“NUMBER SYSTEMS” FROM FUNCTIONS

In class on September 21 we saw how the set of one-to-one functions from a set onto itself satisfy the first three axioms of our axioms for integers. Here the “binary operation” is composition of functions.

Suppose we start with the set \( \{1, 2, 3\} \). Here are the one-to-one and onto functions:

\[
\begin{align*}
    f(1) &= 2, & f(2) &= 1, & f(3) &= 3 \\
    g(1) &= 2, & g(2) &= 3, & g(3) &= 1 \\
    h(1) &= 3, & h(2) &= 2, & h(3) &= 1 \\
    j(1) &= 3, & j(2) &= 1, & j(3) &= 2 \\
    k(1) &= 1, & k(2) &= 3, & k(3) &= 2
\end{align*}
\]

and the identity function \( i \). The composition table is

\[
\begin{array}{cccccc}
\circ & i & f & g & h & j \\
\hline
i & i & f & g & h & j \\
f & f & i & k & j & h \\
g & g & h & j & k & i \\
h & h & g & f & i & k \\
j & j & k & i & f & g \\
k & k & j & h & g & f
\end{array}
\]

Are the “one-to-one and onto” here redundant?

Can you find a general formula for the number of one-to-one and onto functions from \( \{1, 2, \ldots, n\} \) to itself?

Definition/Notation: The set of one-to-one and onto functions from \(\{1, 2, \ldots, n\}\) to itself is called the symmetric group on \(n\) letters and is typically denoted \(S_n\).

\(S_n\) has \(n!\) elements, so is rather large: \(S_5\) has \(5! = 120\) elements, \(S_6\) has \(6! = 720\) elements, \(S_7\) has \(7! = 5040\) elements, ...

We can view an element of \(S_n\) as a “mixing up” of the numbers 1 up to \(n\). As you may have seen from elementary combinatorics, this “mixing up” is called a permutation. In other words, the one-to-one and onto functions from \(\{1, 2, \ldots, n\}\) to itself are called permutations.

If we are going to work with permutations (that is, compose them with one another), we need a reasonable notation. Let’s start with the six permutations in \(S_3\) we already looked at.

Notation for the Permutations in \(S_3\)

The identity function \(i\) sends 1 to 1 (and 2 to 2 and 3 to 3); we write

\[i = (1) \quad \text{or} \quad \begin{pmatrix} 1 & 2 & 3 \\ 1 & 2 & 3 \end{pmatrix} .\]

The function \(f\) sends 1 to 2, and then sends 2 back to 1 (3 is left alone); we write

\[f = (1, 2) \quad \text{or} \quad \begin{pmatrix} 1 & 2 & 3 \\ 2 & 1 & 3 \end{pmatrix} .\]

The function \(g\) sends 1 to 2, sends 2 to 3, and sends 3 back to 1; we write

\[g = (1, 2, 3) \quad \text{or} \quad \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \end{pmatrix} .\]

The function \(h\) sends 1 to 3 and then sends 3 back to 1 (2 is left alone); we write

\[h = (1, 3) \quad \text{or} \quad \begin{pmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \end{pmatrix} .\]

The function \(j\) sends 1 to 3, sends 3 to 2, and sends 2 back to 1; we write

\[j = (1, 3, 2) \quad \text{or} \quad \begin{pmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \end{pmatrix} .\]

The function \(k\) sends 2 to 3 and sends 3 back to 2 (1 is left alone); we write

\[k = (2, 3) \quad \text{or} \quad \begin{pmatrix} 1 & 2 & 3 \\ 1 & 3 & 2 \end{pmatrix} .\]
In the first place, we wrote each permutation as a *cycle*. The fact that each permutation was a cycle here is not typical; with larger values of \( n \), a typical permutation won’t be a cycle. However, any permutation can be written as a composition (or “product”) of cycles.

**Examples:**

1. In \( S_4 \), the permutation

\[
\begin{pmatrix}
1 & 2 & 3 & 4 \\
2 & 1 & 4 & 3
\end{pmatrix}
\]

is the composition of the cycles \((1, 2)\) and \((3, 4)\), which we write as \((1, 2)(3, 4)\).

2. In \( S_5 \), the permutation

\[
\begin{pmatrix}
1 & 2 & 3 & 4 & 5 \\
2 & 3 & 1 & 5 & 4
\end{pmatrix}
\]

is the composition of the cycles \((1, 2, 3)\) and \((4, 5)\), which we write as \((1, 2, 3)(4, 5)\).

Cycles themselves can be written as a composition of cycles involving just two numbers (these are called *transpositions*).

**Examples:**

1. The cycle \((1, 3, 2)\) can be written as the composition of transpositions \((2, 3)(1, 2)\).

2. The cycle \((1, 2, 3, 4)\) can be written as the composition of transpositions \((3, 2)(4, 3)(1, 4)\).