...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 35...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ g ] on line 35...  PARSER --> | PASSED! λ production on line 35...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 35...  PARSER --> | PASSED! Expecting [ T\_EQUALITY\_OP ] found [ == ] on line 35...  PARSER --> | PASSED! Expecting [ Expr ] found [ Id ] on line 35...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 35...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ ) ] on line 35...  PARSER --> | PASSED! Expecting [ T\_OPENING\_BRACE ] found [ T\_OPENING\_BRACE ] on line 35...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 36...  PARSER --> | PASSED! Expecting [ WhileStatement ] found [ T\_WHILE ] on line 36...  PARSER --> | PASSED! Expecting [ T\_WHILE ] found [ while ] on line 36...  PARSER --> | PASSED! Expecting [ T\_OPENING\_PARENTHESIS ] found [ ( ] on line 36...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 36...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ IntExpr ] on line 36...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 1 ] on line 36...  PARSER --> | PASSED! Expecting [ T\_EQUALITY\_OP ] found [ == ] on line 36...  PARSER --> | PASSED! Expecting [ Expr ] found [ BooleanExpr ] on line 36...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ T\_BOOLEAN\_VALUE ] on line 36...  PARSER --> | PASSED! Expecting [ T\_BOOLEAN\_VALUE ] found [ true ] on line 36...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_PARENTHESIS ] found [ ) ] on line 36...  PARSER --> | PASSED! Expecting [ T\_OPENING\_BRACE ] found [ T\_OPENING\_BRACE ] on line 36...  PARSER --> | PASSED! Expecting [ Statement ] found [ Statement ] on line 37...  PARSER --> | PASSED! Expecting [ AssignmentStatement ] found [ T\_ID ] on line 37...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ T\_ID ] on line 37...  PARSER --> | PASSED! Expecting [ T\_ID ] found [ a ] on line 37...  PARSER --> | PASSED! Expecting [ T\_ASSIGNMENT\_OP ] found [ = ] on line 37...  PARSER --> | PASSED! Expecting [ Expr ] found [ IntExpr ] on line 37...  PARSER --> | PASSED! Expecting [ IntExpr ] found [ T\_DIGIT ] on line 37...  PARSER --> | PASSED! Expecting [ T\_DIGIT ] found [ 1 ] on line 37...  PARSER --> | PASSED! Expecting [ T\_ADDITION\_OP ] found [ + ] on line 37...  PARSER --> | PASSED! Expecting [ Expr ] found [ StringExpr ] on line 37...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 37...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 37...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ s ] on line 37...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 37...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ t ] on line 37...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 37...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ r ] on line 37...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 37...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ i ] on line 37...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 37...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ n ] on line 37...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ T\_CHAR ] on line 37...  PARSER --> | PASSED! Expecting [ T\_CHAR ] found [ g ] on line 37...  PARSER --> | PASSED! λ production on line 37...  PARSER --> | PASSED! Expecting [ T\_QUOTE ] found [ T\_QUOTE ] on line 37...  PARSER --> | PASSED! λ production on line 38...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_BRACE ] found [ T\_CLOSING\_BRACE ] on line 38...  PARSER --> | PASSED! λ production on line 39...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_BRACE ] found [ T\_CLOSING\_BRACE ] on line 39...  PARSER --> | PASSED! λ production on line 40...  PARSER --> | PASSED! Expecting [ T\_CLOSING\_BRACE ] found [ T\_CLOSING\_BRACE ] on line 40...  PARSER --> | PASSED! Expecting [ T\_EOPS ] found [ T\_EOPS ] on line 40...  Parse Completed With 0 WARNING(S) and 0 ERROR(S)...  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Beginning Semantic Analysis Session...  S.ANALYZE --> | PASSED! Variable [ a ] on line 3 has been declared...  S.ANALYZE --> | PASSED! Variable [ a ] on line 3 has [ int ] type...  S.ANALYZE --> | PASSED! Variable [ a ] on line 3 has been used...  S.ANALYZE --> | PASSED! Variable [ a ] on line 3 has type int and is assigned the correct type: int...  S.ANALYZE --> | PASSED! Variable [ z ] on line 5 has been declared...  S.ANALYZE --> | PASSED! Variable [ z ] on line 5 has [ string ] type...  S.ANALYZE --> | PASSED! Variable [ z ] on line 5 has been used...  S.ANALYZE --> | PASSED! Variable [ z ] on line 5 has type string and is assigned the correct type: string...  S.ANALYZE --> | PASSED! Variable [ a ] on line 6 has been declared...  S.ANALYZE --> | PASSED! Variable [ a ] on line 6 has been used...  S.ANALYZE --> | PASSED! Variable [ a ] on line 6 has [ int ] type...  S.ANALYZE --> | PASSED! Variable [ a ] on line 7 has been declared...  S.ANALYZE --> | PASSED! Variable [ a ] on line 7 has been used...  S.ANALYZE --> | PASSED! Variable [ a ] on line 7 has [ int ] type...  S.ANALYZE --> | PASSED! Variable [ a ] on line 11 has been declared...  S.ANALYZE --> | PASSED! Variable [ a ] on line 11 has [ int ] type...  S.ANALYZE --> | PASSED! Variable [ a ] on line 11 has been used...  S.ANALYZE --> | PASSED! Variable [ a ] on line 11 has been declared...  S.ANALYZE --> | PASSED! Variable [ a ] on line 11 has been used...  S.ANALYZE --> | PASSED! Variable [ a ] on line 11 has [ int ] type...  S.ANALYZE --> | PASSED! Variable [ a ] on line 11 has type int and is assigned the correct type: int...  S.ANALYZE --> | PASSED! Variable [ b ] on line 13 has been declared...  S.ANALYZE --> | PASSED! Variable [ b ] on line 13 has [ string ] type...  S.ANALYZE --> | PASSED! Variable [ b ] on line 13 has been used...  S.ANALYZE --> | PASSED! Variable [ b ] on line 13 has type string and is assigned the correct type: string...  S.ANALYZE --> | PASSED! Variable [ b ] on line 14 has been declared...  S.ANALYZE --> | PASSED! Variable [ b ] on line 14 has been used...  S.ANALYZE --> | PASSED! Variable [ b ] on line 14 has [ string ] type...  S.ANALYZE --> | PASSED! Variable [ c ] on line 19 has been declared...  S.ANALYZE --> | PASSED! Variable [ c ] on line 19 has [ boolean ] type...  S.ANALYZE --> | PASSED! Variable [ c ] on line 19 has been used...  S.ANALYZE --> | PASSED! Variable [ c ] on line 19 has type boolean and is assigned the correct type: boolean...  S.ANALYZE --> | PASSED! Variable [ d ] on line 21 has been declared...  S.ANALYZE --> | PASSED! Variable [ d ] on line 21 has [ boolean ] type...  S.ANALYZE --> | PASSED! Variable [ d ] on line 21 has been used...  S.ANALYZE --> | PASSED! Variable [ d ] on line 21 has type boolean and is assigned the correct type: boolean...  S.ANALYZE --> | PASSED! Variable [ d ] on line 22 has been declared...  S.ANALYZE --> | PASSED! Variable [ d ] on line 22 has [ boolean ] type...  S.ANALYZE --> | PASSED! Variable [ d ] on line 22 has been used...  S.ANALYZE --> | PASSED! Variable [ a ] on line 22 has been declared...  S.ANALYZE --> | PASSED! Variable [ a ] on line 22 has been used...  S.ANALYZE --> | PASSED! Variable [ a ] on line 22 has [ int ] type...  S.ANALYZE --> | ERROR! Variable [ b ] on line 22 is used before it is declared...  Semantic Analysis Failed With 0 WARNING(S) and 1 ERROR(S)... |