Task_9 - 8 points

Expressions and Math Exam Implementation. Task contains of several topics implementation – inheritance with polymorphism, templates, exceptions and basics of STL. Expressions hierarchy represents mathematical expressions like addition or subtraction etc. Math Exam contains vector of Expressions that needs to be solved. Exceptions handles the unknown commands.

Task should be solved stage by stage by uncommenting consecutive code fragments in Main.cpp and implementation of missing classes or methods.

Stage_1 (2.0 Pts)

At the beginning you have to implement Expressions hierarchy. The implementation of abstract Expression class is given. It has two members which represents left and right arguments. One method is also given, Result which returns the result of expression. Your task is to implement Out Operator and Print method that prints basic information about expression – see example output below. Then expressions' hierarchy needs to be build. Implement following classes Addition, Subtraction, Multiplication and Division that derives from abstract Expression class. Implement all necessary methods and constructors.

Stage_2 (1.0 Pts)

Next step is to implement MathExam class. It contains std::vector of Expression pointers that represents all expressions that need to be solved during exam. The class already contains std::string parameter constructor that reads expressions from given file and push them into std::vector of Expressions.

Implement Answers method that returns vector of results of containing expressions. Additionally implement destructor that deletes all expressions inside the std::vector.

Stage_3 (2.0 Pts)

This time you should extend your MathExam class. Implement another constructor, that takes vector of characters representing expressions and vector of doubles which are values. It should create expressions (based on given characters) one by one and use values as arguments. There's N characters and 2*N values given, that means, that you should use two values to initialize one expression. See std::string constructor for some hints how to operate characters and create expressions based on them.

I.e. Signs {'+','*'}, Values {1,2,3,4}. Create Addition with values 1 and 2, and Multiplication with 3 and 4.

Additionally implement Solve method that prints expressions one by one onto the screen and asks user to provide correct answer. It should count all correct answers and print appropriate information at the end of the method – see example output below.

Stage_4 (2.0 Pts)

This stage requires changing everything into templates. Change implementation of Expressions' hierarchy and ExamClass into template classes. Comment out first three stages in Main.cpp.

Stage_5 (1.0 Pts)

Last stage contains of exceptions implementation. In Exceptions.h implement DivisionByZero exception and modify your Division's Result method to throw an exception when divider equals zero.

Example Output

```
-------------------- Stage_1 (2.0 Pts) --------------------
A2: Expression: 1 + 1 =  2
A3: Expression: 4 + 7 = 11
S2: Expression: 5 - 1 =  4
S3: Expression: 2 - 7 = -5
M2: Expression: 1 * 5 =  5
M4: Expression: 3 * 9 = 27
D2: Expression: 4 / 2 =  2
D4: Expression: 9 / 2 = 4.5
-------------------- Stage_2 (1.0 Pts) --------------------
Expression # 0 Value: 9
Expression # 1 Value: 11
Expression # 2 Value: 1
Expression # 3 Value: 6
Expression # 4 Value: 20
Expression # 5 Value: 8
Expression # 6 Value: 3.5
Expression # 7 Value: 1
-------------------- Stage_3 (2.0 Pts) --------------------
Expression: 1 + 2 = 3
Expression: 2 * 3 = 6
Expression: 4 - 1 = 1
Expression: 6 / 2 = 3

You Got 3 Out Of 4 Right Answers
-------------------- Stage_4 (2.0 Pts) --------------------
-------------------- Stage_5 (1.0 Pts) --------------------
```

```
---------- Stage_1 (2.0 Pts) ----------
---------- Stage_2 (1.0 Pts) ----------
---------- Stage_3 (2.0 Pts) ----------
---------- Stage_4 (2.0 Pts) ----------
A2: Expression: 1 + 1 =  2
A3: Expression: 4 + 7 = 11
M2: Expression: 1 * 5 =  5
M4: Expression: 3 * 9 = 27
A5: Expression: 1 + 2 =  3
D2: Expression: 4 / 2 =  2
D4: Expression: 9 / 2 = 4.5
A4: Expression: 2.4 + 1.4 = 3.8
S3: Expression: (2.9,1.3) + (7.4,9) = (10.3,10.3)
S2: Expression: (5.4,2) - (1.1,4) = (4.3,-2)

Expression # 0 Value: 9
Expression # 1 Value: 11
Expression # 2 Value: 1
Expression # 3 Value: 6
Expression # 4 Value: 20
Expression # 5 Value: 8
Expression # 6 Value: 3
Expression # 7 Value: 1

Expression: 1.4 + 2.1 = 3.5
Expression: 2.9 * 3 = 8.0
Expression: 4.4 - 1.3 = 3.0
Expression: 6.7 / 2.7 = 5

You Got 1 Out Of 4 Right Answers
---------- Stage_5 (1.0 Pts) ----------
D5: Expression: 9 / 2 =  4
Division By Zero Exception
```