Divisibility and the Greatest Common Divisor

Divisibility and the Greatest Common Divisor

The notions of divisibility and factorizations are important tools in number theory. In this chapter we will look at these ideas more closely.

Suppose that m and n are integers with $m \neq 0$. We say that m *divides* n if n is a multiple of m, that is, if there is an integer k such that $n = mk$. If m divides n, we write $m|n$. Similarly, if m does not divide n, then we write $m \nmid n$. For example,

$$3|6 \quad \text{and} \quad 12|132, \quad \text{since} \quad 6 = 3 \cdot 2 \quad \text{and} \quad 132 = 12 \cdot 11.$$

The divisors of 6 are 1, 2, 3, and 6. On the other hand, $5 \nmid 7$, since no integer multiple of 5 is equal to 7. A number that divides n is called a *divisor of* n.

If we are given two numbers, we can look for common divisors, that is, numbers that divide both of them. For example, 4 is a common divisor of 12 and 20, since $4|12$ and $4|20$. Notice that 4 is the largest common divisor of 12 and 20. Similarly, 3 is a common divisor of 18 and 30, but it is not the largest, since 6 is also a common divisor. The largest common divisor of two numbers is an extremely important quantity that will frequently appear during our number theoretic excursions.

> The *greatest common divisor* of two numbers a and b (not both zero) is the largest number that divides both of them. It is denoted by $\gcd(a, b)$. If $\gcd(a, b) = 1$, we say that a and b are *relatively prime*.

Two examples that we mentioned above are

$$\gcd(12, 20) = 4 \qquad \text{and} \qquad \gcd(18, 30) = 6.$$

Another example is

$$\gcd(225, 120) = 15.$$

We can check that this answer is correct by factoring $225 = 3^2 \cdot 5^2$ and $120 = 2^3 \cdot 3 \cdot 5$, but, in general, factoring a and b is not an efficient way to compute their greatest common divisor.[1]

The most efficient method known for finding the greatest common divisors of two numbers is called the *Euclidean algorithm*. It consists of doing a sequence of divisions with remainder until the remainder is zero. We will illustrate with two examples before describing the general method.

As our first example, we will compute $\gcd(36, 132)$. The first step is to divide 132 by 36, which gives a quotient of 3 and a remainder of 24. We write this as

$$132 = 3 \times 36 + 24.$$

The next step is to take 36 and divide it by the remainder 24 from the previous step. This gives

$$36 = 1 \times 24 + 12.$$

Next we divide 24 by 12, and we find a remainder of 0,

$$24 = 2 \times 12 + 0.$$

The Euclidean algorithm says that as soon as you get a remainder of 0, the remainder from the previous step is the greatest common divisor of the original two numbers. So in this case we find that $\gcd(132, 36) = 12$.

Let's do a larger example. We will compute

$$\gcd(1160718174, 316258250).$$

Our reason for doing a large example like this is to help convince you that the Euclidean algorithm gives a far more efficient way to compute gcd's than factorization. We begin by dividing 1160718174 by 316258250, which gives 3 with a remainder of 211943424. Next we take 316258250 and divide it by 211943424. This process continues until we get a remainder of 0. The calculations are given in

[1] An even less efficient way to compute the greatest common divisor of a and b is the method taught to my daughter by her fourth grade teacher, who recommended that the students make complete lists of all the divisors of a and b and then pick out the largest number that appears on both lists!

the following table:

$$\begin{aligned}
1160718174 &= 3 \times 316258250 + 211943424 \\
316258250 &= 1 \times 211943424 + 104314826 \\
211943424 &= 2 \times 104314826 + 3313772 \\
104314826 &= 31 \times 3313772 + 1587894 \\
3313772 &= 2 \times 1587894 + 137984 \\
1587894 &= 11 \times 137984 + 70070 \\
137984 &= 1 \times 70070 + 67914 \\
70070 &= 1 \times 67914 + 2156 \\
67914 &= 31 \times 2156 + \boxed{1078} \leftarrow \gcd \\
2156 &= 2 \times 1078 + 0
\end{aligned}$$

Notice how at each step we divide a number A by a number B to get a quotient Q and a remainder R. In other words,

$$A = Q \times B + R.$$

Then at the next step we replace our old A and B with the numbers B and R and continue the process until we get a remainder of 0. At that point, the remainder R from the previous step is the greatest common divisor of our original two numbers. So the above calculation shows that

$$\gcd(1160718174, 316258250) = 1078.$$

We can partly check our calculation (always a good idea) by verifying that 1078 is indeed a common divisor. Thus

$$1160718174 = 1078 \times 1076733 \quad \text{and} \quad 316258250 = 1078 \times 293375.$$

There is one more practical matter to be mentioned before we undertake a theoretical analysis of the Euclidean algorithm. If we are given A and B, how can we find the quotient Q and the remainder R? Of course, you can always use long division, but that can be time consuming and subject to arithmetic errors if A and B are large. A pleasant alternative is to find a calculator or computer program that will automatically compute Q and R for you. However, even if you are only equipped with an inexpensive calculator, there is an easy three-step method to find Q and R.

Method to Compute Q and R on a Calculator So That $A = B \times Q + R$

1. Use the calculator to divide A by B. You get a number with decimals.
2. Discard all the digits to the right of the decimal point. This gives Q.
3. To find R, use the formula $R = A - B \times Q$.

For example, suppose that $A = 12345$ and $B = 417$. Then $A/B = 29.6043\ldots$, so $Q = 29$ and $R = 12345 - 417 \cdot 29 = 252$.

We're now ready to analyze the Euclidean algorithm. The general method looks like

$$\begin{aligned}
a &= q_1 \times b + r_1 \\
b &= q_2 \times r_1 + r_2 \\
r_1 &= q_3 \times r_2 + r_3 \\
r_2 &= q_4 \times r_3 + r_4 \\
&\vdots \\
r_{n-3} &= q_{n-1} \times r_{n-2} + r_{n-1} \\
r_{n-2} &= q_n \times r_{n-1} + \boxed{r_n} \leftarrow \gcd \\
r_{n-1} &= q_{n+1} r_n + 0
\end{aligned}$$

If we let $r_0 = b$ and $r_{-1} = a$, then every line looks like

$$r_{i-1} = q_{i+1} \times r_i + r_{i+1}.$$

Why is the last nonzero remainder r_n a common divisor of a and b? We start from the bottom and work our way up. The last line $r_{n-1} = q_{n+1} r_n$ shows that r_n divides r_{n-1}. Then the previous line

$$r_{n-2} = q_n \times r_{n-1} + r_n$$

shows that r_n divides r_{n-2}, since it divides both r_{n-1} and r_n. Now looking at the line above that, we already know that r_n divides both r_{n-1} and r_{n-2}, so we find that r_n also divides r_{n-3}. Moving up line by line, when we reach the second line we will already know that r_n divides r_2 and r_1. Then the second line $b = q_2 \times r_1 + r_2$ tells us that r_n divides b. Finally, we move up to the top line and use the fact that r_n divides both r_1 and b to conclude that r_n also divides a. This completes our verification that the last nonzero remainder r_n is a common divisor of a and b.

But why is r_n the *greatest* common divisor of a and b? Suppose that d is any common divisor of a and b. We will work our way back down the list of equations. So from the first equation $a = q_1 \times b + r_1$ and the fact that d divides both a and b, we see that d also divides r_1. Then the second equation $b = q_2 r_1 + r_2$ shows us that d must divide r_2. Continuing down line by line, at each stage we will know that d divides the previous two remainders r_{i-1} and r_i, and then the current line $r_{i-1} = q_{i+1} \times r_i + r_{i+1}$ will tell us that d also divides the next remainder r_{i+1}. Eventually, we reach the penultimate line $r_{n-2} = q_n \times r_{n-1} + r_n$, at which point we conclude that d divides r_n. So we have shown that if d is any common divisor of a and b then d will divide r_n. Therefore, r_n must be the greatest common divisor of a and b.

This completes our verification that the Euclidean algorithm actually computes the greatest common divisor, a fact of sufficient importance to be officially recorded.

Theorem 1 (Euclidean Algorithm)**.** *To compute the greatest common divisor of two numbers a and b, let $r_{-1} = a$, let $r_0 = b$, and compute successive quotients and remainders*

$$r_{i-1} = q_{i+1} \times r_i + r_{i+1}$$

for $i = 0, 1, 2, \ldots$ until some remainder r_{n+1} is 0. The last nonzero remainder r_n is then the greatest common divisor of a and b.

There remains the question of why the Euclidean algorithm always finishes. In other words, we know that the last nonzero remainder will be the desired $\gcd$, but how do we know that we ever get a remainder that does equal 0? This is not a silly question, since it is easy to give algorithms that do not terminate; and there are even very simple algorithms for which it is not known whether or not they always terminate. Fortunately, it is easy to see that the Euclidean algorithm always terminates. The reason is simple. Each time we compute a quotient with remainder,

$$A = Q \times B + R,$$

the remainder will be between 0 and $B - 1$. This is clear, since if $R \geq B$, then we can add one more onto the quotient Q and subtract B from R. So the successive remainders in the Euclidean algorithm continually decrease:

$$b = r_0 > r_1 > r_2 > r_3 > \cdots .$$

But all the remainders are greater than or equal to 0, so we have a strictly decreasing sequence of nonnegative integers. Eventually, we must reach a remainder that equals 0; in fact, it is clear that we will reach a remainder of 0 in at most b steps. Fortunately, the Euclidean algorithm is far more efficient than this. You will show in the exercises that the number of steps in the Euclidean algorithm is at most seven times the *number of digits* in b. So, on a computer, it is quite feasible to compute $\gcd(a, b)$ when a and b have hundreds or even thousands of digits!

Exercises

1. Use the Euclidean algorithm to compute each of the following $\gcd$'s.
(a) $\gcd(12345, 67890)$ **(b)** $\gcd(54321, 9876)$

2.  Write a program to compute the greatest common divisor $\gcd(a, b)$ of two integers a and b. Your program should work even if one of a or b is zero. Make sure that you don't go into an infinite loop if a and b are both zero!

3. Let $b = r_0, r_1, r_2, \ldots$ be the successive remainders in the Euclidean algorithm applied to a and b. Show that after every two steps, the remainder is reduced by at least one half. In other words, verify that

$$r_{i+2} < \frac{1}{2} r_i \qquad \text{for every } i = 0, 1, 2, \ldots .$$

Conclude that the Euclidean algorithm terminates in at most $2\log_2(b)$ steps, where $\log_2$ is the logarithm to the base 2. In particular, show that the number of steps is at most seven times the number of digits in b. [*Hint*. What is the value of $\log_2(10)$?]

4. A number L is called a common multiple of m and n if both m and n divide L. The smallest such L is called the *least common multiple of* m *and* n and is denoted by $\mathrm{LCM}(m, n)$. For example, $\mathrm{LCM}(3, 7) = 21$ and $\mathrm{LCM}(12, 66) = 132$.

(a) Find the following least common multiples.
(i) $\mathrm{LCM}(8, 12)$ (ii) $\mathrm{LCM}(20, 30)$ (iii) $\mathrm{LCM}(51, 68)$ (iv) $\mathrm{LCM}(23, 18)$.

(b) For each of the LCMs that you computed in (a), compare the value of $\mathrm{LCM}(m, n)$ to the values of m, n, and $\gcd(m, n)$. Try to find a relationship.

(c) Give an argument proving that the relationship you found is correct for all m and n.

(d) Use your result in (b) to compute $\mathrm{LCM}(301337, 307829)$.

(e) Suppose that $\gcd(m, n) = 18$ and $\mathrm{LCM}(m, n) = 720$. Find m and n. Is there more than one possibility? If so, find all of them.

5. The "$3n + 1$ *algorithm*" works as follows. Start with any number n. If n is even, divide it by 2. If n is odd, replace it with $3n + 1$. Repeat. So, for example, if we start with 5, we get the list of numbers

$$5, 16, 8, 4, 2, 1, 4, 2, 1, 4, 2, 1, \ldots ,$$

and if we start with 7, we get

$$7, 22, 11, 34, 17, 52, 26, 13, 40, 20, 10, 5, 16, 8, 4, 2, 1, 4, 2, 1, \ldots .$$

Notice that if we ever get to 1 the list just continues to repeat with $4, 2, 1$'s. In general, one of the following two possibilities will occur:[2]

(i) We may end up repeating some number a that appeared earlier in our list, in which case the block of numbers between the two a's will repeat indefinitely. In this case we say that the algorithm *terminates* at the last nonrepeated value, and the number of distinct entries in the list is called the *length of the algorithm*. For example, the algorithm terminates at 1 for both 5 and 7. The length of the algorithm for 5 is 6, and the length of the algorithm for 7 is 17.

(ii) We may never repeat the same number, in which case we say that the algorithm does not terminate.

[2]There is, of course, a third possibility. We may get tired of computing and just stop working, in which case one might say that the algorithm terminates due to exhaustion of the computer!

(a) Find the length and terminating value of the $3n+1$ algorithm for each of the following starting values of n:

(i) $n = 21$ (ii) $n = 13$ (iii) $n = 31$

(b) Do some further experimentation and try to decide whether the $3n + 1$ algorithm always terminates and, if so, at what value(s) it terminates.

(c) Assuming that the algorithm terminates at 1, let $L(n)$ be the length of the algorithm for starting value n. For example, $L(5) = 6$ and $L(7) = 17$. Show that if $n = 8k+4$ with $k \geq 1$, then $L(n) = L(n+1)$. [*Hint*. What does the algorithm do to the starting values $8k + 4$ and $8k + 5$?]

(d) Show that if $n = 128k + 28$ then $L(n) = L(n + 1) = L(n + 2)$.

(e) Find some other conditions, similar to those in (c) and (d), for which consecutive values of n have the same length. (It might be helpful to begin by using the next exercise to accumulate some data.)

6. 💻 Write a program to implement the $3n + 1$ algorithm described in the previous exercise. The user will input n and your program should return the length $L(n)$ and the terminating value $T(n)$ of the $3n + 1$ algorithm. Use your program to create a table giving the length and terminating value for all starting values $1 \leq n \leq 100$.