10. Permutation groups

Definition 10.1. *Let S be a set. A* ***permutation*** *of S is simply a bijection $f\colon S \longrightarrow S$.*

Lemma 10.2. *Let S be a set.*

(1) Let f and g be two permutations of S. Then the composition of f and g is a permutation of S.
(2) Let f be a permutation of S. Then the inverse of f is a permutation of S.

Proof. Well-known. □

Lemma 10.3. *Let S be a set. The set of all permutations, under the operation of composition of permutations, forms a group $A(S)$.*

Proof. (10.2) implies that the rule of multiplication is well-defined. We check the three axioms for a group.

We already proved that composition of functions is associative.

Let $i\colon S \longrightarrow S$ be the identity function from S to S. Then i is a permutation. Let f be a permutation of S. Clearly $f \circ i = i \circ f = f$. Thus i acts as an identity.

Let f be a permutation of S. Then the inverse g of f is a permutation of S and $f \circ g = g \circ f = i$, by definition. Thus inverses exist and G is a group. □

Lemma 10.4. *Let S be a finite set with n elements.*

Then $A(S)$ has $n!$ elements.

Proof. Well-known. □

Definition 10.5. *The group S_n is the set of permutations of the first n natural numbers.*

We want a convenient way to represent an element of S_n. The first way is to write an element σ of S_n as a matrix.

$$\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 3 & 1 & 5 & 4 & 2 \end{pmatrix} \in S_5.$$

Thus, for example, $\sigma(3) = 5$. With this notation it is easy to write down products and inverses. For example suppose that

$$\sigma = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 3 & 1 & 5 & 4 & 2 \end{pmatrix} \qquad \tau = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 4 & 3 & 1 & 2 & 5 \end{pmatrix}.$$

Then

$$\tau\sigma = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 1 & 4 & 5 & 2 & 3 \end{pmatrix}.$$

On the other hand

$$\sigma\tau = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 4 & 5 & 3 & 1 & 2 \end{pmatrix}.$$

In particular S_5 is not abelian.

The problem with this way of representing elements of S_n is that we don't see much of the structure of τ this way. For example, it is very hard to figure out the order of τ from this representation.

Definition 10.6. *Let τ be an element of S_n.*

*We say that τ is a **k-cycle** if there are integers $a_1, a_2, \dots, a_k$ such that $\tau(a_1) = a_2$, $\tau(a_2) = a_3$, and $\tau(a_k) = a_1$ and τ fixes every other integer.*

More compactly

$$\tau(j) = \begin{cases} a_{i+1} & \text{if } j = a_i \text{ and } i < k \\ a_1 & \text{if } j = a_k \\ j & \text{otherwise.} \end{cases}$$

For example

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 1 \end{pmatrix}$$

is a 4-cycle in S_4 and

$$\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 1 & 5 & 3 & 2 & 4 \end{pmatrix}.$$

is a 3-cycle in S_5.

Now given a k-cycle τ, there is an obvious way to represent it, which is much more compact than the first notation.

$$\tau = (a_1, a_2, a_3, \dots, a_k).$$

Thus the two examples above become,

$$(1, 2, 3, 4)$$

and

$$(2, 5, 4).$$

Note that there is some redundancy. For example, obviously

$$(2, 5, 4) = (5, 4, 2) = (4, 2, 5).$$

A two cycle is more often called a **transposition**. The tranposition

$$(a, b)$$

switches a and b and fixes everything else.

Note that a k-cycle has order k.

Definition-Lemma 10.7. *Let σ be any element of S_n.*

*Then σ may be expressed as a product of disjoint cycles. This factorisation is unique, ignoring 1-cycles, up to order. The **cycle type** of σ is the lengths of the corresponding cycles.*

Proof. We first prove the existence of such a decomposition. Let $a_1 = 1$ and define a_k recursively by the formula

$$a_{i+1} = \sigma(a_i).$$

Consider the set

$$\{\, a_i \,|\, i \in \mathbb{N} \,\}.$$

As there are only finitely many integers between 1 and n, we must have some repetitions, so that $a_i = a_j$, for some $i < j$. Pick the smallest i and j for which this happens. Suppose that $i \neq 1$. Then $\sigma(a_{i-1}) = a_i = \sigma(a_{j-1})$. As σ is injective, $a_{i-1} = a_{j-1}$. But this contradicts our choice of i and j. Let τ be the j-cycle $(a_1, a_2, \dots, a_j)$. Then $\rho = \sigma\tau^{-1}$ fixes each element of the set

$$\{\, a_i \,|\, i \leq j \,\}.$$

Thus by an obvious induction, we may assume that ρ is a product of $k-1$ disjoint cycles $\tau_1, \tau_2, \dots, \tau_{k-1}$ which fix this set.

But then

$$\sigma = \rho\tau = \tau_1\tau_2 \dots \tau_k,$$

where $\tau = \tau_k$.

Now we prove uniqueness. Suppose that $\sigma = \sigma_1\sigma_2 \dots \sigma_k$ and $\tau = \tau_1\tau_2 \dots \tau_l$ are two factorisations of σ into disjoint cycles. Suppose that $\sigma_1(i) = j$. Then for some p, $\tau_p(i) \neq i$. By disjointness, in fact $\tau_p(i) = j$. Now consider $\sigma_1(j)$. By the same reasoning, $\tau_p(j) = \sigma_1(j)$. Continuing in this way, we get $\sigma_1 = \tau_p$. But then just cancel these terms from both sides and continue by induction. □

Example 10.8. *Let*

$$\sigma = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 3 & 4 & 1 & 5 & 2 \end{pmatrix}.$$

Look at 1. 1 is sent to 3. But 3 is sent back to 1. Thus part of the cycle decomposition is given by the transposition $(1, 3)$. Now look at what is left $\{2, 4, 5\}$. Look at 2. Then 2 is sent to 4. Now 4 is sent to 5. Finally 5 is sent to 2. So another part of the cycle type is given by the 3-cycle $(2, 4, 5)$.

I claim then that

$$\sigma = (1, 3)(2, 4, 5) = (2, 4, 5)(1, 3).$$

This is easy to check. The cycle type is $(2,3)$.

As promised, it is easy to compute the order of a permutation, given its cycle type.

Lemma 10.9. *Let $\sigma \in S_n$ be a permutation, with cycle type $(k_1, k_2, \dots, k_l)$.*

The the order of σ is the least common multiple m of $k_1, k_2, \dots, k_l$.

Proof. Let k be the order of σ and let $\sigma = \tau_1\tau_2 \dots \tau_l$ be the decomposition of σ into disjoint cycles of length $k_1, k_2, \dots, k_l$.

Pick any integer h. As $\tau_1, \tau_2, \dots, \tau_l$ are disjoint, it follows that

$$\sigma^h = \tau_1^h \tau_2^h \dots \tau_l^h.$$

Moreover the RHS is equal to the identity, if and only if each individual term is equal to the identity.

It follows that

$$\tau_i^k = e.$$

In particular k_i divides k. Thus the least common multiple, m of $k_1, k_2, \dots, k_l$ divides k. But $\sigma^m = \tau_1^m \tau_2^m \tau_3^m \dots \tau_l^m = e$. Thus m divides k and so $k = m$. □

Note that (10.7) implies that the cycles generate S_n. It is a natural question to ask if there is a smaller subset which generates S_n. In fact the 2-cycles generate.

Lemma 10.10. *The transpositions generate S_n.*

Proof. It suffices to prove that every permutation is a product of transpositions.

We give two proofs of this fact.

Here is the first proof. As every permutation σ is a product of cycles, it suffices to check that every cycle is a product of transpositions.

Consider the k-cycle $\sigma = (a_1, a_2, \dots, a_k)$. I claim that this is equal to

$$\sigma = (a_1, a_k)(a_1, a_{k-1})(a_1, a_{k-2}) \dots (a_1, a_2).$$

It suffices to check that they have the same effect on every integer j between 1 and n. Now if j is not equal to any of the a_i, there is nothing to check as both sides fix j. Suppose that $j = a_i$. Then $\sigma(j) = a_{i+1}$. On the other hand the the transposition (a_1, a_i) sends j to a_1 and the next transposition then sends a_1 to a_{i+1}. No other of the remaining transpositions have any effect on a_{i+1}. Thus the RHS also sends $j = a_i$ to a_{i+1}. As both sides have the same effect on j, they are equal. This completes the first proof.

To see how the second proof goes, think of a permutation as just being a rearrangement of the n numbers (like a deck of cards). If we can find a product of transpositions, that sends this rearrangement back to the trivial one, then we have shown that the inverse of the corresponding permutation is a product of transpositions. Since a transposition is its own inverse, it follows that the original permutation is a product of transpositions (in fact the same product, but in the opposite order). In other words if

$$\tau_k \dots \tau_3 \cdot \tau_2 \cdot \tau_1 \cdot \sigma = e,$$

then multiplying on the right by τ_i, in the opposite order, we get

$$\sigma = \tau_1 \cdot \tau_2 \cdot \tau_3 \cdot \dots \tau_k.$$

The idea is to put back the cards into the correct position, one at a time. Suppose that the first $i-1$ cards are in the correct position. Suppose that the ith card is in position j. As the first $i-1$ cards are in the correct position, $j \geq i$. We may assume that $j > i$, otherwise there is nothing to do. Now look at the transposition (i, j). This puts the ith card into the correct position. Thus we are done by induction on i. $\square$