1: Describe a test plan for the various implementations of calc. Given that the program cannot be tested on every possible input, what are three examples of tests that are implied by the spec but not checked by basic arithmetic.sh?

In addition to basic integer addition, I will test negative numbers, letters, decimal numbers, single input, triple input, and very large numbers.

2: You may have noticed that the spec doesn't say anything about what to do if the user supplies too many arguments at the command line. Unfortunately, ambiguity in specifications is very common in practice and can be very confusing for programmers. If you were asked to implement calc given this spec, you would be forced to choose from among at least three interpretations of what calc should do when called with, say, three arguments 3, 4, and 5:

• Calc should sum *all* of the arguments, and print 12.

• Calc should ignore the third argument, and print the sum of the first two: 7.

• Calc should print an error message and return non-zero.

Luckily for you, you are not the programmer (for now). Should your test scripts check what the program does when given more than two arguments? Why or why not?

The test script will not check the validity when given more than 2 arguments as the calc program is written to take 2 inputs only. Upon testing, it will simply print NOT ENOUGH INPUT.