1. Input for Step 5:

When you input an even number (e.g. 100) for total number of people surveyed, and then split the vote evenly between for and against impeachment (50 for and 50 against in this case), the program will output that “More people oppose impeachment than support it,” which is incorrect. There are the same amount of people opposing impeachment as supporting it.

When inputting a negative integer for number of people surveyed and positive integers for supporting and against, the program outputs negative percentages and overall doesn’t make sense even though it continues running.

Similarly, when inputting a positive integer for number of people surveyed but negative integers for supporting or opposing, the program will also output negative percentages.

The program doesn’t check to see if the total amount supporting and opposing adds up to more than the number of people surveyed. So, you could input a number of people supporting and opposing whose sum is greater than the total people surveyed, and you would end up with a total percent that is greater than 100%. For example, inputting 10 people surveyed, with 7 people supporting impeachment and 8 people opposing impeachment would give an output saying 70% support impeachment and 80% oppose impeachment. This doesn’t really make sense.

2. Logic error introduced in logic\_error.cpp

Simply switching the greater-than sign in the if statement to a less-than sign causes the program to definitely produce illogical output. It is a simple error that someone might make which would cause the program to say that more people support impeachment than oppose it when there is more opposition, and vice versa.

The beginning of the if statement looks like this: **if** (forImpeachment < antiImpeachment)

3. Compilation errors in compile\_error.cpp

Removing a semicolon at the end of a line causes the build to fail. The error message is: “Expected ‘;’ after expression.”

Not defining a variable type before the variable name causes the error “Use of undeclared identifier ‘varName.’” E.g. removing double before pctFor would cause the program to say “Use of undeclared identifier ‘pctFor’.”

A missing closing curly bracket ‘}’ at the end of main causes a build failure and the error message is “Expected ‘}’.”

A missing opening curly bracket ‘{‘ for the main function causes a large number of successive errors such as “Expected unqualified-id”, “Unknown type name ‘cout’”, and “Unknown type name ‘cin’”. The program will not compile this way.