$$\frac{2}{\frac{1}{a}+\frac{1}{b}} < \sqrt{ab} < \frac{b-a}{\ln b - \ln a} < \frac{a+b}{2} < \sqrt{\frac{a^2+b^2}{2}} \qquad (0 < a < b), \qquad (4)$$

relating the harmonic, geometric, logarithmic, arithmetic, and root-mean-square means.

For a rigorous proof of (1), observe that

$$\cosh x - 1 \equiv \left(\frac{e^{x/2} - e^{-x/2}}{2}\right)^2 > 0 \qquad \text{for} \quad x > 0.$$

Therefore,

$$\frac{1}{\cosh^2 x} < 1 < \cosh x \qquad \text{for} \quad x > 0,$$

and (1) follows by integrating the preceding inequality from $x = 0$ to $x = t$. The other inequalities of (3) are obvious.

Trees and Tennis Rankings

Curtis Cooper, Central Missouri State University, Warrensburg, MO

Everyone is familiar with the usual kind of "elimination" tennis tournament in which losers drop out and winners continue to play until one undefeated player remains. Such a tournament is nicely described by a *tree* whose vertices $1, 2, \ldots, n$ represent the n players and whose edges represent the matches that were played—the direction of each edge pointing toward the loser. In the tournament of Figure 1, for example, player 4 won by beating player 6 (who beat 11 and 1), player 10, and player 3 (who beat 8, 2, and 5). In addition, player 8 beat player 9 (who beat 7 and 12).

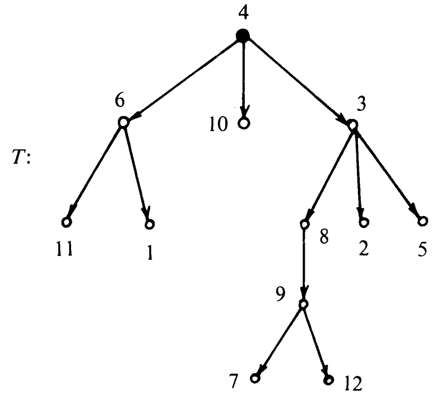

Figure 1.

Suppose we wanted to rank the players of this tournament, subject to the rule: *for each edge of T, player i precedes player j if and only if i beats j.* There is clearly no unique way to do so since T does not contain enough information for a full account of the players' relative strengths. Indeed, although 4 (the *root* of tree T) must begin the list, the next member of the ranking could be 6, 10, or 3. Therefore, the general question to be asked is:

How many rankings can be generated by a given rooted (directed) tree?

Let us first show that for the tournament in Figure 1, the answer is 158400. The analysis of the general case is simple and pretty.

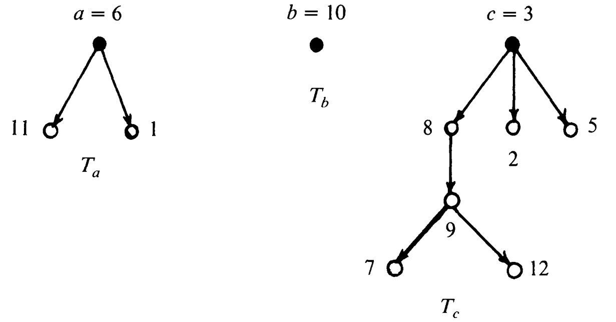

Figure 2a.

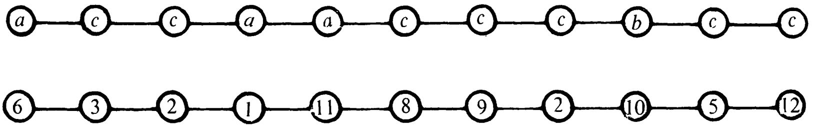

Figure 2b.

Let (Figure 2a) a, b, c be the players who lost to 4. Then a, b, c are the roots of subtrees T_a, T_b, T_c of T. Knowing the number of rankings $R(T_a), R(T_b), R(T_c)$ for these subtrees will enable us to compute the number of rankings $R(T)$ for T itself.

There are $\begin{pmatrix} & 11 & \\ 3 & 1 & 7 \end{pmatrix} = \frac{11}{3!\ 1!\ 7!}$ ways to string 11 beads using 3 amber, 1 blue, and 7 crimson beads. Assume that these colored beads are precisely the numbered vertices of T_a, T_b, T_c. Then for any particular color arrangement on the string, there are respective $R(T_a), R(T_b), R(T_c)$ rankings for each color. (Figure 2b depicts one possible color arrangement and one possible ranking for that given color arrangement.) Thus,

$$R(T) = \frac{11!}{3!\ 1!\ 7!} \times R(T_a) \times R(T_b) \times R(T_c).$$

Clearly, $R(T_a) = 2$ and $R(T_b) = 1$. To obtain $R(T_c)$, we can invoke the preceding argument, applied to subtrees T_8, T_2, T_5 of T_c. This yields

$$R(T_c) = \frac{6!}{4!\ 1!\ 1!} \times 2 \times 1 \times 1.$$

All together, we obtain

$$R(T) = \frac{11!}{3!\ 1!\ 7!} \times 2 \times 1 \times \left(\frac{6!}{4!\ 1!\ 1!} \times 2 \times 1 \times 1 \right) = 158400.$$

Turning now to the general case, we assume that the tournament-representing tree T can be arranged so that the root r occurs by itself at the top, in "level 0," of the tree (Figure 3). Let the players $w_1, w_2, \ldots, w_k$ who lost to winner r be placed (in any order) in level 1 of the tree.

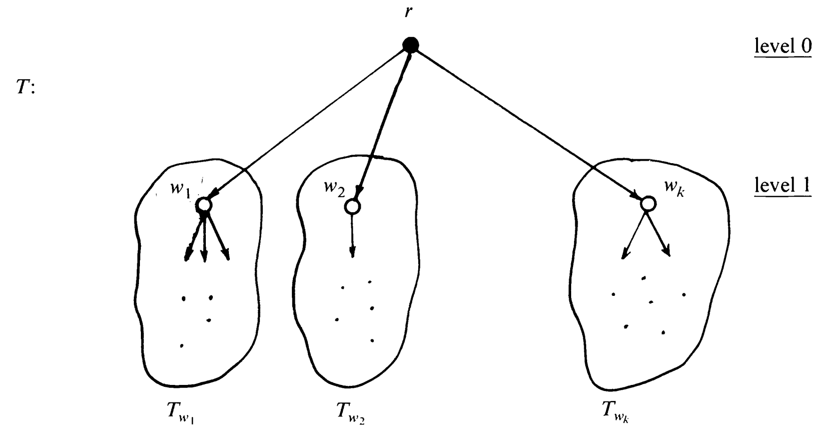

Figure 3.

Each vertex w_i is the root of a subtree T_{w_i} of T; each T_{w_i} represents a subtournament, the ranking of whose players provides us with a problem that is identical to the kind of problem we started with. Let us show that, if we can solve the problem for the subtrees T_{w_i}, we can succeed with T itself.

Every ranking for T has a place for each of the n players and, because the first position must go to winner r, our freedom extends only through the remaining $n - 1$ positions. We don't care how the vertices of the various subtrees get interlaced, so long as the vertices of each T_{w_i}, considered alone, satisfy "i before j means i beat j." Consequently, a ranking for T is constructed by the double procedure:

(1) For each i, select $|T_{w_i}|$ places among the $n - 1$ open positions for the vertices of T_{w_i};

(2) Enter in these places the $|T_{w_i}|$ vertices of T_{w_i} in any acceptable $R(T_{w_i})$ ranked order.

Practically by definition, the number of ways of achieving (1) is the multinomial coefficient

$$\binom{n-1}{|T_{w_1}|\,|T_{w_2}| \cdots |T_{w_k}|} = \frac{(n-1)!}{|T_{w_1}|!\,|T_{w_2}|! \cdots |T_{w_k}|!}.$$

And for each of these partitions, the number of ways of doing (2) is $R(T_{w_1}) \times R(T_{w_2}) \times \cdots \times R(T_{w_k})$. Therefore, the total number of rankings for T is given by the recurrence relation

$$R(T) = \frac{(n-1)!}{|T_{w_1}|!\,|T_{w_2}|! \cdots |T_{w_k}|!} \cdot R(T_{w_1}) R(T_{w_2}) \cdots R(T_{w_k}),$$

with the obvious initial value $R(T) = 1$ for $|T| = 1$.