दिल्ली विश्वविद्यालय

University of Delhi

Parser Project

C++ : Ternary Operator(Without Nesting)

Submitted By:

Anurag Joshi(07)

Aryan Sindwani(09)

Deepak(10)

Submitted To:

Dr Ankit Rajpal

Input Format

- Input : Ternary Expression followed by a newline
- Ex : “b” ? a : 1+2 [\n]

Commands to Run(Linux)

1. lex <filename>
2. yacc <filename> [optional -v, -d, -y]
3. gcc lex.yy.c y.tab.c
4. ./a.out

Parser for C++: Ternary Operator (Without Nesting)

What does the parser do?

The parser analyses the syntax of a ternary expression provided by the user and tells whether the expression is valid or not

Accepted Test Cases:

The parser works perfectly for all the ideal test cases of the following syntax :

(Boolean_Expr) ? (Arithmetic_Expr) : (Arithmetic_Expr)

Here are some test cases for which the parser returns Valid Expression:

(i) (a>b)?e:f

(ii) a>b?e:f

(iii) true?e:f

(iv) “ab+?” ? ”1234” : ”34”+1

(v) (a-b+c>d)?e+f:g-h

(vi) And many more…

So, the parser recognizes any ternary expression as valid if it follows the rule of syntax provided.

Unaccepted Test Cases (Limitations) :

The parser has a number of weaknesses too. Here’s a list of test cases which the parser fails to recognize as Valid Ternary Statements or recognizes an Invalid statement valid.

(i) Since the parser is only valid for cases without nesting, if provided with cases with nesting it marks it as invalid.
Ex: a>b ? c : 2 ? e: f → Invalid

(ii) The parser does not accept unary operators.
Ex: ++a?1:2 → Invalid