Enumerating Pairs of Integers

An Assignment for Math. 55

Exhibit fast computational procedures that enumerate the ordered pairs of positive integers. These procedures must achieve a *bijection* between positive integers k and ordered pairs **(**m, n**)** of positive integers:

$$k := £((m, n)) \text{ is the label of integer pair } (m, n) \text{ ; and}$$
$$(m, n) := \mathbf{P}(k) \text{ is the pair of positive integers labelled by } k .$$

Ideally, the correctness of these procedures will be confirmed by proofs that

$$£(\mathbf{P}(k)) = k \quad\text{and}\quad \mathbf{P}(£((m, n))) = (m, n)$$

for all positive integers k, m and n . Moreover each procedure must be "fast" in the sense that the computation time is practically independent of k until it exceeds the biggest integer upon which your computer or calculator performs arithmetic operations conveniently.

. . . .

One such procedure interleaves the digits of $m = \ldots m_5m_4m_3m_2m_1m_0$ and $n = \ldots n_5n_4n_3n_2n_1n_0$ to form $k = \ldots m_5n_5m_4n_4m_3n_3m_2n_2m_1n_1m_0n_0$, and conversely. What have to be proved correct are the sub-procedures for extracting digits and reassembling them; these are tedious. Here are simpler procedures for all positive integers k, m and n :

$$£((m, n)) := m + (m+n-2)(m+n-1)/2 \;;$$

$$L(k) := \text{integer part of } (1/2 + \sqrt{(2k-1)}) \;;$$
$$M(k) := k - (L(k) - 1)L(k)/2 \;;$$
$$\mathbf{P}(k) := (M(k),\; 1+L(k)-M(k)) \;.$$

Motivation for formula £((m, n)) is best revealed by plotting its values at points **(**m, n**)** in the plane, but motivation is no proof. A proof can be based upon properties of the
Triangular Numbers

$$T_{j+1} := (j+1)j/2 = 1 + 2 + 3 + ... + (j-1) + j = T_j + j \quad\text{for } j = 0, 1, 2, 3, ...$$

($T_0 = T_1 = 0$.) These numbers partition the set of all positive integers k into disjoint intervals

$$T_j < k \le T_{j+1} \quad\text{for } j = 1, 2, 3, ...$$

into some one of which every positive integer k must fall. Given k we find j = L(k) satisfies the last two inequalities because L(k) is a monotone nondecreasing function of k that satisfies

$$L(T_j+1) = j = L(T_{j+1}) ,$$

as can be verified by substitution; do so!

The formula for L(k) would still work if √(2k - 1) were replaced by √(2k - 7/4) , and the proof would be simpler; but then rounding errors could spoil the formula for very big values k . As it is now, L(k) is easily proved correct despite roundoff so long as 2k is less than the smallest positive integer 1000....0001 that cannot be represented exactly in the computer's floating-point arithmetic.

Suppose $k = £((m, n))$; then $T_{m+n-1} < k = £((m, n)) = m + T_{m+n-1} \le T_{m+n}$, so $L(k) = m+n-1$ and then $M(k) = k - T_{L(k)} = m$ and consequently $\mathbf{P}(k) = (m, n)$ as desired. On the other hand, suppose $(m, n) = \mathbf{P}(k)$; then $m+n-1 = L(k)$, and consequently $£((m, n)) = M(k) + T_{L(k)} = k$ as desired. Thus the formulas' correctness is confirmed.

Enumerating the Positive Rationals

An Assignment for Math. 55

The positive rational numbers r = m/n could be identified with the pairs **(**m, n**)** of positive integers were it not necessary to reduce m and n to "lowest terms" by cancelling out their common factors in order to represent every rational r uniquely. For that reason, an enumeration of pairs may prove that the rationals are countable but does not provide an enumeration of them.

We seek an explicit enumeration in the form of a pair of functions £(r) and R(k) defined for all rationals r > 0 and integer indices k > 0 , computable in a time short compared with the integer label k = £(r) when it grows huge, and inverse in the sense that r = R(£(r)) and k = £(R(k)) .

To obtain an explicit enumeration of the positive rationals r , we express 1 + 1/r as a *Terminating Continued Fraction*

$$1 + 1/r = a + 1/(b + 1/(c + 1/(\dots i + 1/(j+1) \dots)))$$

in which each of a, b, c, ..., i and j is a positive integer determined by a repetitive process:

a := integer part of 1 + 1/r ;
b := integer part of 1/(1 + 1/r - a) ;
c := integer part of 1/(1/(1 + 1/r - a) - b) ;
... .

Here the rational numbers of which integer parts are taken have numerators and denominators that shrink in the course of the process, so it must terminate; see Euclid's GCD algorithm. The last integer divisor j+1 exceeds 1 for the sake of the continued fraction's uniqueness.

Thus, every positive rational r can be associated with a finite sequence **(**a, b, c, ..., i, j**)** of positive integers, and *vice-versa*; and the association is *bijective* because different sequences go with different rationals. Next we associate every such finite sequence of positive integers with a finite strictly increasing sequence of nonnegative integers **(**A, B, C, ..., I, J**)** thus:

A := a-1 ; B := A+b ; C := B+c ; ... ; J := I+j .

This association is bijective too because it is reversible:

j = J–I ; ...; c = C–B ; b = B–A ; a = A+1 .

Therefore a bijection has been constructed between the positive rationals r and the finite strictly increasing sequences **(**A, B, C, ..., J**)** of nonnegative integers. Now associate these sequences bijectively with the binary expansions of positive integer indices

$$k := 2^A + 2^B + 2^C + \dots + 2^I + 2^J .$$

Thus, a way has been exhibited to compute quickly a positive integer label k = £(r) for every positive rational r , and inversely to compute quickly a positive rational r = R(k) for every positive integer k . Evidently £(R(k)) = k and R(£(r)) = r for all rationals r > 0 and integers k > 0 , so this is an explicit enumeration of the kind desired. The time taken to compute those functions is roughly proportional to the number of nonzero bits in the binary expansion of k , which grows slowly (logarithmically) with k as it tends to infinity.

A simpler alternative, at first sight, is to compute $k = £(r) := 2^{a-1} \cdot 3^{b-1} \cdot 5^{c-1} \cdot \dots$ as a finite product of prime powers; however, to compute R(k) then we would have to factor k , but nobody knows how to factor gargantuan integers quickly.