Unfortunately, my code is not running. However, I feel like I have a decent understanding of how stack-based calculators are supposed to work; I'm just struggling to get the code right. I understand the concept of last-in-first-out and that the operation in front tells you how many inputs you need to pop out after. If you see addition or subtraction, you already know that you need to pop out two numbers. If, after your operation, you need another operation, pop another input. For example, if the inputs are +55, then you need to only pop the inputs after the addition sign. If it were ++55, then you would need to keep popping variables after popping two until you have popped enough variables to complete the operation. Then you can continue adding operations, such as +55/2. After adding the first two 5's then you must divide by two. You can know that you will only need to pop once after getting the division symbol because you already have a number. Of course, unless you get another operation right after the division, you would need to pop again. When using a stack, the order of operations does not matter. It simply only matters what variable was last in; that is what you will use.

I think where I went wrong in my code was something in my header. When I attempted to compile, I kept getting an error that functions were not defined. I thought that I had included all the library inputs, but I must be missing something.