Problem 1

Implement integer division without using the division operator. That is, given a numerator (dividend) and denominator (divisor), return the quotient and remainder, or return an error when the denominator is zero. Either or both of the inputs may be negative. The remainder should have the same sign as the denominator. Of course, the absolute value of the remainder should be less than that of the denominator.

For example:

Num.	Denom.	Quot.	Rem.
6	5	1	1
6	-5	-2	-4
-6	5	-2	4
-6	-5	1	-1

The solution, in broad strokes

Basically, use binary search to find the quotient.

That is, use the numerator and denominator to bracket the possible range for the quotient, then use binary search within that range to determine the quotient that, when multiplied by the given denominator, brings you closest to the numerator without going over (or under, for negative denominators).

Your solution should be written in one of the following languages (presented in alphabetical order – all are equally acceptable!): C++, Java, Javascript, Python, or Ruby. It should be possible to run automated tests as well as manually invoke your code for user-supplied <numerator, denominator> pairs.

Evaluation

Your solution will be evaluated for its correctness, of course, but also along the following dimensions:

- style (which should be internally consistent at least, and ideally idiomatic given your language of choice)
- commenting and readability
- test coverage and design

Ultimately, we're looking for code quality that's suitable for submission to a shared source control system; nothing more and nothing less. In all, the solution is intended to take an hour or two: no need to go overboard.

As with real code, longer solutions are not necessarily better.

Problem 2

Build a simplified single-page version of something like the Twitter Home page, which should:

- Have an input box + submit button at the top that allows a user to enter new status messages

- Upon submission, insert the latest message at the top of an area below the input box (the user should be able to enter as many messages as they want)

- Restrict the user input to 140 characters

- Show the user's avatar, message, timestamp, and a few action links (like Favorite and Reply) on each message - the timestamp and links don't have to do anything or be dynamic, they can just be styled text, and the avatar is obviously just static

- Be a single page with no back end - refreshing the page obviously clears everything out

- Be styled as if you were going to deploy this to Twitter or Facebook - pretend you're building this as if it would be shown on a live site immediately to millions of users. The interaction and design are an important parts of the exercise, since we're looking for product engineers that can build production-ready features

The exercise is fairly open-ended, as we want to get an idea of your ability to create something great without always having detailed design input. Feel free to use any tools you want (jQuery, SASS, etc.), and it doesn't need to support IE or anything. The deliverable should be a zip file with the page and any assets.