Homework 13

The goal of this homework is to practice the builder pattern and visitor pattern. In the lecture, you are given an example of parsing a string of a binary expression using the builder patter. In that example, all nodes are of the same class Node, which has two pointers, m_leftChild and m_rightChild. This causes memory waste because all terminal nodes also have these two data members although they have no children. Also, it does not have a visitor that can calculate the value of the binary expression.

In this project, you need to design and implement classes to implement the following functionalities:
1. Using the builder pattern, given a binary expression as a string, your program can construct a binary expression tree using two types of nodes: terminal nodes, which have no children data members, and nonterminal nodes, which have children data members. You can refer to the example code in the lecture and Homework 12.
2. You need to develop a visitor that can calculate the value of a binary expression tree. You may need to use a stack. You may need a function such as istringstream() to convert a string to a value.
3. You need to develop classes so that the following main() function will produce the following output. Note that both the terminal node class and the nonterminal class need to have print operations so that they can be used to print out the binary expression tree.

```
int main(int argc, char** argv) {
    ExpBuilder builder;
    ExpParser parser;

    parser.setBuilder(&builder);
    string input="(((30+500)*70)-(60/50))";
    parser.parse(input);
    Node* root = builder.getExpression();
    cout<<input<<endl;
    root->print();

    StackBasedCalVisitor sbsv;
    root->Accept(&sbsv);
    cout<<endl<<"StackBasedSumVisitor Result: "<<sbsv.getResult()<<endl;
    return 0;
}
```

Output:
```
(((30+500)*70)-(60/50))
(((30+500)*70)-(60/50))
StackBasedSumVisitor Result: 37098.8
```

Due: March 22, 11:59PM, 2019.
Turn in one file via handin: the zip file of your whole NetBean directory. No UML. The name of your zip file should be: LastName_FirstName.zip. For

example, if your name is John Smith, you should turn in one files: Smith_John.zip.