Given two positive integers $a \ge b > 0$ we seek their *Greatest Common Divisor* (GCD), which is the biggest integer d that divides both a and b leaving no remainder. Ordinary long division computes a positive integer quotient $q := \lfloor a/b \rfloor$ and leaves a remainder $r := a - q \cdot b$ that satisfies $0 \le r < b$. Clearly every divisor of both a and b divides r too, and conversely every divisor of both b and r divides $a = q \cdot b + r$ too; therefore $\mathrm{GCD}(a, b) = \mathrm{GCD}(b, r)$. But the pair (b, r) is *smaller* than the pair (a, b) in the sense that $b \le a$ and $r < b$. This leads to an algorithm …

Euclid's GCD Algorithm

Given integers $a \ge b > 0$, set $r_0 := a$ and $r_1 := b$ and perform successive long divisions getting, for $j = 1, 2, 3, \ldots, n$ in turn until $r_{n+1} = 0$, quotients q_j and remainders r_j that satisfy

$$r_{j-1} = q_j \cdot r_j + r_{j+1} \quad \text{with} \quad 0 \le r_{j+1} < r_j .$$

(Here at step j we divide r_{j-1} by r_j to get quotient q_j and remainder r_{j+1}, stopping when a remainder $r_{n+1} = 0$. At that point $q_n > 1$; can you see why?) The algorithm stops because this decreasing sequence of $n+1$ positive integers, $r_0 = a \ge r_1 = b > r_2 > \ldots > r_{n-1} > r_n > r_{n+1} = 0$, cannot have $n > b$. Then $\mathrm{GCD}(a, b) = r_n$ because, as explained in the first paragraph,

$$\mathrm{GCD}(a, b) =: \mathrm{GCD}(r_0, r_1) = \mathrm{GCD}(r_1, r_2) = \ldots = \mathrm{GCD}(r_{n-1}, r_n) = \mathrm{GCD}(r_n, r_{n+1}) = r_n .$$

The quotients q_j appear to play no important role in the foregoing algorithm, but appearances can mislead. By translating the algorithm's recurrence into matrix language we find uses for q_j :

Set $\begin{bmatrix} r_0 \\ r_1 \end{bmatrix} := \begin{bmatrix} a \\ b \end{bmatrix}$ first; then for $j = 1, 2, 3, \ldots, n$ in turn confirm that $\begin{bmatrix} r_j \\ r_{j+1} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & -q_j \end{bmatrix} \begin{bmatrix} r_{j-1} \\ r_j \end{bmatrix}$, with $0 \le r_{j+1} < r_j$ and $r_{n+1} = 0$, so $\begin{bmatrix} r_n \\ 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & -q_n \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & -q_{n-1} \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & -q_{n-2} \end{bmatrix} \cdots \begin{bmatrix} 0 & 1 \\ 1 & -q_2 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & -q_1 \end{bmatrix} \begin{bmatrix} r_0 \\ r_1 \end{bmatrix}$.

Now set row $\begin{bmatrix} B & A \end{bmatrix} := \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & -q_n \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & -q_{n-1} \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & -q_{n-2} \end{bmatrix} \cdots \begin{bmatrix} 0 & 1 \\ 1 & -q_2 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & -q_1 \end{bmatrix}$ to obtain two integers A and B (not both positive) satisfying $\mathrm{GCD}(a, b) = r_n = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} r_n \\ 0 \end{bmatrix} = \begin{bmatrix} B & A \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = B \cdot a + A \cdot b$.

We have just found that $\mathrm{GCD}(a, b)$ is a linear combination of a and b with integer coefficients, thus proving the following … (Cf. text p. 137, and p. 201 ex. 58.)

Theorem 1: As $\overline{A}$ and $\overline{B}$ run independently through all integers the expression $\overline{B} \cdot a + \overline{A} \cdot b$ runs through a set of integers among which the smallest positive integer is $\mathrm{GCD}(a, b) = B \cdot a + A \cdot b$.

Hard Exercise: Running $\overline{A}$ and $\overline{B}$ through *all* integers is unnecessary: Theorem 1 remains true after restrictions $|\overline{A}| < a$ and $|\overline{B}| \le b \le a$ are imposed; why? Can you prove $|A| < a/\mathrm{GCD}(a, b)$ and $|B| \le b/\mathrm{GCD}(a, b)$? See below.

There are two ways to compute A and B . The easiest is to evaluate from-left-to-right the matrix product defining $\begin{bmatrix} B & A \end{bmatrix}$ *after* all the q_j's have been computed; this gives rise to a recurrence:

$$s_n := 1 ; \quad s_{n-1} := -q_{n-1} ; \quad \text{for } j = n-2, n-3, \ldots, 2, 1 \text{ in turn } \quad s_j := s_{j+2} - q_j \cdot s_{j+1} .$$

Finally $A := s_1$ and $B := s_2$. Another way to compute them is to evaluate from-right-to-left the matrix product defining row $\begin{bmatrix} B & A \end{bmatrix}$ *simultaneously* with the computation of the q_j's :

This document was created with FrameMaker 4.0.4

$\begin{bmatrix} B_0 & A_0 \end{bmatrix} := \begin{bmatrix} 0 & 1 \end{bmatrix}$; $\begin{bmatrix} B_1 & A_1 \end{bmatrix} := \begin{bmatrix} 1 & -q_1 \end{bmatrix}$; for j = 2, 3, …, n–1 in turn $\begin{bmatrix} B_j & A_j \end{bmatrix} := \begin{bmatrix} 1 & -q_j \end{bmatrix} \begin{bmatrix} B_{j-2} & A_{j-2} \\ B_{j-1} & A_{j-1} \end{bmatrix}$.

Finally $\begin{bmatrix} B & A \end{bmatrix} := \begin{bmatrix} B_{n-1} & A_{n-1} \end{bmatrix}$. Note that q_n never figures in the computation of A and B . Whichever way be chosen to compute A, B and GCD(a, b) = B·a + A·b , the algorithm is called "the Extended Euclidean Algorithm" and has important applications. Here is one of them:

Exercise: Given integers a, c and b > 0 , when does " a·x ≡ c **mod** b " have integer solutions x ? Here " p ≡ q **mod** b " is pronounced " p is congruent to q mod b " and means that p–q is divisible by b . Let d := GCD(a, b) . Exhibit all d noncongruent solutions x if and only if d divides c ; otherwise prove no solution x exists.

Continued Fractions

If d = GCD(a, b) then (a/d)/(b/d) exhibits a/b "in lowest terms" but is not the only unique encoding of rational numbers. By substituting $r_{j-1}/r_j = q_j + 1/(r_j/r_{j+1})$ repeatedly for j = 1, 2, …, n in turn we obtain a *Terminating Continued Fraction*

$$\frac{a}{b} = q_1 + \cfrac{1}{q_2 + \cfrac{1}{q_3 + \cfrac{1}{\ldots + \cfrac{1}{q_{n-1} + \cfrac{1}{q_n}}}}} \; .$$

This is *the* continued fraction for the rational number a/b . Here $q_1 \geq 1$ because $a \geq b > 0$; in fact every $q_j \geq 1$ and the last $q_n \geq 2$ to ensure that the encoding of each rational a/b > 1 by a finite sequence $(q_1, q_2, q_3, \ldots, q_{n-1}, q_n-1)$ of positive integers be unique. Euclid's algorithm converts a rational number given as a ratio of integers into its continued fraction; how do we get back? The obvious way evaluates the continued fraction "bottom-up" : $R_{n+1} := 0$; $R_n := 1$; for j = n, n–1, n–2, …, 2, 1 in turn $R_{j-1} := q_j \cdot R_j + R_{j+1}$; finally $a/b = R_0/R_1$ in lowest terms.

Exercise: Confirm that every integer $R_j = r_j/\mathrm{GCD}(a, b)$.

Translating the bottom-up evaluation of the continued fraction into matrix terms yields first $\begin{bmatrix} R_{j-1} \\ R_j \end{bmatrix} = \begin{bmatrix} q_j & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} R_j \\ R_{j+1} \end{bmatrix}$, then $\begin{bmatrix} R_0 \\ R_1 \end{bmatrix} = \begin{bmatrix} q_1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} q_2 & 1 \\ 1 & 0 \end{bmatrix} \ldots \begin{bmatrix} q_{n-1} & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} q_n & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix}$. This last expression offers two interesting opportunities. One is a way to evaluate the continued fraction "top-down" :

$\begin{bmatrix} h_0 \\ g_0 \end{bmatrix} := \begin{bmatrix} 1 \\ 0 \end{bmatrix}$; $\begin{bmatrix} h_1 \\ g_1 \end{bmatrix} := \begin{bmatrix} q_1 \\ 1 \end{bmatrix}$; for j = 2, 3, …, n in turn $\begin{bmatrix} h_j \\ g_j \end{bmatrix} := \begin{bmatrix} h_{j-1} & h_{j-2} \\ g_{j-1} & g_{j-2} \end{bmatrix} \begin{bmatrix} q_j \\ 1 \end{bmatrix}$; finally $\begin{bmatrix} R_0 \\ R_1 \end{bmatrix} := \begin{bmatrix} h_n \\ g_n \end{bmatrix}$.

This top-down evaluation turns out to be a good way to evaluate endless continued fractions that encode non-rational numbers; successive ratios h_j/g_j can be shown to converge alternatingly.

Exercise: The endless continued fraction in which every $q_j = 1$ represents $\mu := (1 + \sqrt{5})/2$; can you see why?

Another opportunity offered by that long matrix product is a clear proof of *Lamé's Theorem* : To compute d = GCD(a, b) for $a \geq b > 0$ Euclid's algorithm needs $n \leq 1 + \ln(b/d)/\ln(\mu)$ divisions.

Exercise: Prove it by showing every R_j is at least as big as if every $q_j = 1$ except $q_n = 2$, so $R_1 \geq f_{n+1}$, a Fibonacci number, and $f_{n+1} = (\mu^{n+1} - (-1/\mu)^{n+1})/(\mu + 1/\mu) \geq \mu^{n-1}$.

(Cf. text p. 206.)

Exercises:

Suppose given integers $M > 1$ and $N > 1$ have $GCD(M, N) = 1 = n \cdot M - m \cdot N$ for some integers m and n whose signs are not yet determined.

1) Show why m and n must have the same nonzero sign.

Henceforth we can assume that $n > 0$ and $m > 0$; otherwise swap M and N , *etc*.

2) What is $GCD(m, n)$?

3) Show how to replace m and n respectively by $\overline{m}$ and $\overline{n}$ satisfying
$$0 < \overline{m} < M , \quad 0 < \overline{n} < N \quad \text{and} \quad 1 = n \cdot M - m \cdot N = \overline{n} \cdot M - \overline{m} \cdot N .$$

Henceforth we can assume that $0 < m < M$ and $0 < n < N$ and $n \cdot M - m \cdot N = 1$. (†)

4) Exhibit instances of pairs (M, N) and (m, n) which satisfy these assumptions (†), but for which $M > N$ in one instance, and $M < N$ in another.

5) Given that the pairs (M, N) and (m, n) satisfy (†), show how to obtain a pair $(\overline{m}, \overline{n})$ that satisfies $0 < \overline{m} < M$ and $0 < \overline{n} < N$ and $\overline{m} \cdot N - \overline{n} \cdot M = 1$, as if M and N had been swapped.

6) Show why (†) implies that $M–N$ and $m–n$ have the same nonzero signs unless $m = 1 = n$.
(Hint: $(m+n) \cdot (M–N) – 1 = (m–n) \cdot (M+N) + 1$.)