CSCE 240H – Project 1 – Option 1

Project Purpose

You will create a C++ program that prompts the user to evaluate randomly generated arithmetic expressions involving exponentiation (^), multiplication (*), division (/), modular division (%), addition (+), and subtraction. All operands should be randomly generated non-negative, one-digit integers (0-9).

The program should ensure that the expressions do not require division by zero.

The program should also ensure that for all division with the / operator, the divisor evenly divides the dividend (e.g. 8 / 2 is ok, 8 / 3 is not).

For each of the user's answers, the program should tell the user whether or not their answer was correct. If the user answered incorrectly, the program should show a step-by-step evaluation of the expression.

The number of expressions the user evaluates is determined by the user.

The program should output the amount of time that passed between the output of the first question to the user's answer to the last question.

Program output should be user friendly and grammatically correct.

A sample executable to meet these initial, minimal requirements is included in the zip file. Once you unzip the file, you can run the executable at your Linux terminal by entering **./sampleprogram**

Some Headers and Functions You Should Include and Use

Header: cstdlib. Functions: srand, rand

Header: ctime. Function: time – time(0) returns the current time in seconds from the system clock

Example:

```
#include<iostream>
#include<cstdlib>
#include<ctime>
int main() {
  srand(time(0));  // seeds the rand function with the current time
                   // from the system clock so that a different list
                   // of values will be generated when the program
                   // is run

  std::cout << "Here's a random number: " << rand() << std::endl;
  std::cout << "Did you like that number? ";

  char answer;
  int start = time();
  std::cin >> answer;
  std::cout << "it took you " << time(0) - start
            << " seconds to answer me!" << std::endl;
  return 0;
}
```

What You'll Turn In

You will submit a zip file that includes:

1. a PDF with your responses to the following prompts (these can be answered by each partner separately or you can both submit the same comments)
   - Comment on the design of your project (e.g. what functions did you write? what files did you create? what libraries did you include? etc)
   - Comment on the development process (i.e. how did you and your partner work together to complete and test your project?)

- Comment on any addition features/improvements implemented in your program above the minimum requirements illustrated in sample executable (e.g. allow user to modify range of allowable operands, resulting values, number of operators, operations used...)

2. a makefile and all of the source and header files you've written to compile, link, and run the program.

Grading

Comments on the program design – 1 point
Comments on the development process – 1 point
Project meets minimum requirements (i.e. runs at least as well as sample) – 9 points
Improvements above minimum requirements – 2 points

Additional Specifications

- You are only to work with your one assigned partner on this project.
- You and your partner should both submit the project in Blackboard.
- All project code must compile and run on a computer of the instructor's choosing in the Linux lab.
- No late project submissions will be accepted.