Compilation instructions:

open terminal in project folder (mac)

 $type: \verb|g++-o||./start|./module 3.cpp| operand.cpp| parse.cpp| subexpression.cpp| symbol table.cpp| consists of the context of the context$

variable.cpp type: ./start

Test case 1:

Test Case 1	Input: a+b a=5 b=1 a-b a=3 b=3 x*y x=4 y=2 x/y x=8 y=2	Expected output: 6 0 8	Actual output: 6 0 8 4	Pass
		4		

```
alexhong@Alexs-iMac cmsc330p2 % g++ -o ./start ./module3.cpp operand.cpp parse.cpp subexpression.cpp symboltable.cpp variable.cpp
alexhong@Alexs-iMac cmsc330p2 % ./start
Enter 1 to open input file 1
Enter 2 to open input file 3
Enter 4 to input file anae: 1

a+b, a=5, b=1;
Operation: (+)
a = 5
b = 1

Value = 6
a-b, a=3, b=3;
Operation: (-)
a = 3
b = 3
Value = 0

x*y, x=4, y=2;
Operation: (*)
x = 4
y = 2
Value = 8
x/y, x=8, y=2;
Operation: (/)
x = 8
y = 2
Value = 4

Thank you for using the evaluator. good bye
```

Test case 2:

Test Case 2	Input: a! a=5 a>b a=3 b=2 a <b a="" b="3">b a=1 b=2 a<b a="2" b="1</th"><th>Expected output: 120 1 1 0 0 0</th><th>Actual output: 120 1 1 0</th><th>Pass</th>	Expected output: 120 1 1 0 0 0	Actual output: 120 1 1 0	Pass
-------------	--	--------------------------------	--------------------------------------	------

```
alexhong@Alexs-iMac cmsc330p2 % ./start
Enter 1 to open input file 1
Enter 2 to open input file 3
Enter 3 to open input file 3
Enter 4 to input file name:

2

a!, a=5;
Operation: (!)
a = 5
Value = 120

a**s, a=3, b=2;
Operation: (>)
a = 3
b = 2
Value = 1

a**ch, a=2, b=3;
Operation: (c)
a = 2
Value = 1

a**b, a=2, b=3;
Operation: (c)
a = 2
b = 3
Value = 1

a**b, a=1, b=2;
Operation: (>)
a = 2
b = 3
Value = 0

A**ch, a=2, b=1;
Operation: (>)
a = 1
b = 2
Value = 0

A**ch, a=2, b=1;
Operation: (c)
a = 2
Value = 0

Thank you for using the evaluator. good bye
```

Test case 3:

Test Case 3	Input: a=b a=2 b=2 a=1 a=1 b=2 a&b a=1 b=1 a&b a=2 b=1	Expected output: 1 0 1	Actual output: 1 0 1 0	Pass
	a b a=2 b=1	0	1	
		1		

```
alexhong@Alexs-iMac cmsc330p2 % ./start
Enter 1 to open input file 1
Enter 2 to open input file 2
Enter 3 to open input file 3
Enter 4 to input file name: 3
a=b, a=2, b=2;
Operation: (=)
a = 2
b = 2
Value = 1
a=b, a=1, b=2;
Operation: (=)
a = 1
b = 2
Value = 0
a&b, a=1, b=1;
Operation: (&)
a = 1
b = 1
Value = 1
a&b, a=2, b=1;
Operation: (&)
a = 2
b = 1
Value = 0
a|b, a=2, b=1;
Operation: (|)
 = 2
= 1
Value = 1
Thank you for using the evaluator. good bye
```

Lessons learned: I learned the fundamentals of c++ and modifying existing source code. I consulted several online sources in addition to the course material to learn syntax and c++ program structure. I also learned the structure and inheritance protocol for c++ firsthand. Finally, I drew parallels in how to prompt for user input in c++ as well as in Java.

Development process: I started by adding the missing operation .h files. I then focused on taking in user input and feeding to the subexpression and operand classes for parsing. I then focused on making sure that dependent classes were included in the required .cpp files.I then focused on creating a menu that loops for user input.