COMPUTER MUSINGS

a series of impromptu talks by

DON KNUTH

Professor Emeritus of The Art of Computer Programming

Gates Building Room B01, 4:15 pm

Fifth Annual "Christmas Tree Lecture," Wednesday December 3 1997

1997

Lattices of Trees

EVERYBODY WELCOME

"It was a musing." — Peter Gordon

See `http://www-cs-faculty.stanford.edu/~knuth/musings.html` for further information

I lectured on this topic also in MIT Combinatorics Seminar, 10 Dec 1999

▸ **25.** [*30*] (*Pruning and grafting.*) Representing binary trees as in Algorithm B, design an algorithm that visits all link tables $l_1 \ldots l_n$ and $r_1 \ldots r_n$ in such a way that, between visits, exactly one link changes from j to 0 and another from 0 to j, for some index j. (In other words, every step removes some subtree j from the binary tree and places it elsewhere, preserving preorder.)

26. [*M31*] (*The Kreweras lattice.*) Let F and F' be n-node forests with their nodes numbered 1 to n in preorder. We write $F \mathrel{|\!\!<} F'$ ("F coalesces F'") if j and k are siblings in F whenever they are siblings in F', for $1 \le j < k \le n$. Figure 39 illustrates this partial ordering in the case $n = 4$; each forest is encoded by the sequence $c_1 \ldots c_n$ of (9) and (10), which specifies the depth of each node. (With this encoding, j and k are siblings if and only if $c_j = c_k \le c_{j+1}, \ldots, c_{k-1}$.)

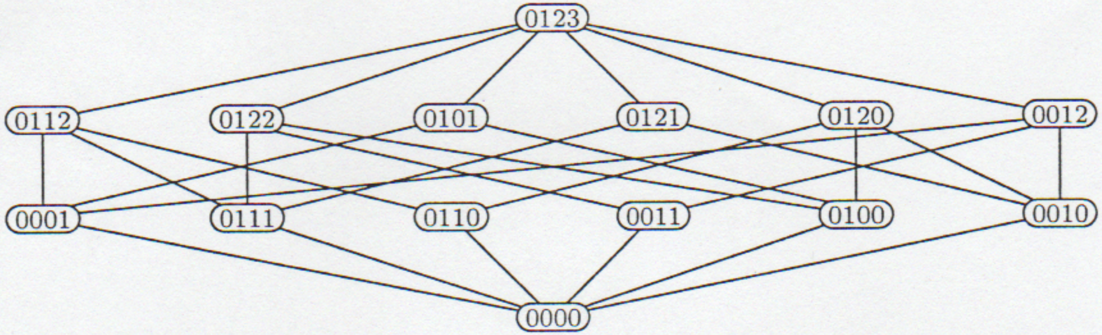


Fig. 39. The Kreweras lattice of order 4. Each forest is represented by its sequence of node depths $c_1c_2c_3c_4$ in preorder. (See exercises 26–28.)

a) Let Π be a partition of $\{1, \ldots, n\}$. Show that there exists a forest F, with nodes labeled $(1, \ldots, n)$ in preorder and with

$$j \equiv k \text{ (modulo } \Pi) \iff j \text{ is a sibling of } k \text{ in } F,$$

if and only if Π satisfies the *noncrossing* property

$$i < j < k < l \text{ and } i \equiv k \text{ and } j \equiv l \text{ (modulo } \Pi) \quad \text{implies} \quad i \equiv j \equiv k \equiv l \text{ (modulo } \Pi).$$

b) Given any two n-node forests F and F', explain how to compute their least upper bound $F \veebar F'$, the element such that $F \mathrel{|\!\!<} G$ and $F' \mathrel{|\!\!<} G$ if and only if $F \veebar F' \mathrel{|\!\!<} G$.

c) When does F' cover F with respect to the relation $\mathrel{|\!\!<}$? (See exercise 7.2.1.4–55.)

d) Show that if F' covers F, it has exactly one less leaf than F.

KREWERAS lattice (1971)

"noncrossing partitions"

number the nodes in preorder

$F \preceq F'$: if j is sibling of k in F'

then j is sibling of k in F

$F \veebar F'$: least upper bound

$F \preceq G$ and $F' \preceq G \iff F \veebar F' \preceq G$

j is sibling of k in $F \veebar F'$

$\Updownarrow$

j is sibling of k in F

and

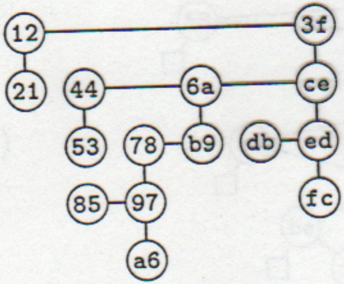
j is sibling of k in F'

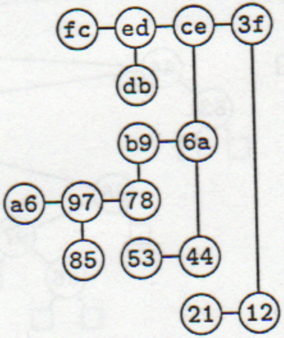
of $x_{j+1} - 1$, the rightmost descendant

(d) Obvious, by (c). Thus the forests are ranked from bottom to top by the number of nonleaf nodes they contain (which is one less than the number of blocks in Π).

(e) Exactly $\sum_{k=0}^{n} e_k(e_k-1)/2$, where $e_0 = n - e_1 - \cdots - e_n$ is the number of roots.

(f) Dualization is similar to the transposition operation in exercise 12, but we use left-sibling and right-child links instead of left-child and right-sibling, and we transpose about the *minor* diagonal:





("Right" links now point downward. Notice that j is the rightmost child of k in F if and only if j is the left sibling of k in F^D. Preorder of F^D reverses the preorder of F, just as postorder of F^T reverses postorder of F.)

(g) From (f) we can see that F' covers F if and only if F^D covers F'^D. (Therefore F^D has $n+1-k$ leaves if F has k.)

(h) $F \barwedge F' = (F^D \veebar F'^D)^D$.

(i) No. If it did, equality would necessarily hold, by duality. But, for example, $0101 \barwedge 0121 = 0000$ and $0101 \veebar 0121 = 0123$, while leaves(0101) + leaves(0121) $\neq$ leaves(0000) + leaves(0123).

[Noncrossing partitions were first considered by H. W. Becker in *Math. Mag.* **22** (1948), 23–26. G. Kreweras proved in 1971 that they form a lattice; see the references in answer 2.3.4.6–3.]

Duality:

$$F \mathrel{\vert\!\!\prec} F' \iff F^D \mathrel{\succ\!\!\vert} F'^D$$

Therefore the greatest lower bound is

$$F \barwedge F' = (F^D \veebar F'^D)^D$$

TAMARI lattice (1951)

$F \dashv F'$: $\text{descendants}(j, F) \le \text{descendants}(j, F')$ for all nodes j

greatest lower bound

$F \perp F'$: $\text{descendants}(j, F \perp F')$

$= \min\big(\text{descendants}(j, F), \text{descendants}(j, F')\big)$

$F \dashv F' \implies F^D \vdash F'^D$

least upper bound

$F \top F' = (F^D \perp F'^D)^D$

semidistributive laws

$F \perp G = F \perp H \implies F \perp (G \top H) = F \perp G$

$F \top G = F \top H \implies F \top (G \perp H) = F \top G$

Relation between lattices:

$$F \mid\!\!< F' \implies F \dashv F'$$

Therefore

$$F \wedge F' \dashv F \perp F'$$

$$F \veebar F' \vdash F \top F'$$

STANLEY lattice (1975)

$F \subseteq F'$: $\text{ancestors}(j, F) \le \text{ancestors}(j, F')$

greatest lower bound

$F \cap F'$

least upper bound

$F \cup F'$

(Стенли)

$$\text{ancestors}(j, F \cap F') = \min(\text{ancestors}(j, F), \text{ancestors}(j, F'))$$
$$\text{ancestors}(j, F \cup F') = \max(\text{ancestors}(j, F), \text{ancestors}(j, F'))$$

distributive laws

$$F \cup (G \cap H) = (F \cup G) \cap (F \cup H)$$
$$F \cap (G \cup H) = (F \cap G) \cup (F \cap H)$$

g) Prove that $F \Kleq F'$ holds if and only if $F'^D \Kleq F^D$. (Because of this property, dual elements have been placed symmetrically about the center of Fig. 39.)

h) Given any two n-node forests F and F', explain how to compute their greatest lower bound $F \barwedge F'$; that is, $G \Kleq F$ and $G \Kleq F'$ if and only if $G \Kleq F \barwedge F'$.

i) Does this lattice satisfy a semimodular law analogous to exercise 7.2.1.5–12(f)?

▶ **27.** [*M33*] (*The Tamari lattice.*) Continuing exercise 26, let us write $F \dashv F'$ if the jth node in preorder has at least as many descendants in F' as it does in F, for all j. In other words, if F and F' are characterized by their scope sequences $s_1 \ldots s_n$ and $s'_1 \ldots s'_n$ as in Table 2, we have $F \dashv F'$ if and only $s_j \le s'_j$ for $1 \le j \le n$. (See Fig. 40.)

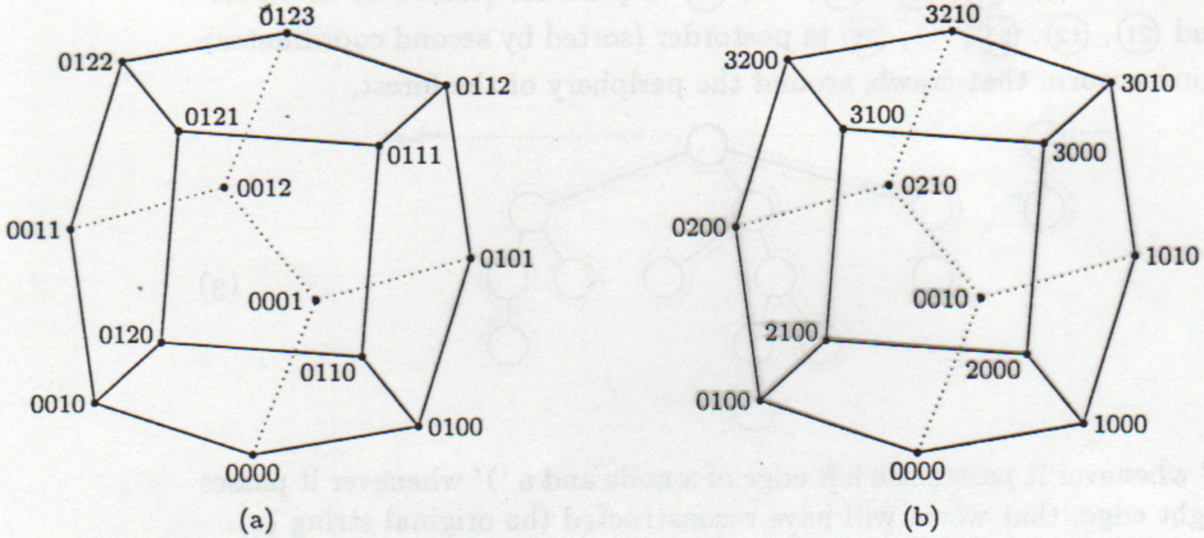


Fig. 40. The Tamari lattice of order 4. Each forest is represented by its sequences of (a) node depths and (b) descendant counts, in preorder. (See exercises 26–28.)

a) Show that the scope coordinates $\min(s_1, s'_1)\min(s_2, s'_2)\ldots\min(s_n, s'_n)$ define a forest that is the greatest lower bound of F and F'. (We denote it by $F \perp F'$.) *Hint:* Prove that $s_1 \ldots s_n$ corresponds to a forest if and only if $0 \le k \le s_j$ implies $s_{j+k} + k \le s_j$, for $0 \le j \le n$, if we define $s_0 = n$.

b) When does F' cover F in this partial ordering?

c) Prove that $F \dashv F'$ if and only if $F'^D \dashv F^D$. (Compare with exercise 26(g).)

d) Explain how to compute a least upper bound, $F \top F'$, given F and F'.

e) Prove that $F \Kleq F'$ in the Kreweras lattice implies $F \dashv F'$ in the Tamari lattice.

f) True or false: $F \barwedge F' \dashv F \perp F'$.

g) True or false: $F \veebar F' \Kleq F \top F'$.

h) What are the longest and shortest paths from the top of the Tamari lattice to the bottom, when each forest of the path covers its successor? (Such paths are called *maximal chains* in the lattice; compare with exercise 7.2.1.4–55(h).)

STANLEY lattice (1975)

$F \subseteq F'$: $\text{ancestors}(j,F) \le \text{ancestors}(j,F')$

greatest lower bound $F \cap F'$

least upper bound $F \cup F'$

(Стенли)

$$\text{ancestors}(j, F \cap F') = \min(\text{ancestors}(j,F), \text{ancestors}(j,F'))$$
$$\text{ancestors}(j, F \cup F') = \max(\text{ancestors}(j,F), \text{ancestors}(j,F'))$$

distributive laws

$$F \cup (G \cap H) = (F \cup G) \cap (F \cup H)$$
$$F \cap (G \cup H) = (F \cap G) \cup (F \cap H)$$

28. [*M26*] (*The Stanley lattice.*) Continuing exercises 26 and 27, let us define yet another partial ordering on n-node forests, saying that $F \subseteq F'$ whenever the depth coordinates $c_1 \ldots c_n$ and $c'_1 \ldots c'_n$ satisfy $c_j \le c'_j$ for $1 \le j \le n$. (See Fig. 41).

a) Prove that this partial ordering is a lattice, by explaining how to compute the greatest lower bound $F \cap F'$ and least upper bound $F \cup F'$ of any two given forests.

b) Show that Stanley's lattice satisfies the distributive laws

$$F \cap (G \cup H) = (F \cap G) \cup (F \cap H), \qquad F \cup (G \cap H) = (F \cup G) \cap (F \cup H).$$

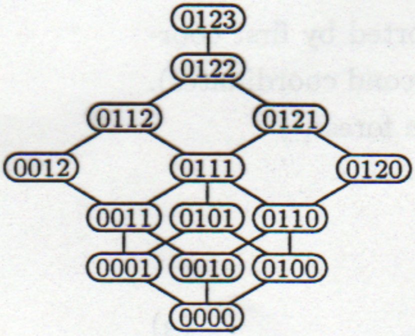


Fig. 41. The Stanley lattice of order 4. Each forest is represented by its sequence of node depths in preorder. (See exercises 26–28.)

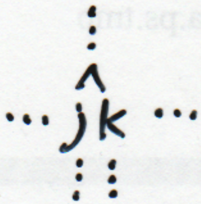
c) When does F' cover F in this lattice?

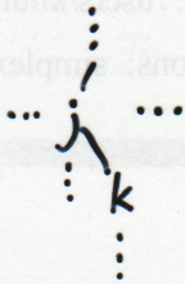
d) True or false: $F \subseteq G$ if and only if $F^R \subseteq G^R$.

e) Prove that $F \subseteq F'$ in the Stanley lattice whenever $F \dashv F'$ in the Tamari lattice.

29. [*HM31*] The covering graph of a Tamari lattice is sometimes known as an "associahedron," because of its connection with the associative law (14), proved in exercise 27(b). The associahedron of order 4, depicted in Fig. 40, looks like it has three square faces and six faces that are regular pentagons. (Compare with Fig. 23 in exercise 7.2.1.2–60, which shows the "permutahedron" of order 4, a well-known Archimedean solid.) Why doesn't Fig. 40 show up in classical lists of uniform polyhedra?

F covered by F'





$$F \dashv F' \quad \Longrightarrow \quad F \subseteq F'$$

Three Catalan Bijections

D. Knuth

REPORT No. 04, 2004/2005, spring

ISSN 1103-467X

ISRN IML-R- -04-04/05- -SE+spring

INSTITUT MITTAG—LEFFLER

Three Catalan Bijections

by Donald E. Knuth

Institut Mittag-Leffler and Stanford University

25 February 2005

This note contains three short programs that implement one-to-one correspondences between four kinds of combinatorial structures:

1) Ordered forests with n nodes and pruning order m;
2) Binary trees with n nodes and Strahler number m;
3) Nested strings (Dyck words) of length $2n$ and log-height m;
4) Kepler towers with n bricks and m walls.

In each case the number of structures of size n is the Catalan number $C_n = \binom{2n}{n}/(n+1)$, and — surprisingly — the bijections also preserve the parameter m.

Given a number $n > 1$, each program generates all C_n objects of one type, bijects them into objects of another type, verifies that the parameter m has not changed, and applies the inverse bijection to prove constructively that the correspondence is indeed one-to-one.

Program 1, called ZEILBERGER, converts between (1) and (2). Program 2, FRANÇON, converts between (2) and (3). And Program 3, VIENNOT, converts between (3) and (4). Incidentally, Kepler towers appear to be a completely new kind of object, recently invented by Xavier Viennot and introduced here for the first time. Simple bijections between (2) and (4), or between (1) and (4), are not yet known, although complex bijections could of course be obtained by composing those given here.

The first bijection was introduced by Doron Zeilberger in 1990, yet its computer implementation is not without interest. Although Zeilberger's algorithm was correct, his proof of correctness was not quite complete; Program 1 therefore removes any lingering doubts that may have existed. More significantly, the program demonstrates a strong property of Zeilberger's bijection that may not have been noticed before: Node x is the leftmost child of node y in the ordered forest if and only if node x is the left child of node y in the corresponding binary tree.

The second bijection was inspired by the work of Jean Françon in 1984, but it is organized here in a new way, based on a heap-like data structure. Therefore it appears to solve an open problem that he stated, namely to construct a "direct" parameter-preserving bijection between objects of types (2) and (3). Moreover, Program 2 has the interesting property that the bijection and its inverse both carry out their work in the same direction as they translate one object to another. By contrast, the inverse bijections in Programs 1 and 3 essentially cause time to run backward when they undo the effects of forward-runnning bijections.

Program 3 introduces a new bijection that was recently explained pictorially to the author by its creator, Xavier Viennot. The resulting computer program has turned out to be remarkably simple and fast.

All three programs have been written with the conventions of "literate programming," as embodied in the CWEB system developed by Silvio Levy and the author. This style of presentation features informal, human-oriented English descriptions alternating with formal, computer-oriented commands. The latter instructions are expressed in the C programming language; but mathematicians unfamiliar with C should still be able to get the gist of the ideas by reading the English commentary. (A detailed explanation of how to read `CWEB` programs — more than almost anybody needs to know — can be found in Chapter 4 of the author's book *The Stanford GraphBase*.)

Incidentally, these programs are independent of each other. They can be downloaded from the author's web site `http://www-cs-faculty.stanford.edu/~knuth/programs.html` and used without restriction.