Introduction to Group Theory

MATH2201/MATH2203

Sandip Chatterjee, Department of Mathematics

Reference

1. A First Course in Abstract Algebra, J.B. Fraleigh
2. Abstract Algebra, I. N. Herstein
3. Abstract Algebra, T. W. Judson
4. Higher Algebra, S.K.Mappa

It's nice to solve a given problem, but it is more important to make an attempt to solve it.

I.N. Herstein

University of Chicago, 1986

With the development of computing in the last several decades, applications that involve abstract algebra and discrete mathematics have become increasingly important in different fields of science and engineering, specifically in computer science. Though theory still occupies a central role in the subject of abstract algebra and no student should go through such a course without a good notion of what a proof is, the importance of applications such as coding theory and cryptography has grown significantly.

The Most Fundamental Phrases

If we can prove a statement true, then that statement is called a **proposition**. A proposition of major importance is called a **theorem**. Sometimes instead of proving a theorem or proposition all at once, we break the proof down into modules; that is, we prove several supporting propositions, which are called **lemmas**, and use the results of these propositions to prove the main result. If we can prove a proposition or a theorem, we will often, with very little effort, be able to derive other related propositions called **corollaries**.

Some Cautions and Suggestions

There are several different strategies for proving propositions. In addition to using different methods of proof, students often make some common mistakes when they are first learning how to prove theorems. To aid students who are studying abstract mathematics for the first time, we list here some of the difficulties that they may encounter and some of the strategies of proof available to them. It is a good idea to keep referring back to this list as a reminder. (Other techniques of proof will become apparent throughout this chapter and the remainder of the text.)

- A theorem cannot be proved by example; however, the standard way to show that a statement is not a theorem is to provide a counterexample

- Quantifiers are important. Words and phrases such as *only*, *for all*, *for every*, and *for some* possess different meanings.

- Never assume any hypothesis that is not explicitly stated in the theorem. You cannot take things for granted.

- Suppose you wish to show that an object exists and is unique. First show that there actually is such an object. To show that it is unique, assume that there are two such objects, say r and s, and then show that r = s.

- Sometimes it is easier to prove the contra-positive of a statement. Proving the statement “If p, then q" is exactly the same as proving the statement “If not q, then not p.”

- Although, it is usually better to find a direct proof of a theorem, this task can sometimes be difficult. It may be easier to assume that the theorem that you are trying to prove is false, and to hope that in the course of your argument you are forced to make some statement that cannot possibly be true.

Remember that one of the main objectives of higher mathematics is proving theorems. Theorems are tools that make new and productive applications of mathematics possible. We use examples to give insight into existing theorems and to foster intuitions as to what new theorems might be true. Applications, examples, and proofs are tightly interconnected-much more so than they may seem at first appearance.

1. Cartesian Products

Given sets A and B, we can define a new set A×B, called the *Cartesian product* of A and B, as a set of ordered pairs. That is,

$$A \times B = \{(a, b) : a \in A \text{ and } b \in B\}$$

Example 1.1. If A = {x, y}, B = {1, 2, 3}, and C = φ, then A × B is the set {(x,1),(x, 2),(x,3), (y,1); (y; 2); (y; 3)} and A × C = φ

We define the Cartesian product of n sets to be

$$A_1 \times ......\times A_n = \{(a_1,....,a_n) : a_i \in A_i \text{ for } i = 1,2,.....,n\}$$

If $A = A_1 = A_2 = ...= A_n$, we often write A^n for A×....×A (where A would be written n times). For example, the set R^3 consists of all of 3-tuples of real numbers.

2. Binary Operations

Definition 2.1. A ***binary operation*** ∗ on a non-empty set S is a rule that assigns to each ordered pair of elements of elements of S a uniquely determined element of S. The element assigned to the ordered pair (a, b) with a, b ∈ S is denoted by a ∗ b.

Remark. In other words, a binary operation of a set S is a function ∗ : S×S → S from the Cartesian product S × S to the set S. The only difference is that the value of the function ∗ at an ordered pair (a, b) is denoted by a ∗ b rather than ∗((a, b)).

Let S = N = {1, 2, 3, . . .}

Example 2.1. a ⋆ b = max(a, b), e.g. 2 ⋆ 3 = 3, 3 ⋆ 2 = 3, 3 ⋆ 3 = 3.

Example 2.2. a ⋄ b = a, e.g. 2 ⋄ 3 = 2, 3 ⋄ 2 = 3, 3 ⋄ 3 = 3.

Example 2.3. a¤b = a^b, e.g. 2¤3 = 2^3= 8, 3¤2 = 3^2 = 9, 3¤3 = 3^3 = 27.

Definition 2.2. A binary operation $\ast$ on a set S is ***commutative***, if

$$a \ast b = b \ast a \qquad \forall a, b \in S.$$

The binary operation $\star$ is commutative, but the binary operations $\diamond$ and ¤ are not commutative.

Let $\ast$ be a binary operation on a set S. and let a, b, c ∈ S. Consider the expression $a \ast b \ast c$. This expression doesn't have a meaning since $\ast$ gives only a meaning to ordered pairs of elements. In fact, there are two ways of making $a{\ast}b{\ast}c$ meaningful, namely $(a \ast b) \ast c$ and $a \ast (b \ast c)$.

For the operation $\star$ we have,

$(3 \star 2) \star 4 = 3 \star 4 = 4$

$3 \star (2 \star 4) = 3 \star 4 = 4.$

In fact, for all a, b, c ∈ N we have $(a \star b) \star c = a \star (b \star c) = \max(a, b, c)$.

For the operation $\diamond$ we have,

$$(3 \diamond 2) \diamond 4 = 3 \diamond 4 = 3$$

$$3 \diamond (2 \diamond 4) = 3 \diamond 2 = 3.$$

In fact, for all a, b, c ∈ N we have $(a \diamond b) \diamond c = a \diamond (b \diamond c) = a$.

But, things are different for the operation ¤. Here we have,

$$(3 \text{¤} 3) \text{¤} 3 = (3^3\ ) \text{¤} 3 = (3^3)^3 = 3^9$$

$$3 \text{¤} (3 \text{¤} 3) = 3 \text{¤} (3^3\ ) = 3^{27}.$$

So, in general, $(a \text{¤} b) \text{¤} c \neq a \text{¤} (b \text{¤} c)$.

Definition 2.3. A binary operation $\ast$ on a set S is called associative, if

$$(a \ast b) \ast c = a \ast (b \ast c).\ \forall a, b, c \in S.$$

In our examples, $\star$ is both commutative and associative, $\diamond$ is not commutative, but associative, ¤ is neither commutative nor associative.

If $*$ is an associative operation on S, then we can write $a * b * c$ for the common value of $(a * b) * c$ and $a * (b * c)$:

$$a * b * c = (a * b) * c = a * (b * c).$$

NB. This works only for associative operations!

Our three examples ⋆, ⋄, ¤ are of course artificially made up operations. But there are many natural examples of binary operations.

(a) $* = +$. Addition of numbers is a binary operation on N, Z, Q, R, C.

(b) $* = \times$. Multiplication of numbers is a binary operation on N, Z, Q, R, C and also on $R^+ = \{r \in R : r > 0\}$ and on $\{1, -1\}$.

(c) Addition and multiplication modulo n are binary operations on the set $Z_n = \{0, 1, . . . , n - 1\}$ of residues modulo n.

(d) Matrix addition and matrix multiplication are binary operations on the set $M_n(R)$ of all n × n matrices with entries in R, also on $M_n(Q)$, $M_n(C)$, $M_n(Z_n)$.

(e) $*$ = $\circ$ = composition of functions. This is a binary operation on the set $F(\Omega) = \{f \mid f : \Omega \to \Omega\}$ of all functions from Ω to itself. Recall: If $f : \Omega \to \Omega$ and $g : \Omega \to \Omega$ are functions from Ω to Ω, then $f \circ g \in F(\Omega)$ is the function defined by $(f \circ g)(x) = f(g(x))\ \forall x \in \Omega$.

ALL the binary operations in Examples (a) – (e) are associative, and all EXCEPT matrix multiplication and composition of functions are commutative.

Important points about binary operations:

(i) The result of the operation must be an element of S. This fails, for example, for + on the set $S = \{-1, 0, 1\}$ (as $1 + 1 = 2 \notin S$). (**Closure Property**)

(ii) The operation must be defined for ***all*** elements of S. This fails, for example for $A * B = A^{-1}BA$ on $M_n(R)$ (as the matrix A^{-1} may not exist).

(iii) The result of the operation must be ***uniquely determined***. This fails, for example, if we set $a * b = c$ where $c^2 = ab$ on C (as for $a = b = 2$, c may be 2 or −2).

Composition Tables

Let S = {a_1, a_2, . . . , a_n} be a finite set, and let ∗ be a binary operation on S. The multiplication table of ∗ is the table

∗	**a_1**	**a_2**	**…**	**a_j**	**…**	**a_n**
a_1	a_1*a_1	a_1*a_2	…	a_1*a_j	…	a_1*a_n
…	…	…	…	…	…	…
a_i	a_i*a_1	a_i*a_2	…	a_i*a_j	…	a_i*a_n
…	…	…	…	…	…	…
a_n	a_n*a_1	a_n*a_2	…	a_n*a_j	…	a_n*a_n

Example 2.4. For multiplication modulo 3, the multiplication table is

×	0	1	2
0	0	0	0
1	0	1	2
2	0	2	1

Remark. Commutativity of a binary operation is instantly recognizable from the multiplication table: $*$ is commutative if and only if its multiplication table is symmetric with respect to the main diagonal. There is no easy way of detecting associativity from the multiplication table.

Definition 2.4. Let $*$ be a binary operation on a set S. An element $e \in S$ is an ***identity element*** for $*$ if

$$e * a = a * e = a \qquad \forall a \in S.$$

Example 2.5. Recall our examples ⋆, ⋄ and ¤.

- $a \star b = \max(a, b)$. $e = 1$ is an identity element: $1 \star a = a \star 1 = a$ $\forall a \in N$.
- $a \diamond b = a$. There is no identity element. Indeed, suppose $e \in N$ is an identity element. Then we must have $e \diamond 1 = 1$. But $e \diamond 1 = e$. Hence $e = 1$. But we also must have $e \diamond 2 = 2$. However, for $e = 1$ we get $1 \diamond 2 = 1 \neq 2$. Hence there is no identity element.
- a¤b = a^b . No identity element. Indeed, if e was an identity element, we would have 2¤e = 2, that is $2^e = 2$. This gives $e = 1$. At the same time we must have e¤2 = 2, that is $e^2 = 2$. But for $e = 1$ we get 1¤2 = $1^2 = 1 \neq 2$. Hence there is no identity element.

Example 2.6.

(a) $* = +$: $e = 0$ ($a + 0 = 0 + a = a, \forall a$)

(b) $* = \times$: $e = 1$ ($a1 = 1a = a, \forall a$)

(c) $* = +$ on Z_n : $e = 0$. $* = \times$ on Z_n: $e = 1$.

(d) The identity element for addition of n × n matrices is $e = O_n$ (the zero matrix). The identity element for matrix multiplication on $M_n(R)$ is $e = I_n$ (the identity matrix).

(e) $* = \circ$ on $F(\Omega)$: $e = id$ (the identity map defined by $id(x) = x$ for all $x \in \Omega$). Indeed, for any function $\varphi : \Omega \to \Omega$ we have $\varphi \circ id = id \circ \varphi = \varphi$

Proposition 2.1. If there is an identity element for a binary operation, then this element is unique.

Proof. Suppose e and f are identity elements for a binary operation $*$ on a set S. Then $e * f = e$, since f is an identity element. At the same time, $e * f = f$ since f is an identity element. Hence $e = f$ (since $*$ is a binary operation implying that $e*f$ is unique).

Definition 2.5. For an associative binary operation $*$ on a set S, and a natural number n we define

$$a^n = a * a * \cdots * a \qquad \text{(operated n-times)}$$

Proposition 2.2. Let $*$ be an associative binary operation $*$ on a set S. Then, for all $a \in S$ and all natural numbers m and n, we have

(i) $a^m * a^n = a^{m+n}$

(ii) $(a^m)^n = a^{mn}$

Proof. Left as exercise