Number Theory

<u>Relative Primality:</u> When $\gcd(m,n)=1$, the integers m and n have no prime factors in common and we say that they are *relatively prime*.

$m \perp n \Leftrightarrow m, n$ are integers and $\gcd(m,n)=1$ **(Or, m and n are Relatively Prime)**

A fraction m/n is in lowest terms if and only if $m \perp n$. Since we reduce fractions to lowest terms by casting out common factor of numerator and denominator,

$m/\gcd(m,n) \perp n/\gcd(m,n)$.

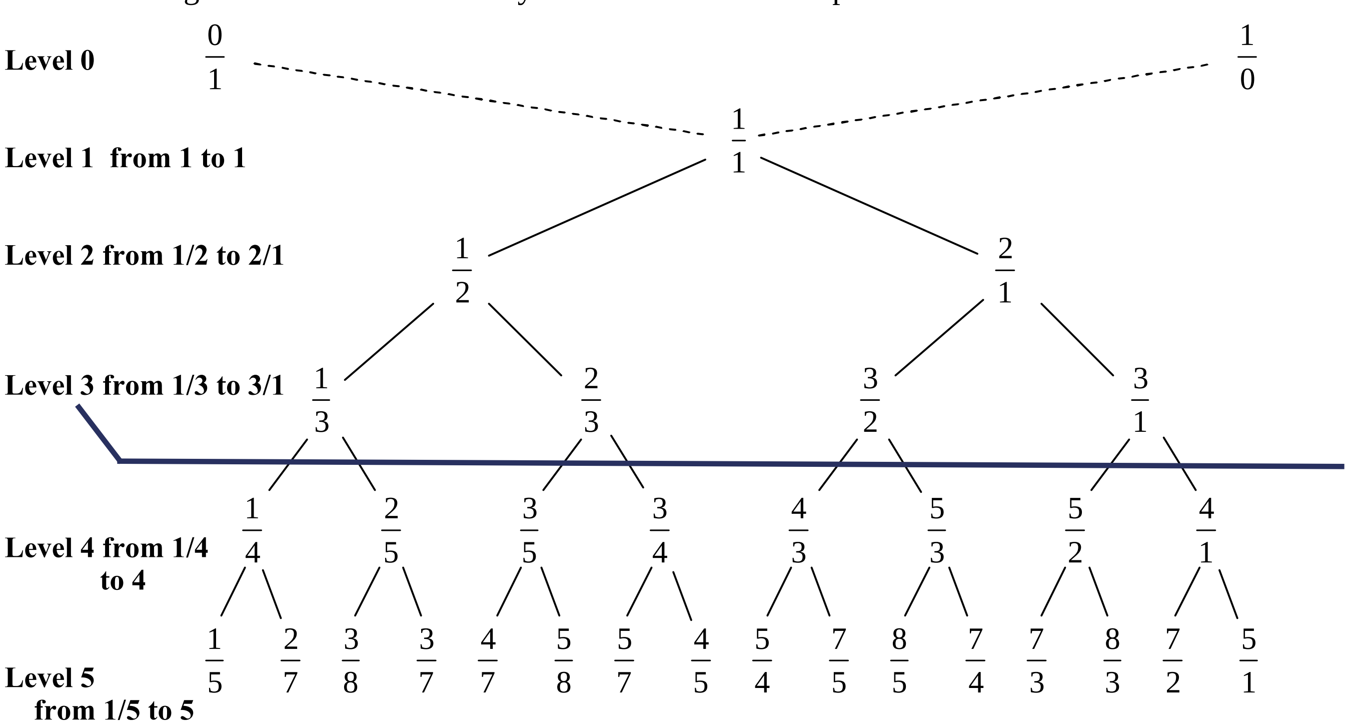
There is a beautiful way to construct the set of all nonnegative fractions m/n with $m \perp n$, called *Stern-Brocot tree*. The idea is to start with the two fractions $\left(\frac{0}{1}, \frac{1}{0}\right)$ and then to repeat the following operation as many times as desired:

Insert $\frac{m+m'}{n+n'}$ between two adjacent fractions $\frac{m}{n}$ and $\frac{m'}{n'}$. The new fraction $\frac{m+m'}{n+n'}$ is called *mediant* of $\frac{m}{n}$ and $\frac{m'}{n'}$. For example, the first step gives us one new entry.

Level 1 $\frac{0}{1}, \frac{1}{1}, \frac{1}{0}$ and the next gives two more:

Level 2 $\frac{0}{1}, \frac{1}{2}, \frac{1}{1}, \frac{2}{1}, \frac{1}{0}$. The next gives four more:

Level 3 $\frac{0}{1}, \frac{1}{3}, \frac{1}{2}, \frac{2}{3}, \frac{1}{1}, \frac{3}{2}, \frac{2}{1}, \frac{3}{1}, \frac{1}{0}$ and then we will get 8, 16 and so on. The entire array can be regarded as an infinite binary tree structure whose top levels look like this:



Proof 1:

Important

If m/n and m'/n' are consecutive fractions at any stage of the construction, we have $m'n - mn' = 1$. We can prove it by induction.

Basis: Initially, $\frac{m}{n} = \frac{0}{1}$ and $\frac{m'}{n'} = \frac{1}{0}$. Thus $m'n - mn' = 1 \cdot 1 - 0 \cdot 0 = 1$

Hypothesis: Let, $m'n - mn' = 1$ is true for two consecutive fractions m/n and m'/n' at any stage of the Stern-Brocot tree.

Induction: Consider, we have new mediant $(m+m')/(n+n')$ between m/n and m'/n'.

m/n, m+m'/n+n', m'/n' $(m+m')n - m(n+n') = mn + m'n - mn - mn' = m'n - mn' = 1$

$m'(n+n') - (m+m')n' = m'n + m'n' - mn' - m'n' = m'n - mn' = 1$ (Proved)

Proof 2:

If $\frac{m}{n} < \frac{m'}{n'}$ and if all values are nonnegative, it's easy to verify that $\frac{m}{n} < \frac{m+m'}{n+n'} < \frac{m'}{n'}$.

$$\frac{m+m'}{n+n'} - \frac{m}{n} = \frac{mn + m'n - mn - mn'}{n(n+n')} = \frac{m'n - mn'}{n(n+n')} = \frac{1}{n(n+n')}.$$

Again, $\frac{m'}{n'} - \frac{m+m'}{n+n'} = \frac{m'n + m'n' - mn' - m'n'}{n'(n+n')} = \frac{m'n - mn'}{n'(n+n')} = \frac{1}{n'(n+n')}$.

Though $n > 0$ and $n' > 0$. we can write, $\frac{1}{n'(n+n')} > 0$ and $\frac{1}{n(n+n')} > 0$ which concludes the prove $\frac{m}{n} < \frac{m+m'}{n+n'} < \frac{m'}{n'}$.

Proof 3:

Important

Is there any positive fraction a/b with $a \perp b$ possibly omitted from Stern-Brocot tree?

Let, a/b is a positive fraction. Thus, $\frac{m}{n} = \frac{0}{1} < \left(\frac{a}{b}\right) < \frac{m'}{n'} = \frac{1}{0}$. The construction forms $(m+m')/(n+n')$ and there are three cases. Either $(m+m')/(n+n') = a/b$ and we win; or $(m+m')/(n+n') < a/b$ and we can set $m \leftarrow m+m'$, $n \leftarrow n+n'$; **(i.e., Move to the Right SubTree)** or $(m+m')/(n+n') > a/b$ and we can set $m' \leftarrow m+m'$, $n' \leftarrow n+n'$. The process can't go **(Move to Left Subtree)** infinitely because ~~of the conditions~~ **m/n < a/b < m'/n' and the inequality gradually comes closer and closer to equality to a/b as**

SKIP the Calculation

$\frac{a}{b} - \frac{m}{n} > 0 \Rightarrow an - bm > 0 \quad \cdots \quad (1)$ and $\frac{m'}{n'} - \frac{a}{b} > 0 \Rightarrow bm' - an' > 0 \quad \cdots \quad (2)$.

$(1) \times (m'+n') + (2) \times (m+n) \Rightarrow (m'+n')(an - bm) + (m+n)(bm' - an') \geq m' + n' + m + n$

$\Rightarrow am'n - bmm' + ann' - bmn' + bmm' - amn' + bm'n - ann' \geq m' + n' + m + n$

$\Rightarrow am'n - bmn' - amn' + bm'n \geq m' + n' + m + n$

$\Rightarrow m'n(a+b) - mn'(a+b) \geq m' + n' + m + n$

$\Rightarrow (a+b)(m'n - mn') \geq m' + n' + m + n$

$\Rightarrow a + b \geq m' + n' + m + n \quad ; \quad [\because m'n - mn' = 1]$

Either m or n or m' or n' increases at each step, so we must win after at most $(a + b)$ steps.

~~The *Farey series* of order *N*, denoted by F_N, is the set of all reduced fractions between 0 and 1 whose denominators are N or less, arranged in increasing order. For example, if $N=6$ we have $F_6 = \frac{0}{1}, \frac{1}{6}, \frac{1}{5}, \frac{1}{4}, \frac{1}{3}, \frac{2}{5}, \frac{1}{2}, \frac{3}{5}, \frac{2}{3}, \frac{3}{4}, \frac{4}{5}, \frac{5}{6}, \frac{1}{1}$.~~

~~We can obtain F_N in general by starting with $F_1 = \frac{0}{1}, \frac{1}{1}$ and then inserting mediants whenever it's possible to do so. To obtain F_N from F_{N-1}, we simply insert the fraction $(m+m')/N$ between consecutive fractions m/n and m'/n' of F_{N-1} whose denominators sum equals to *N*. For example, it's easy to obtain F_7 from the elements of F_6 by inserting $\frac{1}{7}, \frac{2}{7}, \cdots, \frac{6}{7}$ according to the stated rule:~~

$$F_7 = \frac{0}{1}, \frac{1}{7}, \frac{1}{6}, \frac{1}{5}, \frac{1}{4}, \frac{2}{7}, \frac{1}{3}, \frac{2}{5}, \frac{3}{7}, \frac{1}{2}, \frac{4}{7}, \frac{3}{5}, \frac{2}{3}, \frac{5}{7}, \frac{3}{4}, \frac{4}{5}, \frac{5}{6}, \frac{6}{7}, \frac{1}{1}.$$

~~When *N* is prime, $N-1$ new fractions will appear, but otherwise we will have fewer than $N-1$ factors, because this process generates only numerators that are relatively prime to *N*. F_N is a *subtree* of the Stern-Brocot tree, obtained by pruning off unwanted branches. It follows that $m'n - mn' = 1$ whenever m/n and m'/n' are consecutive elements of a Farey series.~~

Let's used the letter L and R to stand for going down to the left or right branch as we proceed from the root of the Stern-Brocot tree to a particular fraction; then a string of L's and R's uniquely identifies a place in the tree. For instance, LRRL means that we go left from $\frac{1}{1}$ down to $\frac{1}{2}$, then right to $\frac{2}{3}$, then right to $\frac{3}{4}$, then left to $\frac{5}{7}$. We can consider LRRL to represent $\frac{5}{7}$. Every positive fraction gets represented in this way as a unique string of L's and R's.

Suppose, we are given a string of L's and R's, we have to find out what fraction corresponds to it in a Stern-Brocot tree. For example, f (LRRL) = $\frac{5}{7}$. We can maintain a 2×2 matrix to find out such fractions.

$$M(S) = \begin{pmatrix} n & n' \\ m & m' \end{pmatrix}$$

A step to left replaces n' by $n+n'$ and m' by $m+m'$; hence

$$M(SL) = \begin{pmatrix} n & n+n' \\ m & m+m' \end{pmatrix} = \begin{pmatrix} n & n' \\ m & m' \end{pmatrix}\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} = M(S)\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$$

Similarly, when we turn right then, we replace *n* by $n+n'$ and *m* by $m+m'$.

$$M(SR) = \begin{pmatrix} n+n' & n' \\ m+m' & m' \end{pmatrix} = \begin{pmatrix} n & n' \\ m & m' \end{pmatrix}\begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} = M(S)\begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$$

Therefore, we can define L and R as 2×2 matrices.

$$L = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \quad \text{and} \quad R = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$$

For example, $M(LRRL) = LRRL = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}\begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$

$$= \begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix}\begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix} = \begin{pmatrix} 3 & 4 \\ 2 & 3 \end{pmatrix}$$

The ancestral fraction that enclose LRRL $= \frac{2+3}{3+4} = \frac{5}{7}$.

Determine the fraction corresponding to the sequence: LRRL without drawing the Stern Brocot tree

read the Algorithm of the NEXT PAGE

Consider, we are given positive integers m and n with $m \perp n$, we have to find out the string of L's and R's that corresponds to m/n in Stern-Brocot tree. We can do it using 'binary search' on Stern-Brocot tree:

$S := I$

while $m/n \neq f(S)$ **do**

  if $m/n < f(S)$ **then** (output(L); $S := SL$)

    else (output(R); $R = SR$) .

This outputs the desired string of L's and R's. For example, if given $m/n = 5/7$, then the algorithm works according to the following way:

Pass 1 : m/n = 5/7 < f(S) = 1, Output : L.

$$L = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} = \frac{0+1}{1+1} = \frac{1}{2}$$

Pass 2 : m/n = 5/7 > f(L) = 1/2 , Output : LR.

$$LR = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} = \begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix} = \frac{1+1}{2+1} = \frac{2}{3}$$

Pass 3 : m/n = 5/7 > f(LR) = 2/3 , Output : LRR.

$$LRR = \begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix}\begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} = \begin{pmatrix} 3 & 1 \\ 2 & 1 \end{pmatrix} = \frac{2+1}{3+1} = \frac{3}{4}$$

Pass 4 : m/n = 5/7 < f(LRR) = 3/4 , Output : LRRL.

$$LRRL = \begin{pmatrix} 3 & 1 \\ 2 & 1 \end{pmatrix}\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 3 & 4 \\ 2 & 3 \end{pmatrix} = \frac{2+3}{3+4} = \frac{5}{7}.$$

There is another algorithm available to find out L's and R's of particular fraction in Stern-Brocot tree using following property

$$\frac{m}{n} = f(RS) \Leftrightarrow \frac{m-n}{n} = f(S), \quad \text{where } m > n$$

$$\frac{m}{n} = f(LS) \Leftrightarrow \frac{m}{n-m} = f(S), \quad \text{where } m < n$$

Write an Algorithm to generate the the L-R sequence to locate a given Fractional value in the Stern-Brocot tree. Demonstrate the algorithm using an example.

i.e. we can transform the binary search algorithm to the following matrix-free procedure.

***include these 2 lines in the algo ==> if (gcd(m,n) > 1) then output ("Not Present in Tree, because m and n are not relatively prime"); return; else if (m=n==1) output ("Found at Root!"); return;**

while $m \neq n$ **do**

if m < n **then** (output(L); $n := n - m$)

else (output(R); $m := m - n$)

For instance, given *m*/*n* = 5/7, we have successively

	Pass 1	Pass 2	Pass 3	Pass 4	
m =	5	5	5 – 2 = 3	3 – 2 = 1	1
n =	7	7 – 5 = 2	2	2	2 – 1 = 1
output :	L	R	R	L	

☺ Good Luck ☺

Q: Determine the level of the Stern Brocot tree that contains the fraction 5/7

Ans: Find the LR sequence as above: LRRL ... this means level no. = 4+1 = 5