Euclid’s Algorithm

How can you find the greatest common divisor of two numbers without knowing the factors? These days it is pretty easy because you can always use the GCD function in Excel but prior to computers, you had to rely on other means. Interestingly, the great Greek mathematician Euclid provided a solution to this problem more than 2000 years ago. This solution is come to be known as the *Euclidean algorithm*, or simply *Euclid’s algorithm*. Here is how it works.

Consider any two numbers b and c greater than zero. In this instance, there exist unique numbers m and r with $0 \leq r < c$ such that $b = m \cdot c + r$. If b is a multiple of c then $r = 0$ and m is the multiple. If b is not a multiple of c then the remainder is positive, and m is the largest number of times that c could be subtracted from b and still leave a positive remainder (of r). Note that when $b < c$, $m = 0$ and $r = b$.

Put another way, if we consider b/c in decimal form the number can be thought of as having two parts, the stuff to the right of the decimal point and the stuff to the left. The stuff to the left is the integer part of this fraction, m, and the stuff to the right (typically called the fractional (or decimal) part) once multiplied by c is r. When $b < c$ we typically call the fraction b/c a *proper* fraction and if $b > c$ the fraction b/c is said to be an *improper* fraction.

Since one of the two must be larger, let that be b. Euclid noted that the GCD between b and c must also be the GCD between c and r. The same goes for further iterations of this same idea. Each time, the numbers become smaller and easier to compare. Finally, we will end up with a zero remainder of these comparisons. **The last positive remainder before this happens is the GCD(*b*, *c*).** Here are two examples.

The left and right columns show the iterative process; the middle shows the underlying structure.

First example: b = 1155 and c = 308.		*Large* = Multiple·*Small* plus *Remainder*	Second example: b = 1155 and c = 307.
1155 = 3·308 + 231	Iteration 1	$b = m_1 \cdot c + r_1$	1155 = 3·307 + 234
308 = 1·231 + **77**	Iteration 2	$c = m_2 \cdot r_1 + r_2$	307 = 1·234 + 73
231 = 3·77 + 0	Iteration 3	$r_1 = m_3 \cdot r_2 + r_3$	234 = 3·73 + 15
	Iteration 4	$r_2 = m_4 \cdot r_3 + r_4$	73 = 4·15 + 13
	Iteration 5	$r_3 = m_5 \cdot r_4 + r_5$	15 = 1·13 + 2
	Iteration 6	$r_4 = m_6 \cdot r_5 + r_6$	13 = 6·2 + **1**
	Iteration 7	$r_5 = m_7 \cdot r_6 + r_7$	2 = 2·1 + 0

Each iteration reduces the size of the numbers being compared. The *Small* number from one iteration becomes the *Large* number for the next and the *Remainder* from one iteration becomes the *Small* number for the next. This process continues until a remainder of zero has been achieved.

In the first instance, GCD(1155, 308) = 77 in three iterations. In the second instance, GCD(1155, 307) = 1 in seven iterations. When GCD = 1 we say the numbers are relatively prime or coprime.

In both instances, these conclusions were obtained without factoring b or c. All that is necessary is simple division with remainders noted.

Even when the numbers are reasonably large, this process converges pretty quickly. Consider for example the somewhat larger numbers b = 1,453,568 and c = 280,137. You can check that these numbers are coprime in 7 iterations, just like the second example above.