The Euclidean Algorithm

The Euclidean algorithm is one of the oldest known algorithms (it appears in Euclid's *Elements*) yet it is also one of the most important, even today.

Not only is it fundamental in mathematics, but it also has important applications in computer security and cryptography.

The algorithm provides an extremely fast method to compute the greatest common divisor (gcd) of two integers.

Definition. Let a, b be two integers. A *common divisor* of the pair a, b is any integer d such that $d \mid a$ and $d \mid b$.

Reminder: To say that $d \mid a$ means that $\exists c \in \mathbb{Z}$ such that $a = d \cdot c$. I.e., to say that $d \mid a$ means that a is an integral multiple of d.

Example. The common divisors of the pair $12, 150$ include $\pm 1, \pm 2, \pm 3, \pm 6$. These are ALL the common divisors of this pair of integers.

Question: How can we be sure there aren't any others?

- Divisors of 12 are $\pm 1, \pm 2, \pm 3, \pm 4, \pm 6, \pm 12$ and no others.
- Divisors of 150 are $\pm 1, \pm 2, \pm 3$, ± 5, ± 6, ± 10, ± 15, ± 25, ± 30, ± 50, ± 75, ± 150 and no others.
- Now take the intersection of the two sets to get the common divisors.

Definition. The *greatest common divisor* (written as $\gcd(a, b)$) of a pair a, b of integers is the biggest of the common divisors.

In other words, the greatest common divisor of the pair a, b is the maximum element of the set of common divisors of a, b.

Example. From our previous example, we know the set of common divisors of the pair $12, 150$ is the set $\{\pm 1, \pm 2, \pm 3, \pm 6\}$. Thus, $\gcd(12, 150) = 6$, since 6 is the maximum element of the set.

The $\gcd(a, b)$ always exists, *except* in one case: $\gcd(0, 0)$ is undefined. Why?

Because any positive integer is a common divisor of the pair $0, 0$ *and the set of positive integers has no maximum element.*

Why is the gcd defined for every other pair of integers?

- Hint: Can you prove that if at least one of the integers a, b is non-zero, then the set of common divisors has an upper bound?
- Why is that enough to prove the claim?
- How is the existence of said maximum related to the well-ordering principle, if it is?

If you can't figure out the answers to these questions, then you don't understand the definitions yet!

TEST: What is $\gcd(a, 0)$ for any integer $a \neq 0$?

COMMENT: Rosen defines $\gcd(0, 0) = 0$. Do you think that is reasonable? What is wrong, if anything, with allowing $\gcd(0, 0)$ to be undefined? Would defining $\gcd(0, 0) = \infty$ be more reasonable?

The following observation means that we may as well confine our attention to pairs of non-negative integers when we study the gcd.

Lemma. *For any integers a, b we have* $\gcd(a, b) = \gcd(|a|, |b|)$.

The proof is left as an exercise for you. Here's a hint: How does the list of divisors of a differ from that of $|a|$?.

At this point, we have an infallible method for computing the gcd of a given pair of numbers:

1. Find the set of positive divisors of each number. (Why is it enough to find just the positive divisors?)
2. Find the intersection of the two sets computed in the previous step.
3. The maximum element of the intersection is the desired gcd.

How efficient is this method? How long do you think it would take to compute all the positive divisors of a larger number such as $a = 1092784930198374849278478587371$?

For large numbers a, we would essentially be forced to try dividing by each number up to the square root of a, in the worst case. (The worst case turns out to be the case where a is prime — we will say more about primes later.)

Suppose that a has 200 decimal digits. Then $10^{199} \leq a < 10^{200}$, so $3 \cdot 10^{99} < \sqrt{a} < 10^{100}$. Dividing by every number up to the square root would involve doing at least $3 \cdot 10^{99}$ divisions.

Suppose we use a supercomputer that can do a billion (10^9) divisions per second. Then the number of seconds it would take the supercomputer to do all the needed divisions (in the worst case) would be at least

$$3 \cdot 10^{99}/10^9 = 3 \cdot 10^{90} \text{ seconds.}$$

How many seconds is that? Well, there are $60 \cdot 60 \cdot 24$ seconds in a day, and $60 \cdot 60 \cdot 24 \cdot 365 = 31536000$ seconds in a year. That's roughly $3.2 \cdot 10^8$ seconds per year. So the number of years it would take the supercomputer to do all the needed divisions (in the worst case) would be at least

$$3 \cdot 10^{90}/(3.2 \cdot 10^8) = 9.375 \cdot 10^{81} \text{ years.}$$

This is rather alarming, once you look up the age of the universe: 14.6 billion years.

CONCLUSION: *It would take MUCH longer than the age of the universe for a fast supercomputer to perform that many divisions!!*

Nevertheless, I can find the gcd of a pair a 200 digit numbers on my Macbook (which is NOT a supercomputer) in a couple of seconds.

THERE MUST BE A BETTER METHOD THAN MAKING LISTS OF DIVISORS!

The better method is called the Euclidean algorithm, of course. It is based on the division algorithm. Let's see how it works on a small example.

Example (Find $\gcd(10319, 2312)$)**.** Divide 10319 by 2312: $10319 = 4 \cdot 2312 + 1071$.

Divide 2312 by 1071: $2312 = 2 \cdot 1071 + 170$.

Divide 1071 by 170: $1071 = 6 \cdot 170 + 51$.

Divide 170 by 51: $170 = 3 \cdot 51 + 17$.

Divide 51 by 17: $51 = 3 \cdot 17 + 0$ *STOP!*

CONCLUSION: $\gcd(10319, 2312) = 17$ (the last non-zero remainder).

In the example, we found the gcd with just five divisions. Try making lists of divisors of the two numbers to compute the gcd. We stopped when we did because we had to: the next step would involve division by zero!

Theorem (Euclidean algorithm)**.** *Given positive integers a, b with $a \geq b$. Put $r_0 = a$ and $r_1 = b$. For each $j \geq 0$, apply the division algorithm to divide r_j by r_{j+1} to obtain an integer quotient q_{j+1} and remainder r_{j+2}, so that:*

$$r_j = r_{j+1}q_{j+1} + r_{j+2} \text{ with } 0 \leq r_{j+2} < r_{j+1}.$$

This process terminates when a remainder of 0 is reached, and the last non-zero remainder in the process is $\gcd(a, b)$.

The proof requires a small lemma, which we state and prove first.

Lemma. *Given integers d, e such that $e = dq + r$, where q, r are integers, we have that $\gcd(e, d) = \gcd(d, r)$.*

Proof. Let c be *any* common divisor of the pair d, e. Then c must divide the left hand side of $e - dr = r$, so c must divide r. Thus c is a common divisor of the pair d, r.

On the other hand, let c be *any* common divisor of the pair d, r. Then c divides the right hand side of $e = dq + r$, so c divides e. Thus c is a common divisor of the pair d, e.

This shows that the pair e, d and the pair d, r have the same set of common divisors. It follows that the maximum is the same, too, in other words, $\gcd(e, d) = \gcd(d, r)$. □

Now we can prove the theorem:

Proof. By the lemma, we have that at each stage of the Euclidean algorithm, $\gcd(r_j, r_{j+1}) = \gcd(r_{j+1}, r_{j+2})$. The process in the Euclidean algorithm produces a strictly decreasing sequence of remainders $r_0 > r_1 > r_2 > \cdots > r_{n+1} = 0$. This sequence must terminate with some remainder equal to zero since as long as the remainder is positive the process could be continued.

If r_n is the last non-zero remainder in the process, then we have

$$r_n = \gcd(r_n, 0) = \gcd(r_{n-1}, r_n) = \cdots = \gcd(r_0, r_1) = \gcd(a, b).$$

Each successive pair of remainders in the process is the same. The proof is complete. □

We can prove more. Let $g = \gcd(a, b) = r_n$. Solving for the remainder r_n in the last equation $r_{n-2} = r_{n-1}q_{n-1} + r_n$ with non-zero remainder gives us that

$$g = r_n = r_{n-2} - r_{n-1}q_{n-1}$$

which shows that g can be expressed as a linear combination of the two preceding remainders in the sequence of remainders. By backwards induction, this is true at each step along the way, all the way back to the pair $r_0 = a$, $r_1 = b$. For instance, since $r_{n-1} = r_{n-3} - r_{n-2}q_{n-2}$ by substituting into the above equation we get

$$\begin{aligned} g &= r_{n-2} - r_{n-1}q_{n-1} = r_{n-2} - (r_{n-3} - r_{n-2}q_{n-2})q_{n-1} \\ &= q_{n-1}r_{n-3} + (1 + q_{n-2}q_{n-1})r_{n-2}, \end{aligned}$$

which is another linear combination, as claimed.

This analysis proves the following result, and it also provides a method for finding such a linear combination.

Theorem (Bezout's theorem)**.** *Let $g = \gcd(a, b)$ where a, b are positive integers. Then there are integers x, y such that $g = ax + by$.*

In other words, the gcd of the pair a, b is always expressible as some integral linear combination of a, b. By substituting backwards successively in the equations from the Euclidean algorithm, we can always *find* such a linear combination.

Example ($\gcd(10319, 2312) = 17$ revisited)**.** We want to find integers x, y such that $17 = 10319x + 2312y$. Let's recall that when we computed this gcd earlier in this lecture, we got $10319, 2312, 1071, 170, 51, 17, 0$ for the sequence of remainders. So $r_0 = 10319$, $r_1 = 2312$, $r_2 = 1071$, $r_3 = 170$, $r_4 = 51$, $r_5 = 17$, and $r_6 = 0$. The equations we got before, written in *reverse* order, are in the first column below, and the calculation of x, y is shown in the second column:

$$\begin{aligned} r_3 &= 3r_4 + r_5 && \Rightarrow 17 = r_5 = r_3 - 3r_4 \\ r_2 &= 6r_3 + r_4 && \Rightarrow 17 = r_3 - 3(r_2 - 6r_3) = -3r_2 + 19r_3 \\ r_1 &= 2r_2 + r_3 && \Rightarrow 17 = -3r_2 + 19(r_1 - 2r_2) = 19r_1 - 41r_2 \\ r_0 &= 4r_1 + r_2 && \Rightarrow 17 = 19r_1 - 41(r_0 - 4r_1) = -41r_0 + 183r_1. \end{aligned}$$

Remembering that $r_0 = 10319, r_1 = 2312$ this calculation proves that $17 = (10319)(-41) + (2312)(183)$, so $x = -41$ and $y = 183$.

Theorem. *Let $g = \gcd(a, b)$ where a, b are integers, not both 0. Then g is the least positive integer which is expressible as an integral linear combination of a, b.*

Proof. (Sketch) Let S be the set of all positive integers expressible in the form $ax + by$ for integers x, y. By the well-ordering principle, the set S has a least element, call it d.

Apply the division algorithm to show that $d \mid a$ and $d \mid b$. This shows that d is a common divisor of the pair a, b.

Now assume that c is any other common divisor of the pair a, b. Since d is expressible in the form $ax + by$, you can show that c must divide d. This shows that $c \leq d$. It follows that d is the greatest common divisor, so $d = g$, as desired. □

This theorem implies Bezout's theorem (again). It also gives a new characterization of the gcd.