Programming for Engineers Portfolio : Quarter 1, Set 2, es1219

These portfolio exercises move more onto the programming skills, though still remain linked to creating programs within the Linux/Ubuntu environment.

1. The source file `e1-x-plus-1.cpp` reads a single integer and then prints it back out. Modify the source file so that it prints out the input number plus `1`.
2. The source file `e2-x-times-y.cpp` reads two integers and prints out their product. Modify the program so that it reads in two floats and prints their product.
3. The source file `e3-exp2-x.cpp` reads a single float and prints it back out. Modify the program so that it prints out `exp2(x)` instead.
4. Create a source file called `e4-choose.cpp` which reads three integers a, b, and c. If $a \neq 0$ the program should print value b, otherwise it should print c.
5. The source file `e5-error.cpp` is supposed to print $x + 10$ if $x < 10$, and $x \times 5$ otherwise. Currently it does not compile due to an error. Create a copy of the file called `e5-fixed.cpp` and fix the error.
6. The source file `e6-crash.cpp` reads in input x and then prints out $148/(x-17)$. Currently it crashes if given the input 17. Modify it so that in this special case it prints the string `error` (with no quotation marks) to `cout` instead.
7. Create a source file `e7-repeat-39.cpp` which reads a string from `stdin`, and then prints the input string back out 39 times, with each output on a different line. (You can use loops or multiple statements - up to you).
8. The source file `e8-add-strings.cpp` reads strings `x` and `y` from `cin`, and then writes a string variable `x_plus_y` to `cout`. Modify the program so that if the user inputs two decimal integers then the output will be the sum of those integers, **without** changing the types of `x`, `y`, and `x_plus_y`. Hint: you can use the functions `to_string` and `stoi` to convert integers to and from strings.
9. Write a program `e9-polynomial.cpp` which reads a float x from `cin`, and then prints the value of the polynomial $-6+3x+6x^2$ to `cout`. For example, if the input is `2` the output should be `24`, and if the input is `4`, the output should be `102`.
10. Create a program `e10-numbers-from-699-to-10000.cpp` which prints the integers from 699 to 10000, with one number per line.