

# Improved Linear-Time Ranking of Permutations

Harold R. Parks<sup>1</sup>, Dean C. Wills<sup>2,\*</sup>

<sup>1</sup>Department of Mathematics, Oregon State University, Corvallis, Oregon, USA.

<sup>2</sup>AppDynamics, San Francisco, California, USA.

**How to cite this paper:** Harold R. Parks, Dean C. Wills. (2021) Improved Linear-Time Ranking of Permutations. *Journal of Applied Mathematics and Computation*, 5(4), 277-282.

DOI: 10.26855/jamc.2021.12.006

**Received:** September 25, 2021

**Accepted:** October 22, 2021

**Published:** November 11, 2021

\*Corresponding author: Dean C. Wills, AppDynamics, San Francisco, California, USA.

Email: dean@lifetime.oregonstate.edu

## Abstract

A ranking function for the permutations on  $n$  symbols assigns a unique integer in the range  $[0, n! - 1]$  to each of the  $n!$  permutations. The corresponding unranking function is the inverse. We present simple  $O(n)$  ranking and unranking functions and permutation representations of a Foata transformation by Karttunen of the rankings introduced by Myrvold and Ruskey. Previous studies in the literature have either focused on lexicographic order, as the only reasonably intuitive order, or focused on the runtime performance of the algorithms. Our approach differs in that we provide an order that has algebraic significance while maintaining optimum performance. In addition, the methodology introduced herein, where mathematics and analysis are performed in the context of a descending transposition representation, is not only useful for analyzing and defining ranking, but also for the representation of all finite groups per Cayley's Theorem, which states that every group is isomorphic to a permutation group. Using this methodology, simple and efficient programs can be written to study and classify groups of different characteristics.

## Keywords

Ranking, Permutations, Group Theory, Algorithms and Data Structures

## 1. Introduction

Our goal in investigating ranking of permutations acting on  $\{1 \cdots n\}$  was to produce a ranking function that satisfied the following natural properties:

- (1) The identity permutation has rank 0. Moreover, permutations in  $S_n$ , the symmetric group, acting on  $\{1 \cdots n\}$  have ranks 0 through  $n! - 1$ .
- (2) If a permutation  $\pi \in S_n$  does not move any of the elements of  $\{m + 1 \cdots n\}$ , then the rank of that permutation should be the same as that of the permutation that agrees with  $\pi$  on  $\{1 \cdots m\}$ .
- (3) If two permutations in  $S_n$  move elements of disjoint sets, then the rank of their composition should equal the sum of their ranks.
- (4) Calculating the rank of a transposition should be clear and simple.

An ideal tool for creating such a ranking is the representation of permutations using a particularly structured sequence of transpositions as given in the following informal definition:

**Definition 1** (DTR). The *Descending Transposition Representation* of a permutation  $\pi \in S_n$ , acting on  $\{1 \cdots n\}$ , is the factorization of  $\pi$  into a product of transpositions

$$\pi = (n a_n) \cdots (2 a_2)(1 1), \text{ where } a_i \leq i. \quad (1.1)$$

For a given  $n$ , the  $DTR(\pi)$  is well-defined and unique. Using this, we can define a natural ranking satisfying the above properties.

**Definition 2** (K-rank). The K-rank of a permutation  $\pi \in S_n$  is

$$\text{rankK}(\pi) = \sum_{i=2}^n (i - a_n)(i - 1)!, \tag{1.2}$$

where  $a_n$  is defined as in the representation (1.1).

It turns out that the descending transposition representation of  $\pi \in S_n$  can be computed with an  $O(n)$  algorithm, so the result is a linear-time ranking.

After developing the DTR and the corresponding ranking, we discovered connections to the rankings given by Myrvold and Ruskey and that of Karttunen in the Online Encyclopedia of Integer Sequences [1].

In [2], Myrvold and Ruskey present two pairs of ranking and unranking functions for the permutations on  $n$  symbols, all with corresponding simple  $O(n)$  algorithms. The approach to ranking permutations used in [2], which allowed them to achieve linear time, is unusual in that it is not based on an intuitively understandable ordering of the permutations. In fact, their unranking algorithms have primacy, and the ranking functions and algorithms arise by inversion. Because they constructed their ranking functions as the inverses of their unranking functions, Myrvold and Ruskey had no reason to expect either ranking function to be intuitively understandable and indeed they commented that each ranking function “is not particularly easy to describe.” Unfortunately, the Myrvold and Ruskey rankings do not satisfy property (2) above. In this paper, using the DTR,<sup>1</sup> we will completely elucidate the rankings defined in [2].

## 2. Descending Transposition Representation

For  $n$  a natural number, the set of all permutations of  $\{1 \cdots n\}$  together with the operation of composition, denoted  $S_n$ , is a realization of the symmetric group on  $n$  letters. If  $n, m \in \mathbb{N}$  with  $n > m$ , then the obvious isomorphism mapping  $S_m$  into  $S_n$  maps the permutation  $\rho \in S_m$  to the permutation  $\sigma \in S_n$  with  $\sigma(i) = \rho(i)$  for  $1 \leq i \leq m$ , and  $\sigma(i) = i$  otherwise. To put all these groups and isomorphisms into one convenient framework, we introduce the telescoping symmetric groups.

**Definition 3.** We define the *telescoping infinite symmetric group* to be the set of permutations of the natural numbers that move but finitely many numbers, that is,

$$T_\infty = \{\text{bijections } \pi: \mathbb{N} \rightarrow \mathbb{N} | \exists n: \forall k > n, \pi(k) = k\} .$$

It is immediate to show that  $T_\infty$  forms a group under composition. Correspondingly, we define the *telescoping symmetric group* on  $n$  to be the subset of  $T_\infty$  that is the identity map on the complement of  $\{1 \cdots n\}$ , that is,

$$T_n = \{\pi \in T_\infty: \forall k > n, \pi(k) = k\}.$$

It is immediate to show that each  $T_n$  is a subgroup of  $T_\infty$ ,  $T_n \simeq S_n$  and  $T_\infty = \cup_n T_n$ . Consistent with Definition 1, we can define the DTR on  $T_\infty$ .

**Definition 4** (DTR on  $T_\infty$ ). The *Descending Transposition Representation* of a permutation  $\pi \in T_\infty$  is the factorization of  $\pi$  into a descending product of transpositions such that

$$\pi = \prod_{i=\infty}^1 \cdots (n \ a_n) \cdots (2 \ a_2) \cdots (1 \ 1), \text{ where } a_i \leq i. \tag{2.1}$$

Because  $\pi \in T_\infty \Rightarrow \exists n: \pi \in T_n$ , we see that all but finitely many of the transpositions in (2.1) are identity transpositions. Thus, it is convenient to abbreviate the DTR for  $\pi \in T_\infty$  in the form

$$\pi = (n \ a_n) \cdots (2 \ a_2) \cdots (1 \ 1) \text{ where } a_i \leq i, \text{ for some choice of } n.$$

To see that the DTR always exists, we reason as follows: If  $\pi \in T_n$ , then  $\pi(n) = a_n \leq n$ . Consequently, the permutation  $\pi_{n-1} = (n \ a_n)\pi$  is an element of  $T_{n-1}$ . Continuing this process, we find that a descending sequence of transpositions left factors a permutation recursively  $\forall k > 1$ . In effect, we are viewing each  $T_k$  as a union of cosets of  $T_{k-1}$ , and are left-factoring a permutation  $\pi_k \in T_k$  by a particular coset representative of  $T_{k-1}$  in  $T_k$ . It is easily proven that this representation and the DTR in its fully expanded form (2.1) is unique.

## 3. Ranking

Karttunen created a ranking using a Foata transformation [4] on the rankings produced by Myrvold and Ruskey in [2], and uploaded this algorithm to the OEIS [1] as PermUnrank3R, providing a Maple procedure; we call this rankK in this paper, crediting the discoverer. It turns out that rankK can be easily defined based on the permutation’s DTR representation, consistent with Definition 2 above.

<sup>1</sup> This work expands on theory introduced in [3].

**Definition 5** (Karttunen-rank). If  $\pi \in T_n$  has the DTR

$$\pi = (n a_n) \cdots (2 a_2) \cdots (1 1) \text{ where } a_i \leq i,$$

then the Karttunen ranking is defined to be

$$\text{rankK}(\pi) = \sum_{i=2}^n (i - a_n) (i - 1)!$$

Note that the Karttunen ranking is independent of  $n$ . We can also write rankK using the factorial number system  $d_n d_{n-1} \cdots d_2 0_1$ , where  $d_i = i - a_i$ , as was used by Knuth in describing an order for Algorithm P in [5]. The use of factorials in the context of unranking Combinations was thoroughly studied in [6].

**Proposition 6.** The Karttunen ranking has the following properties:

- (1) Linear time ranking. The Karttunen ranking of a permutation in  $T_n$  can be computed by Algorithm 6.1 in  $O(n)$  time.
- (2) Linear time unranking. For  $0 \leq i \leq n! - 1$ , the permutation in  $T_n$  having rank  $i$  can be computed by Algorithm 6.2 in  $O(n)$  time.
- (3) Consistency. If  $\pi \in T_m$  and  $n > m$ , then the Karttunen ranking of  $\pi$  as a permutation in  $T_n$  is equal to the Karttunen ranking of  $\pi$  as a permutation in  $T_m$ .

For some types of combinatorial objects, it is a breakthrough to obtain polynomial-time ranking and unranking of any order [7], so linear-time ranking and unranking is particularly noteworthy.

#### 4. Myrvold and Ruskey’s Tables Elucidated

In [2], Myrvold and Ruskey present two linear algorithms rank1 and rank2 and give tables of these rankings for  $n = 4$  on the set  $\{0,1,2,3\}$ . In this section, we will elucidate those rankings. We begin with rank2 because it turns out to be easier to explain. In Table 1, we present the data from Myrvold and Ruskey’s table for rank2. In our presentation of these tables, we have made the small change that we use permutations on the set  $\{1,2,3,4\}$  rather than the set  $\{0,1,2,3\}$ . This change is made to be consistent with subsequent papers (e.g., [8]) and to be consistent with the mathematical literature. We need not, and will not, present their algorithms, as our goal in this section is to elucidate their results using DTRs.

Each row of Table 1 is labelled by a non-negative integer, say  $r$ , in the first column; the second column gives the permutation that Myrvold and Ruskey’s algorithm rank2 assigns to rank  $r$ ; the third column gives a DTR for the permutation in the second column; the fourth column gives the Karttunen rank of the permutation in the second column. Writing permutations as DTRs and using the Karttunen rank for reference, we can make sense of the previously unintuitive rank2: When rank2 is applied to permutations on  $n$  letters, it simply reverses the order assigned by the Karttunen ranking.

If  $\pi \in T_n$  has the DTR  $\pi = (n a_n) \cdots (2 a_2) \cdots (1 1)$  where  $a_i \leq i$ , then

$$\text{rank2}(n, \pi) = (n! - 1) - \text{rankK}(\pi).$$

Note that it follows immediately that

$$\text{rank2}(n, \pi) = (n! - 1) - \sum_{i=2}^n (i - a_n) (i - 1)!.$$

**Table 1. Myrvold and Ruskey rank2 ordering**

rank2	$\pi$	DTR ( $\pi$ )	rankK	rank2	$\pi$	DTR ( $\pi$ )	rankK
0	2341	(41)(31)(21)	23	12	2413	(43)(31)(21)	11
1	3241	(41)(31)(22)	22	13	4213	(43)(31)(22)	10
2	3421	(41)(32)(21)	21	14	4123	(43)(32)(21)	9
3	4321	(41)(32)(22)	20	15	1423	(43)(32)(22)	8
4	2431	(41)(33)(21)	19	16	2143	(43)(33)(21)	7
5	4231	(41)(33)(22)	18	17	1243	(43)(33)(22)	6
6	4312	(42)(31)(21)	17	18	2314	(44)(31)(21)	5
7	3412	(42)(31)(22)	16	19	3214	(44)(31)(22)	4
8	3142	(42)(32)(21)	15	20	3124	(44)(32)(21)	3
9	1342	(42)(32)(22)	14	21	1324	(44)(32)(22)	2
10	4132	(42)(33)(21)	13	22	2134	(44)(33)(21)	1
11	1432	(42)(33)(22)	12	23	1234	(44)(33)(22)	0

In Table 2, we present the data from Myrvold and Ruskey’s table for rank1. The results of rank1 are not as simply related to the Karttunen rank as were the results of rank2. Nonetheless it is easy to see how to use the DTR to construct the ranking.

Put simply, rank1 encapsulates the data in the DTR as read from left to right (in contrast to rank2, which uses the DTR data from right to left). More precisely, we have the following: If we think of the transpositions of  $n$  as ordered  $(n\ 1), \dots, (n\ n)$ , then rank1 increments the leftmost transposition through all possibilities, then “carries” to the second transposition from the left. That process of “carrying” continues from left to right.

If  $\pi \in T_n$  has the DTR  $\pi = (n\ a_n) \cdots (2\ a_2) \cdots (1\ 1)$  where  $a_i \leq i$ , then

$$\text{rank1}(n, \pi) = \sum_{i=2}^n (a_i - 1) \frac{n!}{i!}.$$

**Table 2. Myrvold and Ruskey rank1 ordering**

rank1	$\pi$	DTR ( $\pi$ )	rankK	rank1	$\pi$	DTR ( $\pi$ )	rankK
0	2341	(41)(31)(21)	23	12	3241	(41)(31)(22)	22
1	4312	(42)(31)(21)	17	13	3412	(42)(31)(22)	12
2	2413	(43)(31)(21)	11	14	4213	(43)(31)(22)	10
3	2314	(44)(31)(21)	5	15	3214	(44)(31)(22)	4
4	3421	(41)(32)(21)	21	16	4321	(41)(32)(22)	20
5	3142	(42)(32)(21)	15	17	1342	(42)(32)(22)	14
6	4123	(43)(32)(21)	9	18	1423	(43)(32)(22)	8
7	3124	(44)(32)(21)	3	19	1324	(44)(32)(22)	2
8	2431	(41)(33)(21)	19	20	4231	(41)(33)(22)	18
9	4132	(42)(33)(21)	13	21	1432	(42)(33)(22)	12
10	2143	(43)(33)(21)	7	22	1243	(43)(33)(22)	6
11	2134	(44)(33)(21)	1	23	1234	(44)(33)(22)	0

### 5. Composition of Permutations and Realizations of Groups

The group operation in  $T_\infty$  is composition.

Let us consider two permutations on four letters, say  $\rho = \rho(1)\rho(2)\rho(3)\rho(4) = 2\ 3\ 4\ 1$ , and  $\sigma = 4\ 3\ 2\ 1$ .

To clarify our notation, we note that  $\rho \cdot \sigma(1) = \rho[\sigma(1)] = 1$ , while  $\sigma \cdot \rho(1) = 3$ .

We have  $\rho \cdot \sigma = 1\ 4\ 3\ 2$  and  $\sigma \cdot \rho = 3\ 2\ 1\ 4$ . Each of the preceding four permutations has a Karttunen rank:

$$\begin{aligned} \text{rankK}(\rho) &= 23, \text{rankK}(\sigma) = 20, \\ \text{rankK}(\rho \cdot \sigma) &= 12, \text{rankK}(\sigma \cdot \rho) = 4. \end{aligned}$$

Because rankK is a one-to-one mapping of  $T_\infty$  onto the non-negative integers, we can use composition of permutations and rankK to induce a multiplication  $\odot$  on the non-negative integers. For example, we see from the preceding paragraph that

$$23 \odot 20 = 12 \text{ and } 20 \odot 23 = 4.$$

Of course, the non-negative integers form a group under the multiplication  $\odot$  and rankK is an isomorphism.

A priori it might seem that, for a pair of non-negative integers  $a$  and  $b$ , computing the value of  $a \odot b$  is a laborious process. In fact, Algorithm 6.3 implements this multiplication, in  $O(n)$  time when  $a$  and  $b$  are less than  $n!$ . In that sense, multiplication using  $\odot$  has the same level of complexity as the ordinary multiplication taught in grade school.

Cayley’s Theorem tells us that every group  $G$  is isomorphic to a subgroup of the symmetric group on  $G$  (see e.g. [9]). By numbering the elements of  $G$ , we see that every finite group is isomorphic to a subgroup of  $T_\infty$  and thus is isomorphic to a subgroup of the non-negative integers with the multiplication. Because of the consistency property of rankK, if  $G$  is a subgroup of  $H$  and the numbering of the elements of  $H$  extends the numbering of the elements in  $G$ , then the isomorphic image of  $G$  will be a subgroup of the isomorphic image of  $H$ . Thus, we can study a group and its subgroups as they are realized by their images in the non-negative integers with the operation  $\odot$ . If we are considering a symmetry group, then the symmetries themselves can provide the permutations needed to realize the group as a subgroup of the

non-negative integers under the operation  $\odot$ .

As a simple example, consider the group of symmetries of the square, i.e., the Dihedral Group of order 8,  $D_8$ , that is generated by a quarter-turn rotation  $\rho$  and a reflection  $\sigma$ .

If the vertices of the square are numbered from 1 to 4 clockwise, then we set  $\rho = 2\ 3\ 4\ 1$  and  $\sigma = 4\ 3\ 2\ 1$ , as above. It then turns out that  $D_8$  is realized by

$$D = \{0, 4, 7, 9, 12, 16, 20, 23\}.$$

At first glance, it might seem that the set of numbers  $D$  tells us nothing at all. As an example, to the contrary, consider the element  $4 \in D$ . Since  $4 < 3!$ , we see that 4 must have its  $\odot$ -inverse in the set  $\{0 \cdots 5\}$ , which is the rankK image of  $T_3$ . But also 4 must have its  $\odot$  inverse in  $D$ , since  $D$  is a group under  $\odot$ . Since  $T_3 \cap D = \{0, 4\}$ , we conclude that  $4 \odot 4 = 0$ . Since  $S = \{0 \cdots 23\}$  is a group isomorphic to  $S_4$ , and since  $D \cong D_8$  and  $D \subseteq S$ , we can conclude that  $S_4$  contains a subgroup isomorphic to  $D_8$ .

## 6. Algorithms

### Algorithm 6.1 rankK

```

Data:  $\pi \in T_\infty$ 
Result:  $r \in \mathbb{Z}$ 
 $r \leftarrow 0$ 
for  $k \leftarrow \text{len}(\pi)$  to 2 do
   $i \leftarrow \pi[k]$ 
  while  $i > k$  do
     $i \leftarrow \pi[i]$ 
  end
   $\pi[i] \leftarrow i$ 
   $r \leftarrow (k - i) + kr$ 
end

```

### Algorithm 6.2 unrankK

```

Data:  $r \in \mathbb{Z}$ 
Result:  $\pi \in T_\infty$ 
 $\pi[1] \leftarrow 1$ 
 $i \leftarrow 1$ 
while  $r > 0$  do
   $i \leftarrow i + 1$ 
   $\pi[i] \leftarrow i$ 
   $t[i] \leftarrow i - (r \bmod i)$ 
   $r \leftarrow \lfloor r/i \rfloor$ 
end
for  $j \leftarrow i$  to 2 do
   $\text{swap}(\pi[i], \pi[t[i]])$ 
end

```

### Algorithm 6.3 Composition

```

Data:  $n, a, b \in \mathbb{Z}$ 
Result:  $a \odot b \in \mathbb{Z}$ 
factorial[0]  $\leftarrow 1$ 
for  $i \leftarrow 1$  to  $n - 1$  do
  factorial[i]  $\leftarrow i \cdot \text{factorial}[i - 1]$ 
end
for  $i \leftarrow 0$  to  $n - 1$  do
   $\pi[i] \leftarrow i$ 
end

```

```
for  $r \in a, b$  do
  for  $i \leftarrow n - 1$  to 1 do
     $q, r \leftarrow \text{divmod}(r, \text{factorial}[i])$ 
    swap( $\pi[i], \pi[i - r]$ )
  end
end
end
 $p \leftarrow 0$ 
for  $i \leftarrow n - 1$  to 1 do
   $j \leftarrow \pi[i]$ 
  while  $j > i$  do
     $j \leftarrow \pi[j]$ 
  end
   $\pi[i] \leftarrow j$ 
   $p \leftarrow i(p + i - j)$ 
end
```

## References

- [1] The OEIS Foundation Inc. (2001). The On-Line Encyclopedia of Integer Sequences. [Online]. Available: <https://oeis.org/A060117>.
- [2] W. Myrvold and F. Ruskey. (2001). "Ranking and unranking permutations in linear time." *Inf. Process. Lett.*, vol. 79, no. 6, pp. 281-284, Sep. 2001.
- [3] D. C. Wills. (2009). "Connections between combinatorics of permutations and algorithms and geometry". Ph.D. dissertation, Oregon State University, Corvallis, OR, USA, 2009, aAI3376756.
- [4] D. Foata and M. Schützenberger. (1970). *Théorie géométrique des polynômes Eulériens, ser.* Lecture notes in mathematics. Springer, 1970.
- [5] D. E. Knuth. (1997). *The Art of Computer Programming, Volume 2 (3rd Ed.): Seminumerical Algorithms*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1997.
- [6] A. Genitrini and M. Pepin. (2021). "Lexicographic unranking of combinations revisited." *Algorithms*, vol. 14, no. 3, p. 97, Mar 2021. [Online]. Available: <http://dx.doi.org/10>.
- [7] P. Hartman and J. Sawada. (2019). "Ranking and unranking fixed-density necklaces and Lyndon words," *Theoret. Comput. Sci.*, vol. 791, pp. 36-47, 2019.
- [8] M. Mareš and M. Straka. (2007). "Linear-time ranking of permutations," in *Algorithms – ESA 2007*, ser. Lecture Notes in Computer Science, W. E. Arge L., Hoffmann M., Ed. Springer, Berlin, Heidelberg, 2007, vol. 4698, pp. 187-193.
- [9] J. Meier. (2008). *Groups, Graphs and Trees: An Introduction to the Geometry of Infinite Groups*, ser. London Mathematical Society Student Texts. Cambridge University Press, 2008.