Problem 1b

For this problem we can use Euclid's method for finding the greatest common divisor, but before that we can use a couple of if statements for the special cases where numer = 0 and numer = denom. To use Euclid's algorithm we need to obtain the residue and stop once this becomes 0, so we need to use the modulo operator and a while loop.

We are using two if else statements to differentiate when numer > denom or vice versa. This is because the algorithm requires us to take the larger number and set it equal to the smaller number times a quotient and a residue. We then need to iterate until the residue reaches 0. In each iteration the smaller number takes the place of the larger number and the residue takes the place of the smaller number. This is why we have two variables starting_numer and starting_denom at the beginning because I am inevitably going to override the numer and denom variables. Once the while loop finishes we can get the GCD by using the previous residue before it became 0.

Problem 2c

For this problem we first use an assert to make sure that the amount requested is not negative and then we use another assert to make sure that the total money in the ATM is greater than or equal to the request amount.

Then we use and if else statement to differentiate between the cases where the ATM has twenty-dollar bills and the case where it doesn't have those bills. This guarantees that we can still give money if there are no twenty-bills, but enough ten-bills and also makes sure that we give the least amount of bills possible.

Finally, we invoke the withdraw() function that updates the status of the bank account.