For original.cpp:

* When one of the input values is negative, one negative (or more than one negative number) result may appear, which doesn’t make sense. The reason behind is that int values in C++ can be negative.
* When the second input value or the third input value is larger than the first input, the result will be larger than 100%. The expected results are smaller or qual to 100%. There is no restriction on the magnitude of values in the program.
* When the second and the third input values are equal, the result is “More students like being at school in person than doing school at home”, which is caused by a logical flaw. The program doesn’t encounter the possibility that the last two input values can be equal. As these two values are qual, the program executes the else statement.
* When input value is smaller than −2147483648 or larger than 2147483647 the result, the program will terminate unexpectedly. The reason behind is that for int values C++ use 4 bytes, one bit is for sign (positive or negative). So, int values should in the range [-2^31, 2^31-1].

For logic\_error.cpp

* I changed the “>” in comparison into “<” so that the logic behind this program is incorrect. When the percentage of people prefer in person is larger than the percentage of people like at home, the output will be more students like doing school from home than being at school in person. I’ve made that type of error before.
* I also changed 100.0 in pctLikeInPerson and pctLikeAtHome to 100. This may cause a precision error. When all calculations involved use integers, the result will also be an integer. Although the program specifies precision, the result will be an integer with “.0” at the end. “=” in C++ is assign. When a double variable is assigned to an int variable, the double variable will be an integer.

For compile\_error.cpp

* I deleted a “;” at end of a variable declaration. This causes the compiler report that a “;” is expected at the end. This could be a common mistake for people (like me) who are not very familiar with C++.
* I deleted a set of “()” after “if”. This results in complier error that “()” is required and the expression is needed after “else”. In some programing languages, “()” is not required.