20200116

Remember that S_n represents the set of all permutations of {1, 2, ... n}

There are some basic facts about S_n that we need to have in hand:

1. Closure: If $\pi \in S_n$ and $\sigma \in S_n$ then $\pi \circ \sigma \in S_n$

2. Associativity: If $\pi \in S_n$ and $\sigma \in S_n$ and $\tau \in S_n$ then $\pi \circ (\sigma \circ \tau) = (\pi \circ \sigma) \circ \tau$

3. Identity Element: There is a permutation ι such that $\pi \circ \iota = \iota \circ \pi = \pi \;\; \forall \;\; \pi \in S_n$

4. Inverse: If $\pi \in S_n$ then $\pi^{-1} \in S_n$ and $\pi \circ \pi^{-1} = \pi^{-1} \circ \pi = \iota$

Property 1 says that the composition of two permutations is another permutation. This sounds trivial but it is our first look at a very important concept: **closure**. When we apply an operation to two elements of a set and **always** get another element of the same set, we say that set is **closed** under that operation.

Not all sets are closed under all operations. For example, $\mathbb{N}$ is not closed under the operation of subtraction (for instance, $3 \in \mathbb{N}$ and $4 \in \mathbb{N}$, but $3 - 4 = -1$, which is not in $\mathbb{N}$). However $\mathbb{N}$ **is** closed under addition and multiplication. This concept is vital to us as computer scientists because we frequently work with strongly typed programming languages, where each variable has a specific type that cannot change. If we are dealing with integer variables, we need to be sure the operations we perform will only produce integer values.

Property 2 is called the associative property. It says that if we are composing a sequence of permutations we can group them with parentheses in different ways without changing the result. We will see an application of this later in these notes.

Property 3 asserts that the identity permutation ι can be composed with any permutation without changing it. Once again, we can draw a parallel to other sets and operations. For example, in the set $\mathbb{N}$ and the operation of multiplication, we know that

$$x * 1 = 1 * x = x \quad \forall \;\; x$$

Property 4 asserts that every permutation has an *inverse***.** This property is also true for some sets and operations but not all. For example, in the set $\mathbb{Z}$ and the operation of addition, the identity element is 0 and every element has an inverse (for instance, the inverse of 7 is -7). However in the set $\mathbb{N}$ and the operation of addition, the identity element is still 0 but the non-zero values do not have inverses (for instance there is no integer $x \in \mathbb{N}$ such that $8 + x = 0$).

Property 4 is particularly important when we use permutations in cryptography – there's not much point encoding information with a permutation if there is not some other permutation that will do the decoding.

Each of these properties follows from the definition of permutations and the properties of functions. I recommend that you do some examples and convince yourself that these are true.

(Side note: A set and an operation that satisfy these four properties are called a **group**. Group theory is one of the most important branches of mathematics, with applications in communications, theoretical physics, applied physics, biology, chemistry, robotics and many other fields. We don't have time in CISC-203 to explore this topic but I encourage you to spend some time looking into it.)

Note that there is property possessed by many operations that is **not** true of the composition of permutations: **commutativity**. Commutativity holds when we can switch the left-to-right order of the operands without changing the result. For example, when we are multiplying integers, we know that $x * y = y * x$ ... and the same is true for addition. Not all operations are commutative. For example, subtraction is not commutative: $x - y \neq y - x$ except when $x = y$

Composition of permutations **is not commutative**. In general, $\pi \circ \sigma \neq \sigma \circ \pi$ ... although we will see some special cases where they are equal.

In class I posed a challenge: given permutations π and σ in S_n , can you always find a permutation α such that $\pi \circ \alpha = \sigma$?

The answer is yes. Here's how: we can solve this equation for α in much the same way as we would solve an equation involving numbers ... we just try to get α by itself on one side. (What I mean is, if we are asked to solve $8 + x = 23$ we isolate the x by adding -8 to both sides getting $-8 + 8 + x = -8 + 23$, which simplifies to $0 + x = 15$, and finally $x = 15$. But here the only operation we have is composition. So what can we do with composition to get rid of the π on the left side? Well, remember that $\pi^{-1} \circ \pi = \iota$ ... and $\iota \circ \alpha = \alpha$

So we can start with $\pi \circ \alpha = \sigma$ (in which π and σ are known, and α is the unknown) and apply the following operations that maintain equality

$$\pi \circ \alpha = \sigma$$

$$\pi^{-1} \circ (\pi \circ \alpha) = \pi^{-1} \circ \sigma$$

$$(\pi^{-1} \circ \pi) \circ \alpha = \pi^{-1} \circ \sigma$$

$$\iota \circ \alpha = \pi^{-1} \circ \sigma$$

$$\alpha = \pi^{-1} \circ \sigma$$

and we are done! It's exactly the same process as solving for x in $8 + x = 23$

It's easy to check that this is correct. If we take $\pi \circ \alpha$ and replace α by $\pi^{-1} \circ \sigma$ we get $\pi \circ (\pi^{-1} \circ \sigma)$ which equals $(\pi \circ \pi^{-1}) \circ \sigma$ which equals $\iota \circ \sigma$ which equals σ

But wait a second! This means that in order to find α we need to know π^{-1} ... is there some way we can compute the inverse of π ? The answer is yes ... we will see there are at least two simple ways to compute π^{-1} if we are given π.

Cycle Notation for Permutations

Now we introduce ***another*** representation for permutations ... one that makes it possible to work with permutations very easily.

Consider this permutation:

$$\pi = \begin{bmatrix} 4 & 1 & 5 & 2 & 7 & 3 & 6 \end{bmatrix}$$

What happens if we imagine composing π with itself? Let's trace what happens to the element 1. We are going to apply π twice: the first application maps 1 to 4, and the second application maps that 4 to 2. If we compose with π again, that 2 is mapped back to 1. Treating π as a function, we see $\pi(1) = 4$, $(\pi \circ \pi)(1) = 2$, and $(\pi \circ \pi \circ \pi)(1) = 1$

(Remember, we write $\pi \circ \pi$ as π^2 and $\pi \circ \pi \circ \pi$ as π^3 etc.)

Composing with π even more times will cycle through 4 then 2 then 1 then 4 then 2 then 1 etc. We can write this behaviour as $1 \to 4 \to 2 \to 1 \to 4 \to 2 \to 1$ etc.

If we trace what happens to 2 when we repeatedly compose π with itself and apply the resulting function to 2, we see exactly the same pattern : $2 \to 1 \to 4 \to 2 \to 1 \to 4 \to 2$ etc. The same thing happens if we start with 4 and trace what happens to it when we repeatedly compose π with itself and apply the resulting function to 4: we get the pattern $4 \to 2 \to 1 \to 4 \to 2 \to 1 \to 4$ etc

So in this sense, 1, 4 and 2 form a cycle: 1 goes to 4, 4 goes to 2, and 2 goes to 1. We write this cycle as (1, 4, 2) - it is a **notational** device that describes the three ordered pairs (1,4),(4,2),(2,1) which belong to π

What about the rest of π ? Since we have dealt with 1, 2 and 4, let's see what happens to 3. Following the same analysis as we did for 1, 2 and 4 (but skipping over some of the details) we see this pattern: $3 \to 5 \to 7 \to 6 \to 3 \to 5 \to 7 \to 6 \to 3$ etc. which we write as (3, 5, 7, 6). Again we can see that this is a **non-ambiguous shorthand way** to represent the four ordered pairs (3,5), (5,7), (7,6), (6,3) that make up the rest of π

Thus we can express the entire definition of π with the two cycles (1,4,2)(3,5,7,6) - we call this the **cycle notation** for π All of the information that defines π is there, expressed in a different way (in other words, we can reconstruct the standard representation from the cycle notation).

Notice that from each permutation, we can only get one cycle notation version. (We saw this for π above: no matter which elements we start with, we always get the same repeating patterns for 1, 4 and 2, and for 3, 5, 7, and 6.) Similarly, from any cycle notation representation, we can only reconstruct one permutation in standard notation. This means that **the cycle notation for each permutation is unique** (up to changing the order of the cycles, because (1,4,2)(3,5,7,6) gives the same information as (3,5,7,6)(1,4,2) and up to rotating the elements within each cycle, since (1,4,2) and (4,2,1) and (2,1,4) all represent the same information).

We can also extract the cycle notation for a permutation π without going through the effort of composing π with itself over and over. We can just build the cycles directly from π by observing "1 goes to 4, 4 goes to 2, and 2 goes to 1" to get the cycle (1, 4, 2). Then we can say "What happens to 3?" and observe "3 goes to 5, 5 goes to 7, 7 goes to 6, and 6 goes to 3" to get the cycle (3, 5, 7, 6).