

(1) Given a list of stock prices ordered by time for a single stock, find the difference between the best price to purchase and the best price to then sell the stock.

(2) Given two arrays, A1 and A2, find all the elements in A2 that **are not** in A1. Comment on the algorithmic complexity of your solution.

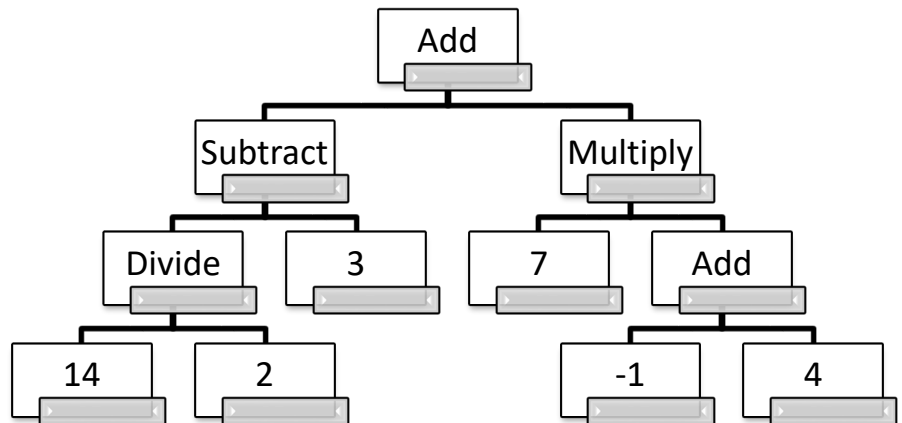
(3) Using the example code, implement a method `int Evaluate(NodeBase root)` to compute the value of an expression tree like the one below generated for the formula $((14/2) - 3) + (7 * (-1+4))$.

```
public enum Operator
{
    Add,
    Subtract,
    Multiply,
    Divide,
    AbsoluteValue
}

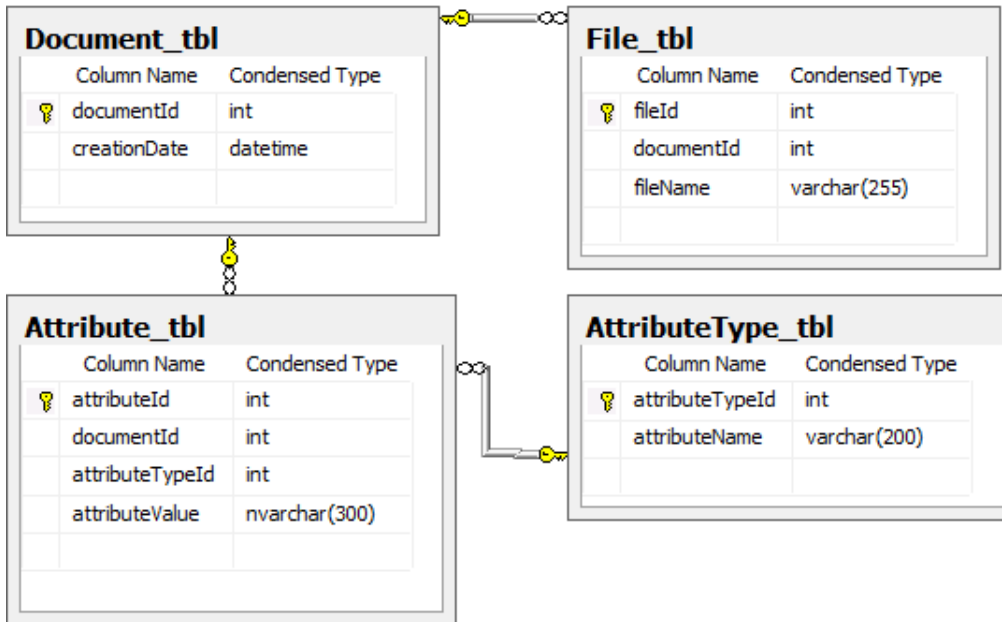
public abstract class NodeBase
{
}

public class NumberNode : NodeBase
{
    public int Value;
}

public class BinaryOperatorNode :
NodeBase
{
    public NodeBase Left;
    public NodeBase Right;
    public Operator Operator;
}
```



(4) Given the below data model and sample data:



AttributeType_tbl	
attributeTypeId	attributeName
1	filingDate
2	formType
3	filingCompany

Document_tbl	
documentId	creationDate
2	3/30/2013
6	3/6/1987
11	2/20/2012

File_tbl		
fileId	documentId	fileName
1	2	OKTO2.pdf
4	6	"Mainsail" Liquid Engine.xlsx
10	11	"Poodle" Liquid Engine.docx

Attribute_tbl			
attributeId	documentId	attributeTypeId	attributeValue
1	6	2	Marketing Materials
2	6	2	Spec Sheet
3	6	1	3/31/1987
4	6	3	Rockomax
5	2	1	1/1/2013
6	2	2	Marketing Materials
7	2	3	Probodobodyne
8	11	2	Marketing Materials
9	11	1	1/2/2012
10	11	3	Rockomax

(4a) Find all documents filed by Rockomax that are of formType "Marketing Materials" and not formType "Spec Sheet". Return the documentId, creationDate, and formType .

-
- i. Before you actually write the query, based on the above sample data, which document(s) will be returned (just write the documentId).
 - ii. Write the query.

(5) Write a function that is capable of adding two very large positive numbers ($\sim 10^{100}$) together, given as strings.