Sorted and/or sortable permutations

Mireille Bousquet-Mélou*

LaBRI, Université Bordeaux 1

351 cours de la Libération

33405 Talence Cedex, FRANCE

bousquet@labri.u-bordeaux.fr

Abstract

In his Ph.D. thesis [21], Julian West studied in depth a map Π that acts on permutations of the symmetric group $\mathfrak{S}_n$ by partially sorting them through a stack. The main motivation of this paper is to characterize and count the permutations of $\Pi(\mathfrak{S}_n)$, which we call sorted permutations. This is equivalent to counting preorders of increasing binary trees. We first find a local characterization of sorted permutations. Then, using an extension of Zeilberger's factorisation of two-stack sortable permutations [23], we obtain for the generating function of sorted permutations an unusual functional equation.

Out of curiosity, we apply the same treatment to four other families of permutations (general permutations, one-stack sortable permutations, two-stack sortable permutations, sorted and sortable permutations) and compare the functional equations we obtain. All of them have similar features, involving a divided difference. Moreover, most of them have interesting q-analogs obtained by counting inversions. We solve (some of) our equations.

1 Introduction

To begin with, we define the sorting procedure and the families of permutations we shall enumerate.

1.1 The sorting procedure

In his Ph.D. thesis [21], Julian West studied a procedure Π that permutes the letters of a word σ having distinct letters in the alphabet $\{1, 2, 3, \ldots\}$. The procedure uses a stack and works as follows (Fig.1). At the beginning, the word $\sigma = \sigma^{(0)}$ lies to the right of the stack, which is empty. If σ has m letters, the procedure will have $2m$ steps. After the ith step, for $i \geq 0$, a word $\sigma^{(i)}$ lies to the right of the stack, while a word $\tau^{(i)}$ lies to the left of the stack. If $\sigma^{(i)}$ is not empty, and if its first letter, say a, is smaller than the top letter of the stack (or if the stack is empty), we add a to the top of the stack. Otherwise, we remove the top letter from the stack and add it at the end of $\tau^{(i)}$. In other words, we add letters to the stack as long as it remains a "Hanoï tower", and otherwise remove letters from the stack. The word $\tau^{(2m)}$ has m letters, and we define it to be $\Pi(\sigma)$, the word obtained by *sorting σ through a stack.* Fig.1 shows four steps of this procedure applied to $\sigma = 2351674$.

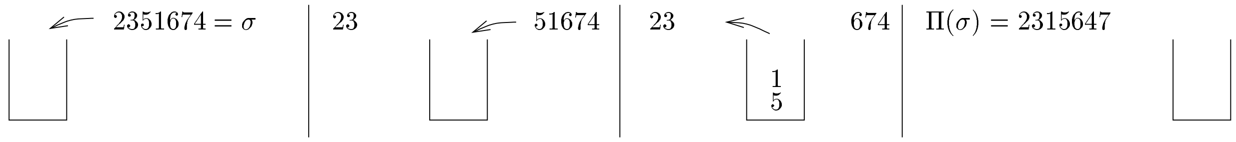


Figure 1: The sorting algorithm applied to $\sigma = 2351674$.

This procedure extends a procedure described by Knuth [14, p. 238] (although Knuth's procedure, nicely described in terms of railway switching networks, goes somehow backwards). As observed by West [22], the

*Partially supported by the Conseil Régional d'Aquitaine.

map Π can alternatively be described recursively by

$$\Pi(\sigma^L n \sigma^R) = \Pi(\sigma^L)\Pi(\sigma^R)n, \tag{1}$$

where n is the largest letter of the word $\sigma = \sigma^L n \sigma^R$. We observe that, if σ has m letters, then $\Pi^{m-1}(\sigma)$ is an increasing word; this shows that Π really *sorts* the letters of σ (although not very fast!).

Clearly, we can restrict our attention to the action of Π on permutations. Let $\mathfrak{S}_n$ be the set of permutations of length n. Following West [21], we represent the action of Π on $\mathfrak{S}_n$ by a *sorting tree*: the nodes of this tree are the elements of $\mathfrak{S}_n$, and an edge connects σ to $\Pi(\sigma)$ for all $\sigma \in \mathfrak{S}_n$ (Fig.2).

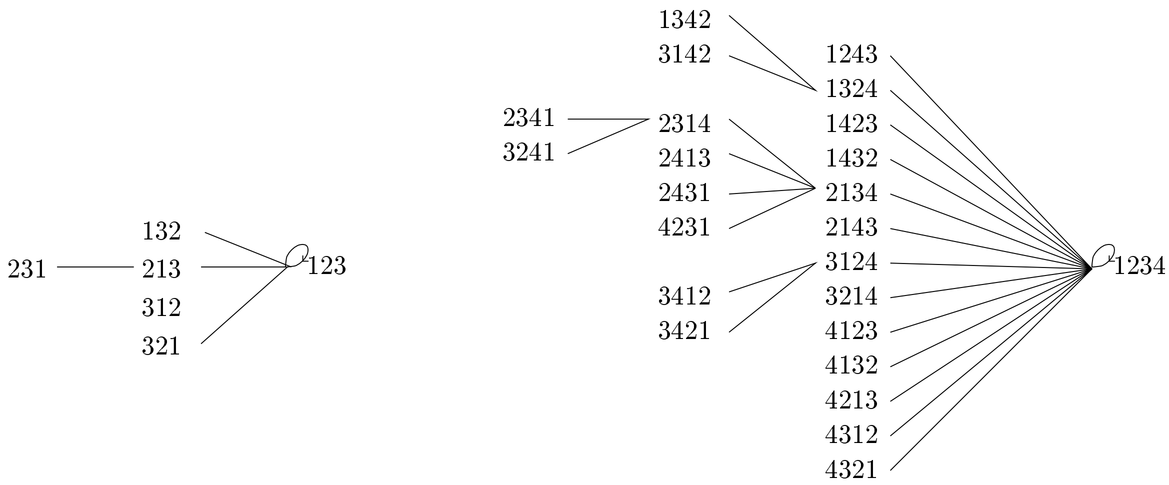


Figure 2: The sorting trees for $\mathfrak{S}_3$ and $\mathfrak{S}_4$.

We can visualize on this tree the four classes of permutations we will consider in this paper.

- **One-stack sortable permutations**

A permutation $\sigma \in \mathfrak{S}_n$ is *one-stack sortable* if $\Pi(\sigma) = 12\ldots n$, *i.e.*, if it occurs in the last two columns of the sorting tree. It is known [14, p.531] that the number of such permutations is the Catalan number $C_n = \binom{2n}{n}/(n+1)$, and that these permutations are exactly the permutations avoiding the pattern 231: there exists no triple (i,j,k) with $1 \le i < j < k \le n$ such that $\sigma(k) < \sigma(i) < \sigma(j)$.

- **Two-stack sortable permutations**

A permutation $\sigma \in \mathfrak{S}_n$ is *two-stack sortable* if $\Pi(\Pi(\sigma)) = 12\ldots n$, *i.e.*, if it occurs in the last three columns of the sorting tree. West characterized these permutations in terms of forbidden patterns [22] and conjectured that their number is $b_n = 2(3n)!/[(2n+1)!(n+1)!]$. This conjecture was first proved by Zeilberger [23]. Two bijective proofs [10, 13] were found later, based on the fact that b_n is the number of non-separable planar maps [5, 7]. Note that the corresponding generating function $\sum b_n x^n$ is cubic over $\mathbb{R}(x)$.

- **Sorted permutations**

A permutation $\tau \in \mathfrak{S}_n$ is *sorted* if it belongs to $\Pi(\mathfrak{S}_n)$. In other words, the sorted permutations are the inner nodes of the sorting tree, or, using West's terminology [21], the nodes of *positive fertility*.

Characterizing and counting these permutations is the main motivation of this paper.

We shall give a linear algorithm that decides whether a permutation is sorted (and, in this case, exhibits one of its pre-images), and a functional equation satisfied by their generating function. So far, we have not been able to say whether this generating function is *D-finite* [19], or at least *differentiably algebraic* [2].

- **Sorted and (one-stack) sortable permutations**

We can describe these permutations by any of the three equivalent conditions:
– $\tau \in \Pi(\mathfrak{S}_n)$ and $\Pi(\tau) = 12\ldots n$,
– τ is the image by Π of a two-stack sortable permutation,
– τ is an inner node of one of the last two columns of the sorting tree.
We will show that their generating function is algebraic of degree 4.

One of the main tools of this paper is a factorisation of permutations, due to Zeilberger, that stabilizes the four classes of permutations described above: essentially, a permutation will be one-stack sortable (resp. two-stack sortable, sorted, sorted and sortable) if and only if its factors are one-stack sortable (resp. two-stack sortable, sorted, sorted and sortable). This property enables us to write, for each of these four classes, a functional equation defining its generating function.

1.2 Functional equations

The initial motivation of this work was the enumeration of sorted permutations. After various attempts, we realized that Zeilberger's factorisation could be applied to these permutations, and led to an unusual functional equation. It was then natural to ask whether the same factorisation, applied to other families of permutations, would also yield interesting functional equations. The answer turned out to be "yes", and we finally got very much interested in the equations themselves. This explains why this paper studies in parallel five families of permutations: general permutations, one-stack sortable permutations, two-stack sortable permutations, sorted permutations, and sorted and sortable permutations.

For each of them, we obtain a functional equation that defines implicitly a bivariate power series $F(x,y)$, and involves a divided difference

$$\Delta F(x,y) \stackrel{\text{def}}{=} \frac{F(x,y)-F(x,0)}{y}.$$

In all cases, we are mostly interested in $F(x,0)$; but there is no obvious way to derive from the equation that defines $F(x,y)$ an equation satisfied by the one-variable series $F(x,0)$.

Such equations are quite frequent in enumerative combinatorics. Examples can be found in the enumeration of permutations [4, 23, 14, p.532–534], of polygons [3, 11], and of maps [5, 7, 8, 9, 20]. To our knowledge, all examples that have been solved so far are polynomial in $F(x,y)$ and $F(x,0)$, and their solution is algebraic over the field $\mathbb{R}(x,y)$.

Three out of our five equations are polynomial in $F(x,y)$ and $F(x,0)$, and can be solved using previously known tools. The last two involve a partial derivative $\partial F/\partial x(x,y)$ (Proposition 4.1). They look very much like each other, but one of them is related to general permutations and has a rational solution, while the other is related to sorted permutations and will remain quite mysterious. However, we have found a method of deriving, from the functional-differential equation satisfied by $F(x,y)$, a (strange) equation satisfied by $F(x,0)$ (Proposition 5.3).

Finally, we will enrich our collection of equations with some q-analogs, obtained by enumerating our classes of permutations by their inversion number (or one of its variations).

1.3 Structure of the paper

In Section 2, we study the combinatorial properties of sorted permutations. In particular, we define a class of permutations (called *canonical* permutations) such that every sorted permutation has a unique canonical pre-image by Π. We also give a local characterization of canonical permutations, and a simple algorithm that decides whether a permutation is sorted. In Section 3 we describe a factorisation of permutations and show it is well-suited to the study of the sorting procedure. In Section 4, we establish and compare our five functional equations. We also give q-analogs of four of them. Section 5 is devoted to the solution of (some of) these equations.

2 Combinatorial properties of sorted permutations

2.1 Some examples

We begin this section with a few very simple remarks that should show some of the difficulties one meets when trying to characterize and count sorted permutations.

First of all, we observe that the last entry of a sorted permutation τ of $\mathfrak{S}_n$ is n. However, this condition is not sufficient to guarantee that τ is sorted, as shown by $\tau = 3214$, which is not sorted (see Fig.2). So, let us consider a permutation τ of $\mathfrak{S}_n$ ending with n, and let us write $\tau = \tau^L(n-1)\tau^R n$. If τ^R is not empty, then (1) implies that

- τ is sorted if and only if $\tau^L(n-1)$ and τ^R are sorted words;

- more precisely, the pre-images of τ are the permutations $\sigma^L n \sigma^R$ where $\Pi(\sigma^L) = \tau^L(n-1)$ and $\Pi(\sigma^R) = \tau^R$.

If τ^R is empty, *i.e.*, $\tau = \tau^L(n-1)n$, there is no obvious way of deciding whether τ is sorted or not. In particular, τ might be sorted while $\tau^L(n-1)$ is not sorted, as shown by $\tau = 32145 = \Pi(35241)$. Also, the permutation $\tau = 23145$ can be written $\Pi(\sigma^L)5$, with $\sigma^L = 2341$, or $\Pi(\sigma^L)\Pi(\sigma^R)5$, with $\sigma^L = 23$ and $\sigma^R = 14$. In other words, the pre-images of τ can give rise to different factorisations of τ of the form $\tau^L\tau^R n$, with τ^L and τ^R sorted.

The aim of this section is to fix the ambiguities illustrated by the above examples. In particular, we shall prove that, given a sorted permutation, one of its pre-images has strictly more inversions than all others (see an example on Fig.3). A permutation σ having more inversions than any other pre-image of $\Pi(\sigma)$ will be called *canonical*. We shall:

1) give a linear algorithm that decides whether a permutation is sorted, and in this case, builds its canonical pre-image,

2) give a local characterization of canonical permutations (which are obviously in one-to-one correspondence with sorted permutations).

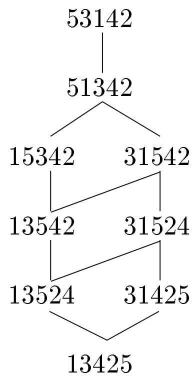


Figure 3: The pre-images of the sorted permutation $\tau = 13245$, ranked by their inversion number (the underlying order is the strong Bruhat order).

2.2 Permutations and trees

It will be convenient to represent permutations by trees. Let us begin with some terminology. A *decreasing binary tree* is a binary tree whose nodes are labelled by distinct positive integers in such a way that each node has a larger label than its children. The tree is said to be *normalized* if the number of its nodes coincides with the label of the root. The set of normalized trees having n nodes is denoted $\mathcal{T}_n$.

Reading a decreasing binary tree in *symmetric order* establishes a one-to-one correspondence with words on the alphabet $\{1, 2, \ldots\}$ having all their letters distinct. The symmetric order $S(t)$ of a tree t is defined recursively by reading first the left subtree of t, then its root, and finally its right subtree. In particular, S induces a standard bijection between normalized trees and permutations. The reverse bijection of S is denoted T (Fig.4).

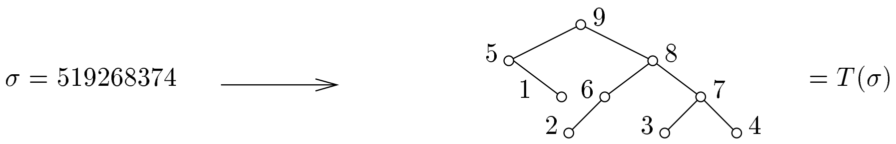


Figure 4: The bijection between permutations and normalized trees.

Let t be a decreasing binary tree having n nodes, and let L be the set of its labels. Let f be the unique order preserving bijection from L to $\{1, 2, \ldots, n\}$. *Normalizing* the tree t means replacing the label i by $f(i)$, for all $i \in L$. We define similarly the normalization of words having distinct letters.

We define recursively the *leftmost branch* and the *leftmost path* of a tree. If t^L (resp. t^R) is the left (resp. right) subtree of t, then the leftmost *branch* of t consists of the root of t and the leftmost branch of t^L. The leftmost *path* of t consists of the root of t and the leftmost path of t^L *if* t^L *is not empty*; otherwise, it consists of the root of t and the leftmost path of t^R. Hence the leftmost path joins the root to the "leftmost" leaf: for the tree of Fig.4, it consists of the nodes labelled $9, 5, 1$. We define symmetrically the rightmost branch and path.

We can now explain why we chose to represent permutations by trees. It turns out that displaying the entries of a permutation σ as the labels of the corresponding tree allows us to say at first glance what is the sorted permutation $\Pi(\sigma)$. Recall that the *postorder* $P(t)$ of a tree t is recursively defined by reading first the left subtree of t, then its right subtree, and finally its root. A simple comparison with the recursive definition of the sorting procedure (1) gives the following result.

Proposition 2.1 *Let σ be a permutation and $t = T(\sigma)$ the corresponding tree. Then the permutation $\Pi(\sigma)$ obtained by sorting σ through a stack is exactly the word $P(t)$ obtained by reading t in postorder. In other words, $\Pi = P \circ T$.*

This proposition relates the sorting procedure to a very basic operation of theoretical computer science. It also enables us to reformulate in terms of trees all questions related to the sorting procedure. In particular, it gives what is probably the simplest way of counting one-stack sortable permutations.

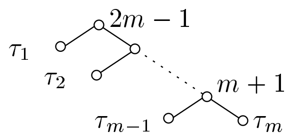
Corollary 2.2
1. *A permutation $\sigma \in \mathfrak{S}_n$ is one-stack sortable if and only if the associated tree $T(\sigma)$ has postorder $12\ldots n$. Consequently, the number of one-stack sortable permutations of length n is the Catalan number $\binom{2n}{n}/(n+1)$.*
2. *A permutation σ is two-stack sortable if and only if the postorder of $T(\sigma)$ avoids the pattern 231.*
3. *A permutation is sorted if and only if it is the postorder of a decreasing binary tree.*

Proof
1. The first assertion is obvious. By induction on the size of $T(\sigma)$, we observe that $P(T(\sigma)) = 12\ldots n$ if and only if σ avoids 231. To prove the second assertion, take an unlabelled binary tree, and label its vertices with $1, 2, \ldots, n$ by visiting them in postorder. We thus obtain a normalized tree whose postorder is $12\ldots n$.
2. A permutation σ is two-stack sortable if and only if $\Pi(\sigma)$ is one-stack sortable, *i.e.*, avoids the pattern 231.

■

A consequence of the above corollary is that sorted permutations cannot be described by forbidding a set of patterns.

Corollary 2.3 *Any pattern occurs as a factor in some sorted permutation. More precisely, if $\tau = \tau_1 \ldots \tau_m \in \mathfrak{S}_m$, then the permutation $\tau_1 \ldots \tau_m (m+1)(m+2)\ldots(2m-1)$ is sorted (see the figure below).*



In the enumeration of sorted permutations, we shall take into account the inversion number. The following lemma explains how to determine the inversion number of a sorted permutation from one of its pre-images.

Lemma 2.4 (West [22]) *Let σ be a permutation. We define $\underline{\mathrm{inv}}(\sigma)$ to be the number of pairs (i, k) where $i < k$ such that there exists $j \in [i, k]$ such that $\sigma(k) < \sigma(i) < \sigma(j)$. Then $\underline{\mathrm{inv}}(\sigma)$ is the number of inversions of $\Pi(\sigma)$.*

Using Rawlings' notations [16], we could call $\underline{\mathrm{inv}}(\sigma)$ the *number of* $2\underline{3}1$ *patterns.* For instance, the permutation $\sigma = 2351674$ has four $2\underline{3}1$ patterns (corresponding to the pairs of letters $(2,1)$, $(3,1)$, $(5,4)$ and $(6,4)$) and $\Pi(\sigma) = 2315647$ has four inversions (given by the same pairs of letters).

2.3 Canonical permutations

Clearly, different trees might have the same postorder (Fig.5). In order to characterize sorted permutations, we are going to describe a *canonical representative* of the pre-images of a sorted permutation.

Definition 2.5 *A permutation σ is said to be* canonical *if the tree $T(\sigma)$ satisfies the following properties:*

- *each node that has a left child x has a nonempty right subtree t^R;*
- *moreover, the first node of t^R (for the symmetric order) has a label y smaller than x.*

We shall say that a tree t is canonical (resp. one-stack sortable, two-stack sortable) if the permutation $\sigma = S(t)$ is canonical (resp. one-stack sortable, two-stack sortable).

Examples. The first tree of Fig.5 is not canonical because the left child of the node 5 has label $x = 1$, whereas the first node of its right subtree has label $y = 3 > 1$. The second tree of the figure is canonical.

Figure 5: Two trees having postorder 13245.

The following proposition implies that the procedure Π induces a bijection between canonical permutations and sorted permutations.

Proposition 2.6 *Any sorted permutation τ has a unique canonical pre-image σ. Moreover, σ has strictly more inversions than any other pre-image of τ.*

Proof. We begin by proving that at least one of the pre-images of τ is canonical, *i.e.*, that τ is the postorder of at least one canonical tree.

As τ is sorted, we know there exists a tree u whose postorder is τ. If u is canonical, we are done. Otherwise, we can perform on u at least one of the following transformations.

First transformation. If u has a node z that has a left child but no right child, we transform the left subtree of z into its right subtree.
Second transformation. If u has a node z that has a left child x and a nonempty right subtree t^R whose first node (in symmetric order) is $y > x$, we remove the left subtree of z and attach it as the left subtree of y.

We note that both transformations
– give a decreasing tree,
– do not change the postorder,
– increase the inversion number of the permutation obtained by reading the tree in symmetric order.
These properties imply that repeating these transformations in any order will finally provide a canonical tree whose postorder is τ, having strictly more inversions than u. Observe that the first transformation is somehow a limit case of the second one.

Let us now prove by induction on the length n of τ that τ has a *unique* canonical pre-image. If $n = 0$ or $n = 1$, the result is obvious. Otherwise, let x be the first letter of τ, and write $\tau = x\tau'$. Let t be a canonical tree whose postorder is τ. Then x labels a leaf of t. Moreover, removing this leaf gives a *canonical* tree t' whose postorder is τ'. By the induction hypothesis, t' is the *unique* canonical tree of postorder τ'. Let us prove that the position of the leaf x in the tree t is also uniquely determined.

Let z be the father of x in t. Then:

1) z must be a vertex of the leftmost path of t' having no left child (because the postorder of t must start with x);

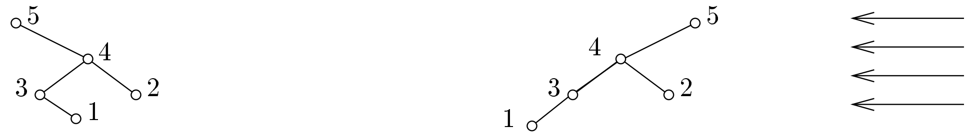
2) z must be larger than x;

3) all vertices of the leftmost path of t' having no left child that lie below z must have labels smaller than x (as t must be canonical).

These three conditions determine at most one vertex of t': the *smallest node of the leftmost path of t' that is larger than x and has no left child.* We know that τ has at least one canonical pre-image: this guarantees the existence of this node z. If z is a leaf of t', then x will be its right child. Otherwise, x will be its left child.

■

Remark. We can also prove that any sorted permutation τ has a (unique) pre-image σ' having strictly *fewer* inversions than all others (Fig.3). The corresponding tree $T(\sigma')$ is, among all trees having postorder τ, the only one that satisfies the following property: each node having a nonempty right subtree t^R has a left child x. Moreover, the first node of t^R (for the symmetric order) has a label y smaller than x. This tree is obtained from the canonical tree of τ when a strong wind blows from the east: if z is a node having no left child, then the right subtree of z becomes its left subtree. For instance, Fig.6 shows the canonical tree of postorder $\tau = 13245$ and its windy version.



$\sigma = 53142$ $\sigma' = 13425$

Figure 6: The pre-images of $\tau = 13245$ having the largest (resp. smallest) inversion number.

The proof of Proposition 2.6 has an interesting consequence, which concerns the number of pre-images of a sorted permutation, called *fertility* by West [21].

Proposition 2.7 *The number of pre-images of a sorted permutation only depends on the* shape *of its canonical pre-image σ,* i.e., *on the binary tree obtained by removing the labels from $T(\sigma)$.*

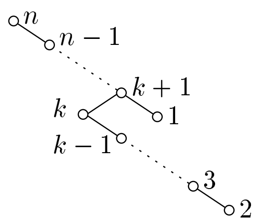
Proof. Starting from the canonical tree of postorder τ, we construct all other pre-images of τ by reversing the first and second transformations described in the proof of Proposition 2.6. The two reverse transformations can be described in unified terms as follows.

Reverse transformation. Assume the tree u has a node z having a nonempty right subtree t^R but no left child. Let x be a vertex of the leftmost branch of t^R. Remove the subtree of root x and append it as the left subtree of z. Label this transformation by the pair (z, x).

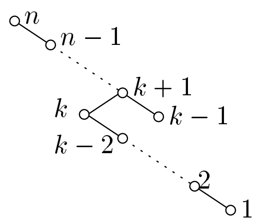
The set of trees of postorder τ is obtained by applying this reverse transformation any number of times, in any order, starting from the canonical tree of τ. We observe that the transformations one can perform on a tree u do not depend on the labels of u, but only on its shape. Fig.7 shows the set of trees having postorder $\tau = 13245$. The edges are labelled by the pairs (z, x).

■

Remarks
1. West proved [21, p.94] that the permutations $\mu_{n,k} = 23\ldots k\ 1(k+1)(k+2)\ldots n$ and $\nu_{n,k} = 12\ldots(k-2)k(k-1)(k+1)\ldots n$ have the same number of pre-images. This is a consequence of the above proposition, as the corresponding canonical trees are respectively



and



and have the same shape.

2. It would be interesting to determine, other than recursively, the number of pre-images of a sorted permutation from the shape of its canonical tree.

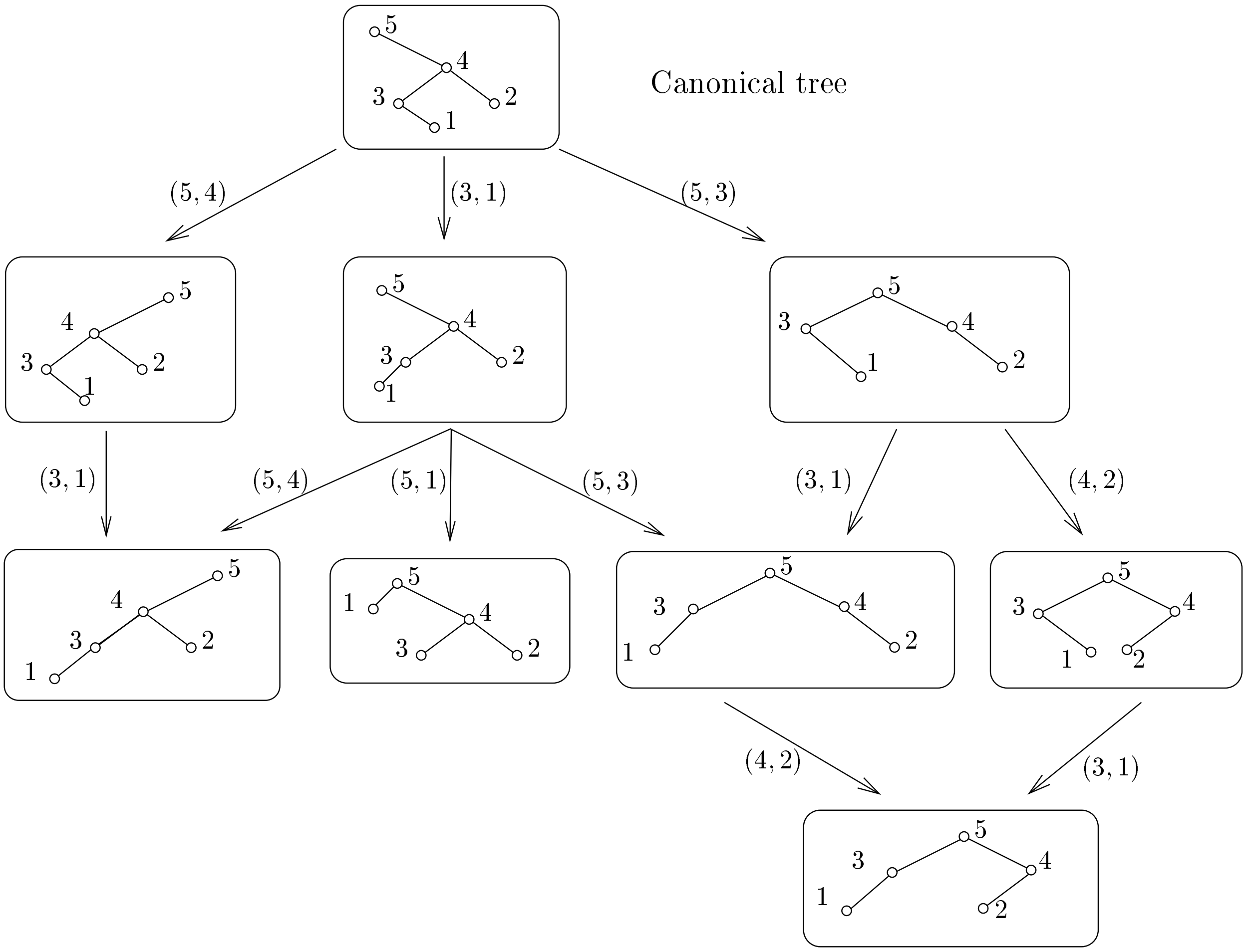
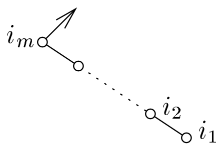


Figure 7: The trees having postorder $\tau = 13245$.

2.4 An algorithm that decides whether a permutation is sorted

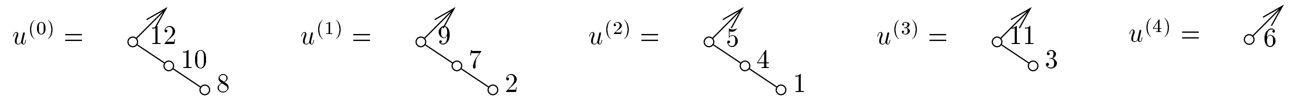
In the proof of Proposition 2.6, we have described how the unique canonical tree having postorder τ can be constructed in an iterative way, by reading τ from right to left, and adding a leaf to the tree at each step. We give below a more concise description of this construction by adding at the same time all nodes that belong to the same increasing factor of τ.

Assume that τ has k descents and write $\tau = \tau^{(k)}\tau^{(k-1)}\dots\tau^{(0)}$ where the $\tau^{(j)}$ are the maximal increasing factors of τ. For $0 \le j \le k$, if $\tau^{(j)} = i_1 \dots i_m$, with $i_1 < i_2 < \cdots < i_m$, let $u^{(j)}$ be the (linear) tree $T(i_m i_{m-1} \dots i_1)$:

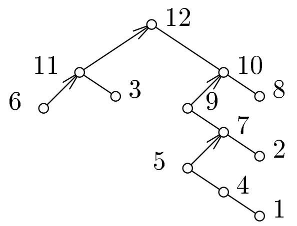
$u^{(j)} =$ 

Observe the arrow attached to the root of $u^{(j)}$, and note that $P(u^{(j)}) = \tau^{(j)}$. We now build canonical trees $t^{(0)}, t^{(1)}, \dots$ as follows.
Step 0**.** Start from the tree $t^{(0)} = u^{(0)}$.
Step i**,** $i = 1, \dots, k$**.** If all nodes of the leftmost path of the tree $t^{(i-1)}$ that have no left child are smaller than the root of $u^{(i)}$, then τ is not sorted and we stop. Otherwise, let $t^{(i)}$ be obtained by attaching $u^{(i)}$ to the smallest node in the leftmost path of $t^{(i-1)}$ that is larger than the root of $u^{(i)}$ and has no left child.
The tree $t^{(k)}$ (when we can construct it) is the canonical pre-image of τ.

Example. Let $\tau = 6.3.11.1.4.5.2.7.9.8.10.12 \in \mathfrak{S}_{12}$. This permutation has $k = 4$ descents, and we obtain the following elementary trees:



We can attach them to each other, step by step; we finally obtain the canonical pre-image of τ:



.

Hence $\tau = \Pi(6.11.3.12.9.5.4.1.7.2.10.8)$ is sorted.

3 Zeilberger's factorisation of permutations

3.1 Factoring permutations

We shall extend to all permutations the factorisation of two-stack sortable permutations described by Zeilberger in [23]. It requires the introduction of a new statistic. For $\sigma \in \mathfrak{S}_n$, we define $z(\sigma)$ by:

$$z(\sigma) = \max\{\ell : \sigma^{-1}(n) < \sigma^{-1}(n-1) < \cdots < \sigma^{-1}(n-\ell+1)\}.$$

For instance, $z(519268374) = 3$. If σ is the empty permutation, of length 0, we set $z(\sigma) = 0$. For $m, n \geq 0$, we define the sets $\mathfrak{S}_{m,n}$ and $\overline{\mathfrak{S}}_{m,n}$ by

$$\mathfrak{S}_{m,n} = \{\sigma \in \mathfrak{S}_{m+n} : z(\sigma) \geq n\} \qquad \text{and} \qquad \overline{\mathfrak{S}}_{m,n} = \{\sigma \in \mathfrak{S}_{m+n} : z(\sigma) = n\}.$$

Note that $\mathfrak{S}_{m,0} = \mathfrak{S}_m$ and that for $m \geq 1$, $\mathfrak{S}_{m,n}$ is the disjoint union of $\mathfrak{S}_{m-1,n+1}$ and $\overline{\mathfrak{S}}_{m,n}$.

The principle of the factorisation is very simple: it splits a permutation into two factors, a prefix and a suffix. Let $m, n \geq 1$ and take $\sigma \in \overline{\mathfrak{S}}_{m,n}$. This means that σ has length $m+n$, that the numbers $m+n, m+n-1, \ldots, m+1$ appear in this order in σ, and that m lies to the left of $m+1$. Let $j \in \{0, 1, \ldots, n-1\}$ be the largest number such that m lies to the left of $m+j+1$. We have

$$\sigma = \ldots(m+n)\ldots(m+n-1)\ldots\ldots(m+j+2)\ldots(m)\ldots(m+j+1)\ldots(m+j)\ldots\ldots(m+1)\ldots$$

Let us write $\sigma = \sigma^{(1)}(m+j+1)\sigma^{(2)}$. The length of $\sigma^{(2)}$ is $i+j$ for some $i \in \{0, 1, \ldots, m-1\}$. Let L be the set of numbers smaller than m occurring in $\sigma^{(2)}$. Then L has cardinality i. Finally, let σ_1 (resp. σ_2) be obtained by normalizing $\sigma^{(1)}$ (resp. $\sigma^{(2)}$). Note that $\sigma_1 \in \mathfrak{S}_{m-i-1,n-j}$ and $\sigma_2 \in \mathfrak{S}_{i,j}$. Let us denote $\Phi(\sigma) = (i, j, L, \sigma_1, \sigma_2)$.

Example. Let $m = 6$ and $n = 3$. For $\sigma = 519268374 \in \overline{\mathfrak{S}}_{6,3}$ we find $j = 1$, $\sigma^{(1)} = 51926$ and $\sigma^{(2)} = 374$. We have $i = 2$ and $L = \{3, 4\}$. Normalizing the permutations gives $\sigma_1 = 31524$ and $\sigma_2 = 132$, and finally $\Phi(\sigma) = (2, 1, \{3, 4\}, 31524, 132)$.

We obtain by inspection the following proposition.

Proposition 3.1 *For $m, n \geq 1$, the map Φ establishes a one-to-one correspondence between $\overline{\mathfrak{S}}_{m,n}$ and the five-tuples $(i, j, L, \sigma_1, \sigma_2)$ such that*

$$0 \leq i < m, \quad 0 \leq j < n, \quad L \subset \{1, \ldots, m-1\}, \quad |L| = i, \quad \sigma_1 \in \mathfrak{S}_{m-i-1,n-j} \quad \textit{and} \quad \sigma_2 \in \mathfrak{S}_{i,j}.$$

Moreover, if $\Phi(\sigma) = (i, j, L, \sigma_1, \sigma_2)$, then

$$\underline{\mathrm{inv}}(\sigma) = \underline{\mathrm{inv}}(\sigma_1) + \underline{\mathrm{inv}}(\sigma_2) + \mathrm{inv}(m, L)$$

where $\mathrm{inv}(m, L) = |\{(a, b) : a \in [1, m] \setminus L,\ b \in L,\ a > b\}|$.

Example. For the permutation σ of the previous example, we have $\underline{\text{inv}}(\sigma) = 5$, $\underline{\text{inv}}(\sigma_1) = 1$, $\underline{\text{inv}}(\sigma_2) = 0$, $\text{inv}(6, L) = 4$ and we check that $\underline{\text{inv}}(\sigma_1) + \underline{\text{inv}}(\sigma_2) + \text{inv}(m, L) = 1 + 0 + 4 = \underline{\text{inv}}(\sigma) = 5$.

Remark. Several other standard statistics can be carried through our factorisation of permutations. See for instance [4] for the enumeration of two-stack sortable permutations, using this factorisation, according to the length, number of descents, number of left-to-right and right-to-left maxima. The inversion number satisfies

$$\text{inv}(\sigma) = \text{inv}(\sigma_1) + \text{inv}(\sigma_2) + \text{inv}(m, L) + (n - j)(i + j + 1) - 1,$$

and this kind of relation does not give simple functional equations.

3.2 Factoring trees

Let us now describe the factorisation in terms of trees. First of all, we note that the statistic $z(\sigma)$ is easily determined from the tree $t = T(\sigma)$: if t has n nodes, then $z(\sigma)$ is the largest ℓ such that $n, n-1, \ldots, n-\ell+1$ lie on the rightmost branch of t. When we do not want to make the underlying permutation explicit, we will use the notation $z(t)$ instead of $z(\sigma)$. By analogy with $\mathfrak{S}_{m,n}$ and $\overline{\mathfrak{S}}_{m,n}$, we define, for $m, n \geq 0$, the sets $\mathcal{T}_{m,n}$ and $\overline{\mathcal{T}}_{m,n}$ by

$$\mathcal{T}_{m,n} = \{t \in \mathcal{T}_{m+n} : z(t) \geq n\} \qquad \text{and} \qquad \overline{\mathcal{T}}_{m,n} = \{t \in \mathcal{T}_{m+n} : z(t) = n\}.$$

Let $m, n \geq 1$ and take $t \in \overline{\mathcal{T}}_{m,n}$. This means that the nodes $m+n, m+n-1, \ldots, m+1$ lie on the rightmost branch of t, and that m is the left child of one of them – say, of $m+j+1$, with $0 \leq j < n$. Let $t^{(2)}$ be the right subtree of the node $m+j+1$. Let $i+j$ be the number of its nodes, and L the set of its labels smaller than m. Then $|L| = i$. Let $t^{(1)}$ be obtained from t by replacing the subtree of root $m+j+1$ by the subtree of root m. Let t_1 (resp. t_2) be obtained by normalizing $t^{(1)}$ (resp. $t^{(2)}$). Define $\Phi(t) = (i, j, L, t_1, t_2)$. Then Φ establishes a one-to-one correspondence between $\overline{\mathcal{T}}_{m,n}$ and the five-tuples (i, j, L, t_1, t_2) such that

$$0 \leq i < m, \quad 0 \leq j < n, \quad L \subset \{1, \ldots, m-1\}, \quad |L| = i, \quad t_1 \in \mathcal{T}_{m-i-1, n-j} \;\text{ and }\; t_2 \in \mathcal{T}_{i,j}.$$

The factorisation of trees is schematized in Fig.8.

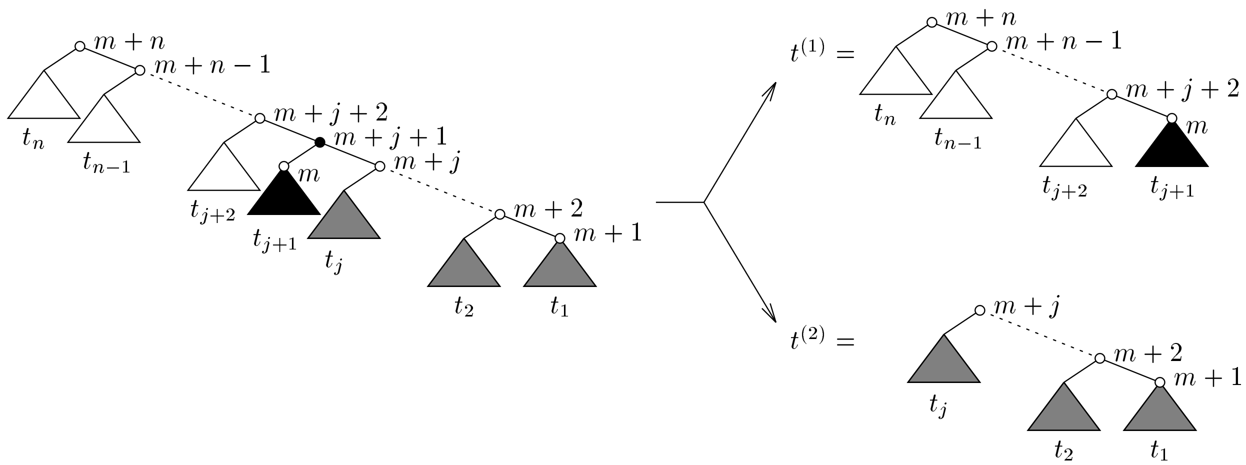


Figure 8: The factorisation of trees (the set of labels of the grey trees is L).

3.3 Recursive characterizations

The following proposition provides recursive characterizations for one-stack sortable permutations, two-stack sortable permutations and canonical permutations. We shall use it to obtain, in the next section, our functional equations.

Proposition 3.2 *Let σ be a permutation of $\overline{\mathfrak{S}}_{m,n}$, where $m, n \geq 1$, and let $t = T(\sigma)$ be the corresponding normalized tree. Let (i, j, L, t_1, t_2) be the five-tuple obtained by factoring t.*

1) σ *is one-stack sortable if and only if* $L=\emptyset$ *and* t_1 *and* t_2 *are one-stack sortable.*
2) σ *is two-stack sortable if and only if* $L=\{m-i, m-i+1, \ldots, m-1\}$ *and* t_1 *and* t_2 *are two-stack sortable.*
3) σ *is canonical if and only if* t_1 *and* t_2 *are canonical and either*
- $j=0$ *and* t_2 *is nonempty, or*
- $j>0$ *and* t_2 *has a nonempty left subtree.*

In particular, if σ *is canonical, then* $i>0$.

Proof. We use the pictorial description of the factorisation (Fig.8). Observe that

$$P(T(\sigma)) = P(t_n)\cdots P(t_1)(m+2)\cdots(m+n),$$

$$P(t^{(1)}) = P(t_n)\cdots P(t_{j+1})(m+j+2)\cdots(m+n) \quad \text{and} \quad P(t^{(2)}) = P(t_j)\cdots P(t_1)(m+2)\cdots(m+j).$$

We conclude using Corollary 2.2 for the first two characterizations and Definition 2.5 for the last one.

■

4 Functional equations

In this section, we establish and compare five functional equations that define implicitly the generating functions for the following five families of permutations: general permutations, one-stack sortable permutations, two-stack sortable permutations, sorted permutations and sorted and sortable permutations. These functional equations are derived from the factorisation of permutations described in the previous section.

Notations. We shall use the following standard definitions and notations. For $n \geq 1$, the q-analog of n is

$$[n] = 1+q+\cdots+q^{n-1} = \frac{1-q^n}{1-q}.$$

The q-analog of $n!$ is $[n]! = [1][2]\ldots[n]$. By convention, $[0]! = 1$. Finally, for $0 \leq k \leq n$, the q-analog of the binomial coefficient $\binom{n}{k}$ is

$$\begin{bmatrix} n \\ k \end{bmatrix} = \frac{[n]!}{[k]![n-k]!}.$$

Proposition 3.1 explains our interest in the following classical interpretation of the q-binomial coefficient:

$$\sum_{L\subset\{1,\ldots,m-1\}:|L|=i} q^{\mathrm{inv}(m,L)} = q^i \begin{bmatrix} m-1 \\ i \end{bmatrix}.$$

Let $\mathcal{F}$ be a set of permutations. By the *ordinary* (resp. *exponential*) *generating function of* $\mathcal{F}$ we mean the two-variable series

$$F(x,y) = \sum_{m,n\geq 0} f_{m,n}\, x^m y^n \qquad \left(\text{resp.} \quad F(x,y) = \sum_{m,n\geq 0} f_{m,n}\, \frac{x^m}{m!}\, y^n\right),$$

where $f_{m,n} = |\mathcal{F}\cap\mathfrak{S}_{m,n}|$ is the number of permutations σ of $\mathcal{F}$ of length $m+n$ such that $z(\sigma)\geq n$. Similarly, let $\mathtt{inv}$ denote any of the statistics inv (the inversion number) or $\underline{\mathrm{inv}}$. The *ordinary* (resp. *Eulerian*) $\mathtt{inv}$-*generating function of* $\mathcal{F}$ is

$$\bar{F}(x,y) = \sum_{m,n\geq 0} \bar{f}_{m,n}\, x^m y^n \qquad \left(\text{resp.} \quad \bar{F}(x,y) = \sum_{m,n\geq 0} \bar{f}_{m,n}\, \frac{x^m}{[m]!}\, y^n\right),$$

where $\bar{f}_{m,n} = \sum_{\sigma\in\mathcal{F}\cap\mathfrak{S}_{m,n}} q^{\mathtt{inv}(\sigma)}$. Observe that $F(x,0)$ and $\bar{F}(x,0)$ are respectively the length generating function and the length+$\mathtt{inv}$ generating function for the permutations of $\mathcal{F}$.

Proposition 4.1 *Zeilberger's factorisation, applied to our five classes of permutations, yields the following functional equations.*
Linear equation. *The ordinary generating function $A(x,y)$ for one-stack sortable permutations is completely characterized by the equation*

$$A(x,y) = \frac{1}{1-y} + \frac{x}{1-y}\ \frac{A(x,y)-A(x,0)}{y}. \tag{2}$$

Quadratic equations. *The ordinary generating functions for two-stack sortable permutations and for sorted and sortable permutations (respectively, $B(x,y)$ and $C(x,y)$) are completely characterized by the equations*

$$B(x,y) = \frac{1}{1-y} + x\,[1+yB(x,y)]\ \frac{B(x,y)-B(x,0)}{y}, \tag{3}$$

$$C(x,y) = \frac{1}{1-y} + x(1-y)\,[1+yC(x,y)]\ \frac{C(x,y)-C(x,0)}{y}. \tag{4}$$

Differential equations. *The exponential generating functions for general permutations and for sorted permutations (respectively $D(x,y)$ and $E(x,y)$) are completely characterized by the equations*

$$\frac{\partial D}{\partial x}(x,y) = [1+yD(x,y)]\ \frac{D(x,y)-D(x,0)}{y}, \tag{5}$$

$$\frac{\partial E}{\partial x}(x,y) = (1-y)\,[1+yE(x,y)]\ \frac{E(x,y)-E(x,0)}{y}, \tag{6}$$

and the initial conditions $D(0,y) = E(0,y) = 1/(1-y)$.

We delay the proof of this proposition to make a few comments.

1. The series $A(x,y), B(x,y), C(x,y), D(x,y)$ and $E(x,y)$ are uniquely defined by these equations: in each of these series, the coefficient of x^n is a rational function in y that can be computed by induction on n using the relevant equation. In particular, we obtain for sorted permutations and for sorted and sortable permutations of length at most 30 the data presented in Table 1.

2. The five equations involve a common factor: a discrete derivative (or divided difference)

$$\Delta F(x,y) \stackrel{\text{def}}{=} \frac{F(x,y)-F(x,0)}{y}.$$

As we wrote in the introduction, such equations arise frequently in enumerative combinatorics. Observe that there is no obvious way to derive an equation satisfied by $F(x,0)$ itself.

3. Two pairs of equations are very similar, and only differ by a factor $(1-y)$. Equation (3) is equivalent to the equation obtained by Zeilberger for two-stack sortable permutations [23].

4. The series $D(x,y)$ has an extremely simple expression. Let $d_{m,n}$ be the number of permutations $\sigma \in \mathfrak{S}_{m+n}$ such that $z(\sigma) \geq n$. Clearly, $d_{m,n} = (m+n)!/n!$ (shuffle the word $(m+n)(m+n-1)\ldots(m+1)$ with any permutation of $\mathfrak{S}_m$). Consequently, the exponential generating function for general permutations is $D(x,y) = 1/(1-x-y)$. It is very easy to check that $1/(1-x-y)$ satisfies (5), but how can one *derive* this rational expression from (5)?

5. Using Proposition 3.1, we can also take into account the statistics $\underline{\text{inv}}$ in the factorisation of permutations. We thus obtain for four of our equations a nice q-analog.

Proposition 4.2 *The equations of Proposition* 4.1 *admit the following q-analogs.*
Quadratic equations. *The ordinary* $\underline{\text{inv}}$*-generating function for two-stack sortable permutations and the ordinary* inv*-generating function for sorted and sortable permutations (respectively, $\bar{B}(x,y)$ and $\bar{C}(x,y)$) are completely characterized by the equations*

$$\bar{B}(x,y) = \frac{1}{1-y} + x\left[1+y\bar{B}(xq,y)\right]\ \frac{\bar{B}(x,y)-\bar{B}(x,0)}{y}, \tag{7}$$

$$\bar{C}(x,y) = \frac{1}{1-y} + x(1-y)\left[1+y\bar{C}(xq,y)\right]\ \frac{\bar{C}(x,y)-\bar{C}(x,0)}{y}. \tag{8}$$

***q*-Differential equations.** *The Eulerian $\underline{\text{inv}}$-generating function for general permutations and the Eulerian* inv*-generating function for sorted permutations (respectively $\bar{D}(x,y)$ and $\bar{E}(x,y)$) are completely characterized by the equations*

$$\frac{\bar{D}(x,y)-\bar{D}(xq,y)}{x(1-q)} = \left[1+y\bar{D}(xq,y)\right]\ \frac{\bar{D}(x,y)-\bar{D}(x,0)}{y}, \tag{9}$$

$$\frac{\bar{E}(x,y)-\bar{E}(xq,y)}{x(1-q)} = (1-y)\left[1+y\bar{E}(xq,y)\right]\ \frac{\bar{E}(x,y)-\bar{E}(x,0)}{y}, \tag{10}$$

and the initial conditions $\bar{D}(0,y)=\bar{E}(0,y)=1/(1-y)$.

Remarks

1. Clearly, the last four equations of Proposition 4.1 are obtained from Proposition 4.2 in the limit case $q=1$. Enumerating one-stack sortable permutations according to the statistic $\underline{\text{inv}}$ is irrelevant, as these permutations avoid the pattern 231. For their enumeration according to the number of inversions, see [1].

2. We obtain a different information on the series $\bar{D}(x,y)$ (general permutations) if we use the standard factorisation of trees into their left and right subtrees. We find:

$$\frac{\bar{D}(x,0)-\bar{D}(xq,0)}{x(1-q)} = \bar{D}(x,0)^2 \qquad \text{and} \qquad \bar{D}(x,y) = \bar{D}(x,0)\left[1+y\bar{D}(x,y)\right]. \tag{11}$$

One checks easily that (11) implies (9). But conversely, deriving (11) from (9) does not seem so simple. Note that Rawlings [16, 17] has studied a close relative to the statistics $\underline{\text{inv}}$, and essentially obtained the first equation in (11).

Proof of Propositions 4.1 and 4.2

1. We begin with the enumeration of general permutations. Let $\bar{d}(m,n)$ denote the polynomial in q that counts permutations of $\mathfrak{S}_{m,n}$ according to the statistics $\underline{\text{inv}}$.

The set $\mathfrak{S}_{0,n}$ is reduced to $\{n(n-1)\dots 1\}$ and hence $\bar{d}_{0,n}=1$ for $n\geq 0$. This gives $\bar{D}(0,y)=1/(1-y)$. Moreover, for $m\geq 1$, we have $\mathfrak{S}_{m,n}=\mathfrak{S}_{m-1,n+1}\cup\overline{\mathfrak{S}}_{m,n}$ and Proposition 3.1 gives:

$$\bar{d}_{m,n} = \bar{d}_{m-1,n+1} + \sum_{i=0}^{m-1}\sum_{j=0}^{n-1} \begin{bmatrix} m-1 \\ i \end{bmatrix} q^i\ \bar{d}_{m-i-1,n-j}\ \bar{d}_{i,j}. \tag{12}$$

Multiplying by $y^n x^{m-1}/[m-1]!$ and summing on $m\geq 1$ and $n\geq 0$ gives the result.

2. For one-stack sortable permutations, we use Proposition 3.2 to obtain an analog of Eq. (12). Let $a_{m,n}$ be the number of one-stack sortable permutations σ of length $m+n$ such that $z(\sigma)\geq n$. Then for $m\geq 1$,

$$a_{m,n} = a_{m-1,n+1} + \sum_{j=0}^{n-1} a_{m-1,n-j}\ a_{0,j}.$$

Using $a_{0,j}=1$ and summing on m and n gives the result.

3. For two-stack sortable permutations, we find, for $m\geq 1$,

$$\bar{b}_{m,n} = \bar{b}_{m-1,n+1} + \sum_{i=0}^{m-1}\sum_{j=0}^{n-1} q^i\ \bar{b}_{m-i-1,n-j}\ \bar{b}_{i,j},$$

where $\bar{b}_{m,n}$ is the polynomial in q that counts two-stack sortable permutations of $\mathfrak{S}_{m,n}$ according to the statistics $\underline{\text{inv}}$. Again, $\bar{b}_{0,n}=1$ and we obtain our functional equation by summing on m and n.

4. Counting sorted permutations according to their inversions is equivalent to counting canonical permutations according to the statistic $\underline{\text{inv}}$ (see Proposition 2.6 and Lemma 2.4). Using Proposition 3.2, we find, for $m\geq 1$,

$$\bar{e}_{m,n} = \bar{e}_{m-1,n+1} + \sum_{i=1}^{m-1}\sum_{j=0}^{n-1} \begin{bmatrix} m-1 \\ i \end{bmatrix} q^i\ \bar{e}_{m-i-1,n-j}\ (\bar{e}_{i,j}-\bar{e}_{i,j-1}),$$

with the convention $\bar{e}_{i,-1} = 0$. In the above equation, $\bar{e}_{m,n}$ denotes the polynomial in q that counts sorted permutations of $\mathfrak{S}_{m,n}$ according to the statistics inv. We use $\bar{e}_{0,n} = 1$, multiply by $y^n x^{m-1}/[m-1]!$ and sum on m and n to obtain the result.

5. Counting sorted and sortable permutations according to their inversions is equivalent to counting two-stack sortable canonical permutations according to the statistic $\underline{\text{inv}}$. Hence, we need to combine two of the properties we have already studied. We find, for $m \geq 1$,

$$\bar{c}_{m,n} = \bar{c}_{m-1,n+1} + \sum_{i=1}^{m-1} \sum_{j=0}^{n-1} q^i \; \bar{c}_{m-i-1,n-j} \; (\bar{c}_{i,j} - \bar{c}_{i,j-1}),$$

with the convention $\bar{c}_{i,-1} = 0$. In the above equation, $\bar{c}_{m,n}$ denotes the polynomial in q that counts sorted and sortable permutations of $\mathfrak{S}_{m,n}$ according to the statistics inv. We sum on m and n to obtain the result.

■

Length	Sorted	Sorted and sortable	Length	Sorted	Sorted and sortable
1	1	1	16	48729809104	1599816
2	1	1	17	576039659209	5212650
3	2	2	18	7213070102518	17098590
4	5	4	19	95373808983223	56473664
5	17	10	20	1327842798808220	187572584
6	68	25	21	19416307366048221	626430568
7	326	69	22	297499363267839558	2101977231
8	1780	192	23	4766432683120731044	7084963950
9	11033	562	24	79699553284422816437	23976649328
10	76028	1663	25	1388383661114307067780	81447876258
11	578290	5065	26	25156549558328842669336	277627821135
12	4803696	15592	27	473403195053530875676679	949393445553
13	43297358	48874	28	9239492647978583159102374	3256266981128
14	420639362	154651	29	186785371461376448191242175	11199653726786
15	4382320595	495418	30	3906561056937710831259467950	38620292110925

Table 1. The number of sorted (resp. sorted and sortable) permutations.

5 Solving the functional equations

The five functional equations we have obtained are of three different sorts. The simplest one is related to one-stack sortable permutations. It is linear in $A(x, y)$. Two others are (q-)quadratic in the unknown series. They are related to two-stack sortable permutations and sorted and sortable permutations respectively. The last two equations involve a (q-)derivative with respect to x.

Notations. Given a ring $\mathbb{L}$ and n indeterminates $x_1, \ldots, x_n$, we denote by
- $\mathbb{L}[x_1, \ldots, x_n]$ the ring of polynomials in $x_1, \ldots, x_n$ with coefficients in $\mathbb{L}$,
- $\mathbb{L}[[x_1, \ldots, x_n]]$ the ring of formal power series in $x_1, \ldots, x_n$ with coefficients in $\mathbb{L}$,

and if $\mathbb{L}$ is a field, we denote by
- $\mathbb{L}(x_1, \ldots, x_n)$ the field of rational functions in $x_1, \ldots, x_n$ with coefficients in $\mathbb{L}$.

5.1 Linear equations and the kernel method

Proposition 5.1 (Knuth [14]) *The ordinary length generating function $A(x, 0)$ for one-stack sortable permutations is:*

$$A(x, 0) = \frac{1 - \sqrt{1 - 4x}}{2x} = \sum_{n \geq 0} \frac{1}{n+1} \binom{2n}{n} x^n.$$

Proof. We use a method, sometimes called the *kernel method*, that can be found in several papers, *e.g.* [9, 11, 14, p.532]. Equation (2) can be rewritten as

$$[y(1-y) - x] A(x, y) = y - x A(x, 0).$$

Let $Y = (1 - \sqrt{1-4x})/2 = x + O(x^2)$. Then $Y(1-Y) = x$. Substituting Y for y in the above equation shows that $A(x, 0) = Y/x$. Of course, we could also write an algebraic expression for $A(x, y)$.

■

5.2 Quadratic equations and the quadratic method

The equations (3) for two-stack sortable permutations and (4) for sorted and sortable permutations can be solved via the so-called *quadratic method*, which is due to Brown [6, 12, section 2.9.1].

Proposition 5.2 *The ordinary length generating function $B_0(x) = B(x, 0)$ for two-stack sortable permutations is cubic over the field $\mathbb{R}(x)$:*

$$x^2 B_0(x)^3 + x(2+3x)B_0(x)^2 + (1 - 14x + 3x^2)B_0(x) + x^2 + 11x - 1 = 0.$$

This implies that

$$B_0(x) = 1 + 2\sum_{n\geq 1} \frac{(3n)!}{(2n+1)!(n+1)!} x^n.$$

The ordinary length generating function $C_0(x) = C(x, 0)$ for sorted and sortable permutations is algebraic of degree 4:

$$x^3 C_0(x)^4 + x^2(3+4x)C_0(x)^3 + x(3 - 29x + 6x^2)C_0(x)^2 + (1 - 7x + 29x^2 + 4x^3)C_0(x) - (1-x)^3 = 0.$$

Proof. In Eq. (3), let us form a perfect square containing all powers of $B(x, y)$:

$$(y-1)\left[2xyB(x,y) - xyB_0(x) + x - y\right]^2 = \Delta(y) \tag{13}$$

where $\Delta(y)$ is the following polynomial in y with coefficients in $\mathbb{R}(x, B_0(x))$:

$$\Delta(y) = [1 + xB_0(x)]^2 y^3 - [1 - 2x + xB_0(x)][1 + xB_0(x)]y^2 - x[2xB_0(x) - x - 2]y - x^2.$$

Let $Y = Y(x) = x + x^2 + O(x^3)$ be the (unique) power series in x such that $Y = 2xYB(x, Y) - xYB_0(x) + x$. Substituting $Y(x)$ for y in (13) shows that Δ has a double root at $y = Y(x)$. This implies that the resultant of Δ and $\partial\Delta/\partial y$, seen as polynomials in y, is zero. Computing this resultant gives the cubic equation satisfied by $B_0(x)$.

It is not difficult to conjecture the expression of the coefficients of $B_0(x)$ from their first values. This suggests to introduce the auxiliary series $U = U(x)$ defined by $U = x(1+U)^3$. Then we check that $B_0(x) = 1 + U - U^2$ (both series satisfy the same equation). We complete the proof by applying the Lagrange inversion formula.

We apply the same method to Eq. (4). We find:

$$\left[2xy(y-1)C(x,y) + xy(1-y)C_0(x) + x(y-1) + y\right]^2 = \Delta(y)$$

with

$$\begin{aligned}\Delta = {} & x^2y^4C_0(x)^2 - 2xC_0(x)[1 - x + xC_0(x)]y^3 \\ & + [(1-x)^2 + 2x(1-2x)C_0(x) + x^2C_0(x)^2] + 2x[xC_0(x) - x - 1]y + x^2.\end{aligned}$$

Again, Δ has a double root at $y = Y(x)$ where $Y = Y(x)$ is the formal power series in x defined by $Y = 2xY(1-Y)C(x,Y) + xY(Y-1)C_0(x) + x(1-Y)$. Computing the resultant of Δ and $\partial\Delta/\partial y$ gives the algebraic equation satisfied by $C_0(x)$.

■

Remarks

1. The first part of the above proposition was already proved in [10, 13, 23].

2. Let c_n denote the coefficient of x^n in $C_0(x)$. The numbers c_n have large prime factors (see Table 1). We can prove they are not hypergeometric as follows: we first construct the linear recurrence with polynomial coefficients they satisfy (using, for instance, the MAPLE package GFUN [18]) and then look for all hypergeometric solutions

of this recurrence (using the algorithm HYPER [15]). We find that there is no such solution: the sequence $(c_n)_n$ is not hypergeometric.

This does not rule out the existence of an expression of the form

$$c_n = \sum_k F_{n,k}$$

where $F_{n,k}$ would be (doubly) hypergeometric. Such an expression could, for example, derive from an application of the Lagrange inversion formula. By manipulationg the equation that defines $C_0(x)$, we found that $\mathbb{Q}(x, C_0(x)) = \mathbb{Q}(x, V(x))$ where

$$(1-4x)V(x)^4 + xV(x)^2 - xV(x) + x^2 = 0.$$

This equation is quadratic in x and hence, not suitable for a direct application of the Lagrange inversion formula (which requires linear equations in x). We can actually prove that we cannot write $C_0(x)$ as a rational function of x and U, where U would be related to x via an algebraic equation $P(x,U) = 0$ of degree one in x. Hence the Lagrange inversion formula (in its simplest form) cannot be applied to obtain an expression of $C_0(x)$.

3. So far, we have found no q-analog of the quadratic method that would enable us to solve Eqs. (7) and (8).

5.3 Differential equations

We finally come to the functional-differential equation that defines the generating function for sorted permutations (6). It is very similar to the equation obtained for general permutations (5). The case of general permutations turns out to be extremely simple, as $D(x,y) = 1/(1-x-y)$. The case of sorted permutations is (and will remain) much more intriguing. However, we shall obtain a characterization of the series $E(x,0)$ that does not involve the series $E(x,y)$.

Notations. Let $f(x,y)$ be a formal power series in x with rational coefficients in y. We denote by f' the derivative $\partial f/\partial x$. We denote by Lf the formal Laplace transform of f with respect to x:

$$f(x,y) = \sum_{n\geq 0} a_n(y)\ \frac{x^n}{n!} \implies Lf(x,y) = \sum_{n\geq 0} a_n(y)\ x^n.$$

The Laplace transform has the following integral representation:

$$Lf(x,y) = \frac{1}{x}\int_0^\infty e^{-u/x} f(u,y)du.$$

Observe that

$$Lf(x,y) = f(0,y) + xL(f')(x,y). \tag{14}$$

Proposition 5.3 *Let*

$$\mathcal{E}(x) = \sum_{m\geq 0} e_{m,0}\ \frac{x^{m+1}}{(m+1)!}$$

where $e_{m,0}$ is the number of sorted permutations of length m. Note that $\mathcal{E}(x) = \int_0^x E(u,0)du$.
Let $f(x,y)$ be the following power series in x, with polynomial coefficients in y:

$$f(x,y) = \exp\left[(y-1)\mathcal{E}(x)\right].$$

Then the Laplace transform of f satisfies:

$$Lf\left(\frac{y}{1-y}, y\right) = 1-y. \tag{15}$$

Equivalently,

$$\int_0^\infty e^{-u(1-y)/y} \exp\left[(y-1)\mathcal{E}(u)\right] du = y.$$

This equation is equivalent to a recurrence relation defining the sequence $(e_{m,0})_m$, and hence, characterizes completely the series $\mathcal{E}(x)$.

Proof. This proposition is a special case of a more general approach that also allows us to derive the simple expression $D(x,y) = 1/(1-x-y)$ from Eq. (5).

Equations (5) and (6) have the following form:

$$\frac{\partial F}{\partial x}(x,y) = c(y)\,[1 + yF(x,y)]\ \frac{F(x,y) - F(x,0)}{y}, \tag{16}$$

where $c(y) = 1$ for general permutations and $c(y) = 1 - y$ for sorted permutations. Eq. (16), together with the initial condition $F(0,y) = 1/(1-y)$, defines $F(x,y)$ as a formal power series in x with rational coefficients in y. More precisely, $F(x,y)$ admits an expansion of the following form:

$$F(x,y) = \sum_{n\geq 0} \frac{P_n(y)}{(1-y)^{n+1}}\ \frac{x^n}{n!}$$

where $P_n(y) \in \mathbb{R}[y]$. We observe that Eq. (16) is a Riccati equation in $F(x,y)$. We linearize it by introducing the series

$$G(x,y) = \exp\left[-c(y)\int_0^x F(u,y)du\right], \tag{17}$$

so that $F = -G'/[c(y)G]$. We find

$$G(0,y) = 1, \quad G'(0,y) = \frac{c(y)}{y-1}, \tag{18}$$

and

$$y\ G'' + c(y)(yF(x,0) - 1)\ G' - c(y)^2 F(x,0)\ G = 0.$$

This equation can be rewritten as

$$[yG'' - c(y)G'] + c(y)F(x,0)\,[yG' - c(y)G] = 0,$$

which, using (18), gives

$$yG' - c(y)G = \frac{c(y)}{y-1}\ f(x,y) \tag{19}$$

with

$$f(x,y) = \exp\left[-c(y)\int_0^x F(u,0)du\right].$$

Taking the Laplace transform in (19) gives, thanks to (14):

$$[y - xc(y)]L(G')(x,y) - c(y) = \frac{c(y)}{y-1}\ Lf(x,y). \tag{20}$$

The definition (17) of G implies that it admits an expansion of the form

$$G(x,y) = \sum_{n\geq 0} \frac{Q_n(y)}{(1-y)^n}\ \frac{x^n}{n!}$$

where $Q_n(y) \in \mathbb{R}[y]$. Hence we can set $x = y/c(y)$ in (20) (this should remind the reader of the kernel method used in Section 5.1). We obtain

$$Lf\left(\frac{y}{c(y)}, y\right) = 1 - y. \tag{21}$$

Let us now apply this result to Eqs. (5) and (6).

• **General permutations.** When $c(y) = 1$, the series $F(x,y)$ is the exponential generating function $D(x,y)$ for general permutations. The series $f(x,y) = \exp\left[-\int_0^x D(u,0)du\right]$ only depends on x, and we shall denote it $f(x)$. Eq. (21) gives $Lf(x) = 1 - x$. Hence $f(x) = 1 - x$, and $D(x,0) = 1/(1-x)$. This is exactly (fortunately!) the exponential generating function for general permutations. Then, we integrate (19) and find $G(x,y) = (1-x-y)/(1-y)$, and finally,

$$D(x,y) = \frac{1}{1-x-y} = \sum_{m,n\geq 0} \frac{(m+n)!}{n!}\frac{x^m}{m!}y^n.$$

Hence, our – admittedly complicated – method is at least able to recover the expected result: the number of permutations σ of length $m+n$ such that $z(\sigma) \geq n$ is $(m+n)!/n!$.

• **Sorted permutations.** The success of our method on a problem we knew how to solve encourages us to apply the same method to the more tricky equation (6). When $c(y) = 1-y$, the series $F(x,y)$ is the exponential generating function $E(x,y)$ for sorted permutations. With the notations of Proposition 5.3, we have $f(x,y) = \exp[(y-1)\mathcal{E}(x)]$. Equation (21) gives (15).

To complete the proof of this proposition, we have to show that the functional equation we obtained completely characterizes $\mathcal{E}(x)$. Let us write $e_{i,0} = e_i$ for short. We have:

$$f(x,y) = \exp[(y-1)\mathcal{E}(x)] = \prod_{i\geq 1} \exp\left[(y-1)\frac{e_{i-1}}{i!}x^i\right].$$

This gives

$$Lf(x,y) = \sum_{r_1,r_2,r_3,\ldots\geq 0} x^{\sum ir_i}\frac{(\sum ir_i)!}{\prod r_i!}(y-1)^{\sum r_i}\prod_{i\geq 1}\left(\frac{e_{i-1}}{i!}\right)^{r_i}. \tag{22}$$

Let us observe that the identity (15) can be rewritten as

$$Lf\left(x, \frac{x}{1+x}\right) = \frac{1}{1+x}.$$

Thus, let us replace y by $x/(1+x)$ in (22) and expand the series we obtain. Taking the coefficient of x^n gives, for $n \geq 1$:

$$\sum_{\lambda}(-1)^{\ell(\lambda)+|\lambda|}\frac{|\lambda|!}{\prod r_i!}\binom{n-|\lambda|+\ell(\lambda)-1}{\ell(\lambda)-1}\prod_{i\geq 1}\left(\frac{e_{i-1}}{i!}\right)^{r_i} = 1,$$

where the sum is over all nonempty partitions λ of weight at most n, $\ell(\lambda)$ denotes the number of parts of λ, and r_i is the number of parts equal to i. This equation defines e_{n-1} in terms of $e_0, e_1, \ldots, e_{n-2}$, and hence the series $\mathcal{E}(x)$ is completely characterized by the functional equation we obtained.

∎

Final comments. Obviously, we have not completely solved the equations of Section 4. Two main questions arise:

- Eq. (6) defines a series $E(x,y)$. Proposition 5.3 gives a characterization of $E(x,0)$ that does not involve $E(x,y)$, but is of a very unusual form. Is there a more standard equation defining $E(x,0)$? for instance, an algebraic differential equation?

- Eqs. (7) and (8) cry for a q-analog of the quadratic method. Do $\bar{B}(x,0)$ and $\bar{C}(x,0)$ satisfy a q-algebraic equation, *i.e.*, a polynomial equation $P(x,q,\bar{F}(x),\bar{F}(xq),\ldots,\bar{F}(xq^k)) = 0$?

Acknowledgements. I would like to thank Cyril Banderier, Philippe Flajolet and Bruno Salvy for interesting discussions about the possible singularity structure of the series $E(x,0)$, as well as Bruno Gauthier for his assistance in the use of the package Hyperg.

References

[1] E. Barcucci, A. Del Lungo, S. Lanini, M. Macrì and R. Pinzani, The inversion number of some permutations with forbidden subsequences, *Proceedings of SOCA'96*, Tianjin (1996) 21–32.

[2] F. Bergeron and C. Reutenauer, Combinatorial resolution of systems of differential equations III: a special class of differentially algebraic series, *Europ. J. Combinatorics* **11** (1990) 501–512.

[3] M. Bousquet-Mélou, A method for the enumeration of various classes of column-convex polygons, *Discrete Math.* **154** (1996) 1–25.

[4] M. Bousquet-Mélou, Multi-statistic enumeration of two-stack sortable permutations, *Electronic J. Combin.* **5** (1998) R21.

[5] W. G. Brown, Enumeration of non-separable planar maps, *Canad. J. Math.* **15** (1963) 526–545.

[6] W. G. Brown, On the existence of square roots in certain rings of power series, *Math. Annalen* **158** (1965) 82-89.

[7] W. G. Brown and W. T. Tutte, On the enumeration of rooted non-seperable planar maps, *Canad. J. Math.* **16** (1964) 572–577.

[8] R. Cori, B. Jacquard and G. Schaeffer, Description trees for some families of planar maps, Proceedings of the 9th Conference "Formal Power Series and Algebraic Combinatorics", Vienna, 1997, pp. 196–208.

[9] R. Cori and J. Richard, Énumération des graphes planaires à l'aide des séries formelles en variables non commutatives, *Discrete Math.* **2** No.2 (1972) 115–162.

[10] S. Dulucq, S. Gire and O. Guibert, A combinatorial proof of J. West's conjecture, *Discrete Math.* **187** (1998) 71–96.

[11] S. Feretić and D. Svrtan, On the number of column-convex polyominoes with given perimeter and number of columns, in *Proceedings of the* 5^{th} *conference Formal Power Series and Algebraic Combinatorics*, A. Barlotti, M. Delest, and R. Pinzani eds., pages 201–214, Florence, june 1993.

[12] I. P. Goulden and D. M. Jackson, *Combinatorial enumeration,* John Wiley and Sons, New York, 1983.

[13] I. P. Goulden and J. West, Raney paths and a combinatorial relationship between rooted non-separable planar maps and two-stack-sortable permutations, *J. Combin. Theory Ser. A* **75** (1996) 220–242.

[14] D. E. Knuth, *The art of computer programming* Vol.1, *Fundamental algorithms*, Addison-Wesley, Reading, Massachusetts, 1973.

[15] M. Petkovšek, H. S. Wilf, and D. Zeilberger, $A = B$, A. K. Peters, Wellesley, Massachusetts, 1996.

[16] D. Rawlings, The ABC's of classical enumeration, *Ann. Sci. Math. Québec* **10** No.2 (1986) 207–235.

[17] D. Rawlings, The Euler-Catalan identity, *Europ. J. Combinatorics* **9** (1988) 53–60.

[18] B. Salvy and P. Zimmermann, GFUN: a MAPLE package for the manipulation of generating and holonomic functions in one variable, *ACM Transactions on Mathematical Software*, **20** (1994) 163–167.

[19] R. P. Stanley, Differentiably finite power series, *Europ. J. Combinatorics* **1** (1980) 175–188.

[20] W. T. Tutte, On the enumeration of planar maps, *Bull. Amer. Math. Soc.* **74** (1968) 64–74.

[21] J. West, Permutations with restricted subsequences and stack-sortable permutations, Ph.D. thesis, MIT, 1990.

[22] J. West, Sorting twice through a stack, *Theoret. Comput. Sci.* **117** (1993) 303–313.

[23] D. Zeilberger, A proof of Julian West's conjecture that the number of two-stack-sortable permutations of length n is $2(3n)!/((n+1)!(2n+1)!)$, *Discrete Math.* **102** (1992) 85–93.