1. Additive Persistence Round 2

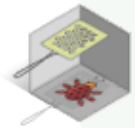
In a recent lab, you were tasked with writing a program that calculated the additive root of a number. This was calculated by adding the sum of each of the digits in a number. If this sum was a single digit long, that was the root, otherwise you needed to repeat the process. For this homework problem you need to do much the same, except instead of calculating the sum of each digit, you need to tally the sum of each pair of digits to calculate the **paired additive root**.

For instance, for the number `9857843` isn't a paired additive root because it has more than 2 digits, you you need to calculate the sum of paired digits which is:

```
9+85+78+43 -> 215
```

Please note that the left most digit will be alone if the number has an odd number of digits. Unfortunately, `215` isn't a paired additive root either (too many digits). So to repeat: `2+15 -> 17`. Thus `17` is the paired additive root of `9857843`.

You need to write a program in `PairedAdditiveRoot/main.cpp` that takes in positive integers until the End-Of-File is encountered and returns the paired additive root of each. I've already written the IO part of the code for you (please note the use of `cin` in the conditional to check if there is more input to be read). You are welcome to modify this starter code however you like.



If you are having trouble

If the test cases aren't working, please ensure that you are compiling and running your code from the terminal so you can see where your code is misbehaving. Also, please be sure to add debugging print statements (like what each two digit pair is) so you can diagnose potential problems.