K-sorted Permutations with Weakly Restricted Displacements

Ting Kuo

Department of Marketing Management, Takming University of Science and Technology
Taipei 11451, Taiwan, ROC
tkuo@takming.edu.tw

Received 4 February 2014; Revised 5 April 2014; Accepted 30 April 2014

Abstract. A permutation \( \pi = (\pi_1, \pi_2, \ldots, \pi_n) \) of \( \{1, 2, \ldots, n\} \) is called \( k \)-sorted if and only if \( |i - \pi_i| \leq k \), for all \( 1 \leq i \leq n \). We propose an algorithm for generating the set of all \( k \)-sorted permutations of \( \{1, 2, \ldots, n\} \) in lexicographic order. An inversion occurs between a pair of \( (\pi_i, \pi_j) \) if \( i < j \) but \( \pi_i \geq \pi_j \). Let \( \text{Kn} \) denote the maximum number of inversions in \( k \)-sorted permutations. For \( k \)-sorted permutations with weakly restricted displacements, i.e., \( \lfloor n/2 \rfloor \leq k \leq n-1 \), we propose a concise formula of \( \text{Kn} \) by using the generating functions approach.

Key words: \( k \)-sorted permutation, generating function, inversion, lexicographic order, ordinal representation

1 Introduction

A linear ordering of the elements of the set of \( n \) marks \( \{1, 2, 3, \ldots, n\} \) is called a permutation. Permutations are one of the most important combinatorial objects in computing. Many studies have been done on algorithms for generating permutations [8, 16]. However, little research has been devoted on a special kind of permutations, called \( k \)-sorted permutations. A \( k \)-sorted permutation is a class of restricted permutation that every mark is located at most \( k \) displacements from its right place. A restricted permutation is a permutation that the positions of the marks are subject to some restraints. The most widely known restricted permutations are called derangements that are permutations without fixed points, means no mark is located at its right place. In general, there are three directions of research in restricted permutations: enumerating, generating, and analyzing. Enumerating refers to derive a formula that can calculate the number of restricted permutations of a class of restricted permutations. Generating refers to design an algorithm that can generate all restricted permutations of a class of restricted permutations. Analyzing refers to analyze certain property of the restricted permutation of a class of restricted permutations. Many studies deal with the enumeration of restricted permutations \([12, 17, 18, 19]\). But, few research on the other two directions. Bongiovanni et al. [2] introduced a problem of quasi sorting as the one of transferring a given permutation to a \( k \)-sorted permutation. In that paper, they proposed the bases for the development of algorithms for generating \( k \)-sorted permutations through element comparisons but did not propose an algorithm for systematically generating all \( k \)-sorted permutations. Berman [1] and Dutton [4] focus on analyzing \( k \)-sorted permutations. The motivation of this paper is twofold. First, by using an ordinal representation \([9, 10]\), we propose an algorithm for systematically generating \( k \)-sorted permutations, in lexicographic order. That is, generate all \( k \)-sorted permutations step-by-step in increasing order. Second, as a measure of disordered, inversion of a permutation is a crucial property in the study of permutations \([7, 14]\). When a list of size \( n \) is nearly sorted, a straight insertion sort algorithm is highly efficient since only a number of comparisons equal to the number of inversions in the original list, plus at most \( n-1 \), is required \([4]\). So, we analyze the maximal number of inversions of \( k \)-sorted permutations. In short, this study focuses on generating and analyzing a class of restricted permutations, called \( k \)-sorted permutations.

Let \( S_n \) denote the set of all permutations of \( n \) marks \( \{1, 2, \ldots, n\} \). That is, a permutation \( \pi = (\pi_1, \pi_2, \ldots, \pi_n) \) belongs to \( S_n \) if and only if

\[
\pi_i \in \{1, 2, \ldots, n\}, \text{ for all } i = 1, 2, \ldots, n, \text{ and }
\pi_i \neq \pi_j, \text{ for all } i \neq j.
\]

For a permutation \( \pi \), if it satisfies \( \pi_i = i \) then we say that the mark \( i \) is located at its right place. A \( k \)-sorted permutation is a permutation that satisfies the following definition \([4]\).

**Definition 1.** A permutation \( \pi = (\pi_1, \pi_2, \ldots, \pi_n) \) in \( S_n \) is called \( k \)-sorted if and only if
In other words, every mark is located at most \( k \) displacements from its right place.

Let \( K_n \) denote the set of all \( k \)-sorted permutations of \( n \) marks \( \{1, 2, \ldots, n\} \). It is obvious that \( 0 \leq k \leq n - 1 \), and that \( K_n \subseteq S_n \). We propose a concise formula of the maximum number of inversions in \( k \)-sorted permutations with weakly restricted displacements. The name of “\( k \)-sorted permutations with weakly restricted displacements” was inspired by Lehmer [12]. In that paper, he proposed generating functions for \( k \)-sorted permutations that is called permutations with strongly restricted displacements, with \( k = 1, 2, \) and \( 3 \). In this paper, for \( \lfloor n/2 \rfloor \leq k \leq n - 1 \), we call them \( k \)-sorted permutations with weakly restricted displacements. Notice that \( n \) and \( k \) are two positive integers throughout this paper. For completeness, let’s include some definitions that are related to it.

**Definition 2.** In a permutation \( \pi = (\pi_1, \pi_2, \ldots, \pi_n) \), an inversion occurs between a pair of \(( \pi_i, \pi_j )\) if \( i < j \) but \( \pi_i > \pi_j \).

**Definition 3.** Let \( I(\pi_i) \) denote the number of inversions of \( \pi_i \), then \( I(\pi_i) \) is the number of \( j \)'s such that \( i < j \) but \( \pi_i > \pi_j \).

**Definition 4.** Let \( I(\pi) \) denote the number of inversions of a permutation \( \pi = (\pi_1, \pi_2, \ldots, \pi_n) \), then

\[
I(\pi) = \sum_{i=1}^{n} I(\pi_i).
\]

It is well known that the value of \( I(\pi) \) is a measure of disordered in a permutation \( \pi = (\pi_1, \pi_2, \ldots, \pi_n) \). In this study, we focus on the maximum number of inversions in \( K_n \).

**Definition 5.** Let \( I(n, k) \) denote the maximum number of inversions in \( K_n \), that is

\[
I(n, k) = \max \{ I(\pi) \}, \text{ for all } \pi \in K_n.
\]

It is trivial that when \( k = 0 \), for all \( n \), the number of permutations of \( K_n \) is one and \( I(n, k) \) is zero. Berman [1] gave a value of \( 2kn \) as an upper bound of \( I(n, k) \). Dutton [4] improved the upper bound down to a value of \( 0.6kn \) by proposing a formula as follows

\[
I(n, k) = 2kn - \min \{ f(t_1), f(t_2) \}, \text{ with}
\]

\[
t_1 = \left\lfloor \frac{n}{m} \right\rfloor, t_2 = \left\lfloor \frac{n}{m} \right\rfloor, m = \left\lfloor \frac{-1 + \sqrt{8k^2 + 8k + 1}}{2} \right\rfloor,
\]

and the function \( f(t) \) is defined as follows

\[
f(t) = \left( k(k + 1) - \frac{n}{t} \left\lfloor \frac{n}{t} \right\rfloor + 1 \right) + n \left\lfloor \frac{n}{t} \right\rfloor. \tag{1}
\]

Here, \( \lfloor x \rfloor \) (read “the floor of \( x \)” ) stands for the greatest integer that less than or equal to \( x \), and \( \lceil x \rceil \) (read “the ceiling of \( x \)” ) the least integer that greater than or equal to \( x \). Dutton’s solution is a more or less sophisticated approach.

The first contribution of this paper is that we propose an algorithm for generating the set of all \( k \)-sorted permutations of \( n \) marks \( \{1, 2, \ldots, n\} \) in lexicographic order. The second contribution of this paper is that, by using the generating functions approach, for \( k \)-sorted permutations of \( n \) marks \( \{1, 2, \ldots, n\} \) with weakly restricted displacements, we propose a concise formula of the maximum number of inversions in \( K_n \) as follows

\[
I(n, k) = 2nk - k(k + 1) - \frac{n(n - 1)}{2}. \tag{2}
\]

The rest of this paper is organized as follows. In Section 2, we discuss representation schemes of permutations. In Section 3, we propose an algorithm to generate, in lexicographic order, \( K_n \) and to compute \( I(n, k) \). In Section 4, we present several recurrences of \( I(n, k) \)'s that are used in the following section. In Section 5, we derive a concise formula of \( I(n, k) \) by using the generating function approach. Conclusions are summarized in Section 6.

## 2 Ordinal Representation Scheme

In combinatorics and mathematics, several representation schemes have been used for permutation, such as two-line form [6], cycle notation [6], permutation matrix [3], inversion vector [15], inversion table [7]. From a
different operational point of view, we proposed a new representation scheme of permutation that is called ordinal representation [9, 10]. Now, let’s take a brief look at it.

**Definition 6.** For a permutation \( \pi \) in the form of ordinal representation, that is \([D_1D_2\cdots D_n]\), it belongs to \( S_n \) if and only if

\[
1 \leq D_j \leq i, \text{ for all } i = 1, 2, \ldots, n.
\]

Here, \([D_1D_2\cdots D_n]\) is called the *ordinal digits* of a permutation \( \pi \).

The meaning of ordinal digits is easy to understand, if we imagine that a permutation is the result of a successive withdrawing of items individually, one after the other without replacement, from an ordered item set of \( n \) marks \{1, 2, \ldots, \( n \)\}. At the beginning of withdrawing, there are \( n \) choices we can choose to be the first component of \( \pi \). That is why we have \( 1 \leq D_n \leq n \). Once we have chosen an item as the first component of \( \pi \), there are \( n-1 \) choices left in the ordered item set. So, we have \( 1 \leq D_{n-1} \leq n-1 \). In the end, only one choice left, so we have \( 1 \leq D_1 \leq 1 \). In other words, the component \( \pi_{n-i+1} \) of \( \pi \) is determined by \( D_i \). Furthermore, the value of \( D_i \) is one plus the number of items that are less than \( \pi_{n-i+1} \) and to the right of it.

Since each permutation \( \pi \) in \( S_n \) corresponds uniquely to an integer \( q \) in the range of \([0, n!-1]\), we have the following theorem.

**Theorem 1.** In \( S_n \), there is a *one-to-one* correspondence between \([D_1D_2\cdots D_n]\) and \((\pi_1\pi_2\cdots\pi_n)\).

**Proof.** Clearly, it is easy to convert an integer \( q \) to its factorial representation [11]. First, we divide the integer \( q \) by \((n-1)!\) and set the quotient to \( C_{n-1} \), then the remainder is divided by \((n-2)!\) and the quotient is set to \( C_{n-2} \), and so on. That is, any integer \( q \) between \( 0 \) and \( n!-1 \) can be represented as

\[
q = C_{n-1} \times (n-1)! + C_{n-2} \times (n-2)! + \cdots + C_1 \times 1! + C_0 \times 0! .
\]

(3)

Here, the following constraints

\[
0 \leq C_j \leq j, \text{ for all } j = 0, 1, \ldots, n-1,
\]

are imposed to ensure uniqueness. These \( C_j \)'s are called the *factorial digits* of integer \( q \) [11]. By Definition 6 we know that

\[
1 \leq D_j \leq i, \text{ for all } i = 1, 2, \ldots, n.
\]

Hence, we have a *one-to-one* correspondence between \( D_i \) and \( C_j \) as follows:

\[
D_i = C_j + 1, \text{ where } i = j+1,
\]

for all \( i = 1, 2, \ldots, n \). ■

Thus, if we ordering all permutations in \( S_n \) in lexicographic order, for example when \( n = 7 \), then we can use the ordinal digits \([1 1 1 1 1 1 1]\) to represent the first (i.e., 0th) permutation \( \pi = (1 2 3 4 5 6 7) \) and \([7 6 5 4 3 2 1]\) to the last (i.e., 5039th) permutation \( \pi = (7 6 5 4 3 2 1) \), respectively. It is easy to see that \( \pi_1 = D_n \) for all permutations in \( S_n \).

## 3 Algorithm

Although many algorithms have been done for generating \( S_n \) and various permutation problems [8, 16], we know of no published algorithms for generating \( K_n \) in lexicographic order. By using ordinal representation, we have proposed a method for generating \( S_n \) in lexicographic order [9, 10]. In this study, we extend that method to generate \( K_n \) in lexicographic order and to compute \( I(n, k) \). The computation is based on an interesting property that the number of *inversions* of a permutation \( \pi \) is equal to the summation of \( \pi \)'s ordinal digits minus \( n \). This property is demonstrated in the following theorem.

**Lemma 1.** For a permutation \( \pi \) in the form of ordinal representation, \( \pi = [D_1D_2\cdots D_n] \), we have

\[
D_{n-i+1} = I(\pi_i) + 1, \text{ for all } i = 1, 2, \ldots, n.
\]

**Proof.** By Definition 3, we know that \( I(\pi_i) \) is the number of \( j \)'s such that \( i < j \) but \( \pi_i > \pi_j \). From the meaning of ordinal digits mentioned above, we know that the value of \( D_i \) is one plus the number of items that are less than \( \pi_{n-i+1} \) and to the right of it. In other words, we have

\[
D_i = I(\pi_{n-i+1}) + 1.
\]

That is, \( D_{n-i+1} = I(\pi_i) + 1 \). ■

35
Theorem 2. For a permutation $\pi$ in the form of ordinal representation, $\pi = [D_nD_{n-1}\cdots D_1]$, we have $I(\pi) = \sum^n_i D_i - n$.

Proof. By Definition 4 and Lemma 1, we have $I(\pi) = \sum^n_i I(\pi_i) = \sum^n D_i - n$. ■

Therefore, by using ordinal representation scheme, we can systematically generate the whole $K_n$ in lexicographic order, and by using Theorem 2, we can compute $I(\pi)$ directly and immediately. Note that, by Definitions 3 and 4, totally $n!$ comparisons are needed to compute $I(\pi)$. These two tasks can be described as the following algorithm.

Algorithm. Generate $K_n$ in lexicographic order and compute $I(n, k)$.
Input: $n$ and $k$.
Output: $K_n$ and $I(n, k)$.

Begin
$I(n, k) = 0$
For $D_n = 1$ To $n$
For $D_{n-1} = 1$ To $n - 1$
... 
For $D_1 = 1$ to 1
Let item set $A = \{1, 2, \ldots, n\}$
For $i = n$ To 1
Retrieve the $D_i$th item of $A$
If the $D_i$th item satisfies Definition 1 then
Let $\pi_{n-i+1} = \text{the } D_i \text{th item of } A$
Delete the $D_i$th item of $A$
Goto Next $i$
Else
Select case $i$
Case $n$
Goto Next $D_n$
Case $n - 1$
Goto Next $D_{n-1}$
... 
Case 1
Goto Next $D_1$
End Select
End if
Next $i$
Output $\pi = (\pi_1, \pi_2, \ldots, \pi_n)$ and Compute $I(\pi)$
If $I(\pi) > I(n, k)$ Then Let $I(n, k) = I(\pi)$
Next $D_1$
... 
Next $D_{n-1}$
Next $D_n$
Output $I(n, k)$
End

By using this algorithm, for $k = 2$ and $3 \leq n \leq 10$, we find the numbers of $K_n$ are 6, 14, 31, 73, 172, 400, 932, and 2177, respectively, and list some of them in Table 1. All of these numbers are same as those numbers described in [18]. In Table 1, we also list their corresponding $I(\pi)$ and $I(n, k)$.

By using this algorithm, for $2 \leq n \leq 39$ and $1 \leq k \leq n - 1$, we find their $I(n, k)$’s. Then according to these $I(n, k)$’s, in Section 4, we present several recurrences of $I(n, k)$’s that are further used for deriving a concise formula of $I(n, k)$ for $k$-sorted permutations with weakly restricted displacements.
### Table 1. $K_n$ for $k = 2$ and $3 \leq n \leq 5$.  

<table>
<thead>
<tr>
<th>$n$</th>
<th>$\pi = (\pi_1, \pi_2, \ldots, \pi_n)$</th>
<th>$\pi_1$</th>
<th>$\pi_2$</th>
<th>$\pi_3$</th>
<th>$\pi_4$</th>
<th>$I(\pi)$</th>
<th>$I(n, k)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>1</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>1</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>4</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>2</td>
<td>1</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>11</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>12</td>
<td>3</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>13</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>4</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>14</td>
<td>3</td>
<td>4</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>5</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>5</td>
<td>3</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>1</td>
<td>2</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>5</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>1</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>2</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>11</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>12</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>5</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>13</td>
<td>1</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>14</td>
<td>1</td>
<td>4</td>
<td>5</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>16</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>17</td>
<td>2</td>
<td>1</td>
<td>4</td>
<td>3</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>18</td>
<td>2</td>
<td>1</td>
<td>4</td>
<td>5</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>19</td>
<td>2</td>
<td>1</td>
<td>5</td>
<td>3</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>2</td>
<td>1</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>21</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>4</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>22</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>23</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>24</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>5</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>25</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>26</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>5</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>27</td>
<td>3</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>28</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>2</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>29</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>4</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>5</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>31</td>
<td>3</td>
<td>4</td>
<td>1</td>
<td>2</td>
<td>5</td>
<td>4</td>
</tr>
</tbody>
</table>
4 Recurrences of \( I(n, k) \)’s

In this section, we present some examples of \( I(n, k) \)’s in Table 2, and propose several recurrences that arise naturally from Table 2. By carefully observing the numbers in Table 2, it is not hard to come up with the following recurrences that are corresponding to the diagonals (with gray color) of Table 2. For convenience, with integer \( a \geq 1 \), we use “the \( a^{th} \) diagonal of Table 2” to stand for those numbers that are presented in the \( a^{th} \) diagonal of Table 2. For example, the 1\(^{st} \) diagonal of Table 2 stands for \{1, 3, 6, 10, 15, 21, 28, 36, 45, 55, 66, 78, 91, 105, 120, 136, 153, 171, 190\}, and the 2\(^{nd} \) diagonal of Table 2 stands for \{1, 4, 8, 13, 19, 26, 34, 43, 53, 64, 76, 89, 103, 118, 134, 151, 169, 188, 208\}.

In case of the 1\(^{st} \) diagonal of Table 2, i.e., \( k = n - 1 \), we have the following recurrence
\[
I(n, n-1) = I(n-1, n-2) + n - 1, \quad \text{for } n \geq 2,
\]
with \( I(1,0) = 0 \). In case of the 2\(^{nd} \) diagonal of Table 2, i.e., \( k = n - 2 \), we have the following recurrence
\[
I(n, n-2) = I(n-1, n-3) + n - 1, \quad \text{for } n \geq 4,
\]
with \( I(1,1) = 1 \). In case of the 3\(^{rd} \) diagonal of Table 2, i.e., \( k = n - 3 \), we have the following recurrence
\[
I(n, n-3) = I(n-1, n-4) + n - 1, \quad \text{for } n \geq 6,
\]
with \( I(1,2) = 4 \). Now, it is not difficult to generalize these recurrences as follows. In case of the \( a^{th} \) diagonal of Table 2, i.e., \( k = n - a \), we have the following recurrence
\[
I(n, n-a) = I(n-1, n-a-1) + n - 1, \quad \text{for } n \geq 2a,
\]
with
\[
I(2a-1, a-1) = (a-1)^2.
\]

Although we have found these recurrences, we are not satisfied yet. Our goal is to find a concise formula of \( I(n, k) \) that can give us an answer quickly. This goal is achieved in next section.

\[
\begin{array}{cccccccccccccc}
 n\& k & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 \& 12 & 13 & 14 & 15 & 16 & 17 & 18 & 19 \\
 2 & 1 & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \ & \&
right as possible. Since all permutations in $S_n$ are $(n-1)$-sorted, the maximum number of inversions occurs in the last permutation, i.e., $\pi = (n \ n-1 \cdots 2 \ 1)$. Thus,

$$I(n,n-1) = \frac{n(n-1)}{2}. \quad (10)$$

Definitely, the identity (10) is a special case of the identity (9). Now, we start by discussing the second diagonal of Table 2. First, let $A(x)$ denote the ogf of the sequence $(a_3, a_4, a_5, \ldots)$ in the second diagonal of Table 2, i.e., $k = n-2$, as follows

$$A(x) = a_3 x^0 + a_4 x^1 + a_5 x^2 + \cdots + a_k x^{k-3} + \cdots.$$  

Then, we can rewrite the corresponding recurrence (5) as

$$a_i = a_{i-1} + i - 1, \quad \text{for } i \ge 4, \text{ with } a_3 = 1. \quad (11)$$

To find $A(x)$, we multiply both sides of the recurrence (11) by $x^i$ and sum over $i \ge 4$, then we have

$$\sum_{i=4}^{\infty} a_i x^i = \sum_{i=4}^{\infty} a_{i-1} x^i + \sum_{i=4}^{\infty} (i-1) x^i.$$  

That is,

$$(A(x) - a_3) x^3 = A(x) x^4 + \frac{x^2}{(1-x)^2} - x^2 - 2 x^3.$$  

Since $a_3 = 1$, we obtain

$$A(x)(x^3 - x^4) = -\frac{x^2}{(1-x)^2} - x^2 - x^3.$$  

Thus, we have

$$A(x) = \frac{1 + x - x^2}{(1-x)^3}. \quad (12)$$

In order to obtain an explicit formula for the sequence $(a_3, a_4, a_5, \ldots)$, we have to expand $A(x)$ in a series of partial fraction. Fortunately, it is easy to expand $A(x)$ as follows.

$$A(x) = \frac{2}{(1-x)^3} - \frac{1}{(1-x)^2} - \frac{x^2}{(1-x)^2} + \frac{x^3}{1-x}$$

$$= 2 \sum_{i=0}^{\infty} \binom{i+2}{i} x^i - \sum_{i=0}^{\infty} \binom{i+1}{i} x^i - \sum_{i=0}^{\infty} \binom{i}{i} x^i$$

$$= \sum_{i=0}^{\infty} \left( \frac{i^2 + 5i + 2}{2} \right) x^i.$$  

(13)

Here, the coefficient of $x^0$ equals to 1, is the value of $a_3$, i.e., $I(3, 1)$; the coefficient of $x^1$ equals to 4, is the value of $a_4$, i.e., $I(4, 2)$; the coefficient of $x^2$ equals to 8, is the value of $a_5$, i.e., $I(5, 3)$; and so on.

Similarly, let $B(x)$ denote the ogf of the sequence $(b_3, b_4, b_5, \ldots)$ in the third diagonal of Table 2, i.e., $k = n-3$, as follows

$$B(x) = b_3 x^0 + b_4 x^1 + b_5 x^2 + \cdots + b_i x^{i-5} + \cdots.$$  

The corresponding recurrence (6) can be rewritten as

$$b_i = b_{i-1} + i - 1, \quad \text{for } i \ge 6, \text{ with } b_3 = 4.$$  

As before, we have

$$B(x) = \frac{4 - 3x}{(1-x)^3}. \quad (14)$$

By expanding $B(x)$ in a form of partial fraction, we obtain an explicit formula for the sequence $(b_3, b_4, b_5, \ldots)$ as

$$B(x) = \sum_{i=0}^{\infty} \left( \frac{i^2 + 9i + 8}{2} \right) x^i.$$  

(15)

Here, the coefficient of $x^0$ equals to 4, is the value of $b_3$, i.e., $I(5, 2)$; the coefficient of $x^1$ equals to 9, is the value of $b_4$, i.e., $I(6, 3)$; the coefficient of $x^2$ equals to 15, is the value of $b_5$, i.e., $I(7, 4)$, and so on.

Hence, in general, let $F(x)$ denote the ogf of the $a^{th}$ diagonal of Table 2, i.e., $k = n-a$, we have

$$F(x) = \sum_{i=0}^{\infty} \left( \frac{i^2 + (4a-3)i + 2(a-1)^2}{2} \right) x^i.$$  

Here, the coefficient of $x^0$ is the value of
\[ I(2a-1+i, a-1+i). \]

So, we have \( n = 2a - 1 + i, \) that is \( i = n - 2a + 1. \) Thus, by replacing \( a \) by \( n - k, \) we have \( i = 2k - n + 1. \) Finally, by replacing \( a \) by \( n - k, \) and \( i \) by \( 2k - n + 1, \) we have

\[ I(n,k) = 2nk - k(k + 1) - \frac{n(n-1)}{2}. \]

For example, for \( n = 9, \) what is the maximum number of inversions of 6-sorted permutations? First, since \( a = n-k = 3, \) we know that \( I(9,6) \) is in the third diagonal of Table 2. Second, since the first number in the third diagonal of Table 2 is the coefficient of \( x^0. \) So, by \( i = n - 2a + 1 = 4, \) or by \( i = k - a + 1 = 4, \) we know that \( I(9,6) \) is the fifth number in the third diagonal of Table 2, that is 30.

By (16), the coefficient of \( x^4 \) is the value of \( I(9,6), \) we have,

\[ I(9,6) = \frac{4^2 + (4 \times 3 - 3) \times 4 + 2(3-1)^2}{2} = 30. \]

Alternatively, by (9), we also have

\[ I(9,6) = 2 \times 9 \times 6 - 6(6+1) - \frac{9 \times (9-1)}{2} = 30. \]

### 6 Conclusions

The algorithm we proposed is easy to implement without any preprocessing and aiding by auxiliary data structures. It is quite suit for generating the set of all \( k \)-sorted permutations of \( n \) marks \( \{1, 2, \ldots, n\} \) in lexicographic order. We derive a concise formula of \( I(n, k) \) for permutations with weakly restricted displacements, i.e., \( \lfloor n/2 \rfloor \leq k \leq n - 1, \) by using the generating function approach. The generating function approach is a powerful and elegant way in analytic combinatorics [5]. The beauty of the generating function approach lies not in the result itself, but rather in its wide applicability. Our results can be extent to develop the formula of the distribution of inversions in \( K_n. \)

### Acknowledgement

The author is very grateful to anonymous referees for making a number of helpful suggestions.

### References


