First notions of group theory

GU4041, fall 2023

Columbia University

September 12, 2023

Outline

Prime factorization

Definition

1. A *prime number* is an integer $p > 1$ whose only divisors are 1 and p.
2. Two integers m, n are *relatively prime* if their only common factor is 1.

Theorem

Every integer $n > 1$ can be written as a product of prime numbers.

Proof.

We proceed by contradiction. Let $n > 1$ be the smallest integer that cannot be written as a product of prime numbers. If n is prime we have a contradiction. If not, we can factor $n = a \cdot b$ with $1 < a, b < n$. By hypothesis both a and b can be written as products of prime numbers, and so $n = a \cdot b$ can be as well. □

Prime factorization

Definition

1. A *prime number* is an integer $p > 1$ whose only divisors are 1 and p.
2. Two integers m, n are *relatively prime* if their only common factor is 1.

Theorem

Every integer $n > 1$ can be written as a product of prime numbers.

Proof.

We proceed by contradiction. Let $n > 1$ be the smallest integer that cannot be written as a product of prime numbers. If n is prime we have a contradiction. If not, we can factor $n = a \cdot b$ with $1 < a, b < n$. By hypothesis both a and b can be written as products of prime numbers, and so $n = a \cdot b$ can be as well. □

Prime factorization

Definition

1. A *prime number* is an integer $p > 1$ whose only divisors are 1 and p.
2. Two integers m, n are *relatively prime* if their only common factor is 1.

Theorem

Every integer $n > 1$ can be written as a product of prime numbers.

Proof.

We proceed by contradiction. Let $n > 1$ be the smallest integer that cannot be written as a product of prime numbers. If n is prime we have a contradiction. If not, we can factor $n = a \cdot b$ with $1 < a, b < n$. By hypothesis both a and b can be written as products of prime numbers, and so $n = a \cdot b$ can be as well. $\square$

Prime factorization

Definition

1. A *prime number* is an integer $p > 1$ whose only divisors are 1 and p.
2. Two integers m, n are *relatively prime* if their only common factor is 1.

Theorem

Every integer $n > 1$ can be written as a product of prime numbers.

Proof.

We proceed by contradiction. Let $n > 1$ be the smallest integer that cannot be written as a product of prime numbers. If n is prime we have a contradiction. If not, we can factor $n = a \cdot b$ with $1 < a, b < n$. By hypothesis both a and b can be written as products of prime numbers, and so $n = a \cdot b$ can be as well. □

Prime factorization

Definition

1. A *prime number* is an integer $p > 1$ whose only divisors are 1 and p.
2. Two integers m, n are *relatively prime* if their only common factor is 1.

Theorem

Every integer $n > 1$ can be written as a product of prime numbers.

Proof.

We proceed by contradiction. Let $n > 1$ be the smallest integer that cannot be written as a product of prime numbers. If n is prime we have a contradiction. If not, we can factor $n = a \cdot b$ with $1 < a, b < n$. By hypothesis both a and b can be written as products of prime numbers, and so $n = a \cdot b$ can be as well. □

Prime factorization

Definition

1. A *prime number* is an integer $p > 1$ whose only divisors are 1 and p.
2. Two integers m, n are *relatively prime* if their only common factor is 1.

Theorem

Every integer $n > 1$ can be written as a product of prime numbers.

Proof.

We proceed by contradiction. Let $n > 1$ be the smallest integer that cannot be written as a product of prime numbers. If n is prime we have a contradiction. If not, we can factor $n = a \cdot b$ with $1 < a, b < n$. By hypothesis both a and b can be written as products of prime numbers, and so $n = a \cdot b$ can be as well. □

Unique factorization

Theorem (Fundamental theorem of arithmetic)

Every integer $n > 1$ has a unique factorization as a product of prime numbers. More precisely, suppose

$$n = \prod_{i=1}^{r} p_i^{a_i} = \prod_{j=1}^{s} q_j^{b_j}$$

where the p_i and q_j are all primes and the a_i, b_j are positive integers. Then $r = s$, we can assume $p_i = q_i$ for $i = 1, \ldots, r$, up to permutation ; and then $a_i = b_i$.

For the proof, see chapter 2 of Gallagher's notes. If there is time at the end of the course we can review the proof.

Unique factorization

Theorem (Fundamental theorem of arithmetic)

Every integer $n > 1$ has a unique factorization as a product of prime numbers. More precisely, suppose

$$n = \prod_{i=1}^{r} p_i^{a_i} = \prod_{j=1}^{s} q_j^{b_j}$$

where the p_i and q_j are all primes and the a_i, b_j are positive integers. Then $r = s$, we can assume $p_i = q_i$ for $i = 1, \ldots, r$, up to permutation ; and then $a_i = b_i$.

For the proof, see chapter 2 of Gallagher's notes. If there is time at the end of the course we can review the proof.

Unique factorization

Theorem (Fundamental theorem of arithmetic)

Every integer $n > 1$ has a unique factorization as a product of prime numbers. More precisely, suppose

$$n = \prod_{i=1}^{r} p_i^{a_i} = \prod_{j=1}^{s} q_j^{b_j}$$

where the p_i and q_j are all primes and the a_i, b_j are positive integers. Then $r = s$, we can assume $p_i = q_i$ for $i = 1, \ldots, r$, up to permutation ; and then $a_i = b_i$.

For the proof, see chapter 2 of Gallagher's notes. If there is time at the end of the course we can review the proof.

Unique factorization

Theorem (Fundamental theorem of arithmetic)

Every integer $n > 1$ has a unique factorization as a product of prime numbers. More precisely, suppose

$$n = \prod_{i=1}^{r} p_i^{a_i} = \prod_{j=1}^{s} q_j^{b_j}$$

where the p_i and q_j are all primes and the a_i, b_j are positive integers. Then $r = s$, we can assume $p_i = q_i$ for $i = 1, \ldots, r$, up to permutation ; and then $a_i = b_i$.

For the proof, see chapter 2 of Gallagher's notes. If there is time at the end of the course we can review the proof.

Unique factorization

Theorem (Fundamental theorem of arithmetic)

Every integer $n > 1$ has a unique factorization as a product of prime numbers. More precisely, suppose

$$n = \prod_{i=1}^{r} p_i^{a_i} = \prod_{j=1}^{s} q_j^{b_j}$$

where the p_i and q_j are all primes and the a_i, b_j are positive integers. Then $r = s$, we can assume $p_i = q_i$ for $i = 1, \ldots, r$, up to permutation ; and then $a_i = b_i$.

For the proof, see chapter 2 of Gallagher's notes. If there is time at the end of the course we can review the proof.

The greatest common divisor

Definition

Let $m, n \in \mathbb{N}$ The greatest common divisor (GCD) of m and n, denoted $GCD(m,n)$, or simply (m,n), is the largest positive integer d such that d divides both m and n.

One way to find (m,n) is to factor $m = \prod_i p_i^{a_i}$, $n = \prod_i p_i^{b_i}$, where now $a_i, b_i \geq 0$; then

$$GCD(m,n) = \prod_i p_i^{min(a_i,b_i)}.$$

But prime factorization is believed to be computationally hard. (Otherwise there would be no internet security.)
The Euclidean algorithm is much faster and is computationally easy (polynomial time).

The greatest common divisor

Definition

Let $m, n \in \mathbb{N}$ The greatest common divisor (GCD) of m and n, denoted $GCD(m,n)$, or simply (m,n), is the largest positive integer d such that d divides both m and n.

One way to find (m,n) is to factor $m = \prod_i p_i^{a_i}$, $n = \prod_i p_i^{b_i}$, where now $a_i, b_i \geq 0$; then

$$GCD(m,n) = \prod_i p_i^{min(a_i,b_i)}.$$

But prime factorization is believed to be computationally hard. (Otherwise there would be no internet security.)

The Euclidean algorithm is much faster and is computationally easy (polynomial time).

The greatest common divisor

Definition

Let $m, n \in \mathbb{N}$ The greatest common divisor (GCD) of m and n, denoted $GCD(m, n)$, or simply (m, n), is the largest positive integer d such that d divides both m and n.

One way to find (m, n) is to factor $m = \prod_i p_i^{a_i}$, $n = \prod_i p_i^{b_i}$, where now $a_i, b_i \geq 0$; then

$$GCD(m, n) = \prod_i p_i^{min(a_i, b_i)}.$$

But prime factorization is believed to be computationally hard. (Otherwise there would be no internet security.)

The Euclidean algorithm is much faster and is computationally easy (polynomial time).

The greatest common divisor

Definition

Let $m, n \in \mathbb{N}$ The greatest common divisor (GCD) of m and n, denoted $GCD(m, n)$, or simply (m, n), is the largest positive integer d such that d divides both m and n.

One way to find (m, n) is to factor $m = \prod_i p_i^{a_i}$, $n = \prod_i p_i^{b_i}$, where now $a_i, b_i \geq 0$; then

$$GCD(m, n) = \prod_i p_i^{min(a_i, b_i)}.$$

But prime factorization is believed to be computationally hard. (Otherwise there would be no internet security.)
The Euclidean algorithm is much faster and is computationally easy (polynomial time).

Euclidean algorithm, part 1

We assume $n \geq m$. Write $n_1 = n$, $m_1 = m$ and divide the larger by the smaller:

$$n_1 = d_1 \cdot m_1 + r_1$$

where r_1 is the remainder.

Of course $r_1 < m_1$. So now set $n_2 = m_1, m_2 = r_1$, and write

$$n_2 = d_2 \cdot m_2 + r_2.$$

Set $n_3 = m_2, m_3 = r_2$ and continue in this way until we find $n_k = d_k \cdot m_k$ without remainder.

We claim that $m_k = GCD(m, n)$.

First: m_k divides $n_k = m_{k-1}$; but

$$m_k = r_{k-1} = n_{k-1} - d_{k-1}m_{k-1} = n_{k-1} - d_{k-1}n_k.$$

So m_k divides n_{k-1}. By induction we see m_k divides all the n_i and m_i, hence divides m and n.

Euclidean algorithm, part 1

We assume $n \geq m$. Write $n_1 = n$, $m_1 = m$ and divide the larger by the smaller:

$$n_1 = d_1 \cdot m_1 + r_1$$

where r_1 is the remainder.
Of course $r_1 < m_1$. So now set $n_2 = m_1, m_2 = r_1$, and write

$$n_2 = d_2 \cdot m_2 + r_2.$$

Set $n_3 = m_2, m_3 = r_2$ and continue in this way until we find $n_k = d_k \cdot m_k$ without remainder.
We claim that $m_k = GCD(m, n)$.
First: m_k divides $n_k = m_{k-1}$; but

$$m_k = r_{k-1} = n_{k-1} - d_{k-1}m_{k-1} = n_{k-1} - d_{k-1}n_k.$$

So m_k divides n_{k-1}. By induction we see m_k divides all the n_i and m_i, hence divides m and n.

Euclidean algorithm, part 1

We assume $n \geq m$. Write $n_1 = n$, $m_1 = m$ and divide the larger by the smaller:

$$n_1 = d_1 \cdot m_1 + r_1$$

where r_1 is the remainder.
Of course $r_1 < m_1$. So now set $n_2 = m_1, m_2 = r_1$, and write

$$n_2 = d_2 \cdot m_2 + r_2.$$

Set $n_3 = m_2, m_3 = r_2$ and continue in this way until we find $n_k = d_k \cdot m_k$ without remainder.
We claim that $m_k = GCD(m, n)$.
First: m_k divides $n_k = m_{k-1}$; but

$$m_k = r_{k-1} = n_{k-1} - d_{k-1}m_{k-1} = n_{k-1} - d_{k-1}n_k.$$

So m_k divides n_{k-1}. By induction we see m_k divides all the n_i and m_i, hence divides m and n.

Euclidean algorithm, part 1

We assume $n \geq m$. Write $n_1 = n$, $m_1 = m$ and divide the larger by the smaller:

$$n_1 = d_1 \cdot m_1 + r_1$$

where r_1 is the remainder.
Of course $r_1 < m_1$. So now set $n_2 = m_1, m_2 = r_1$, and write

$$n_2 = d_2 \cdot m_2 + r_2.$$

Set $n_3 = m_2, m_3 = r_2$ and continue in this way until we find $n_k = d_k \cdot m_k$ without remainder.
We claim that $m_k = GCD(m, n)$.
First: m_k divides $n_k = m_{k-1}$; but

$$m_k = r_{k-1} = n_{k-1} - d_{k-1}m_{k-1} = n_{k-1} - d_{k-1}n_k.$$

So m_k divides n_{k-1}. By induction we see m_k divides all the n_i and m_i, hence divides m and n.

Euclidean algorithm, part 1

We assume $n \geq m$. Write $n_1 = n$, $m_1 = m$ and divide the larger by the smaller:

$$n_1 = d_1 \cdot m_1 + r_1$$

where r_1 is the remainder.
Of course $r_1 < m_1$. So now set $n_2 = m_1, m_2 = r_1$, and write

$$n_2 = d_2 \cdot m_2 + r_2.$$

Set $n_3 = m_2, m_3 = r_2$ and continue in this way until we find $n_k = d_k \cdot m_k$ without remainder.
We claim that $m_k = GCD(m, n)$.
First: m_k divides $n_k = m_{k-1}$; but

$$m_k = r_{k-1} = n_{k-1} - d_{k-1}m_{k-1} = n_{k-1} - d_{k-1}n_k.$$

So m_k divides n_{k-1}. By induction we see m_k divides all the n_i and m_i, hence divides m and n.

Euclidean algorithm, part 1

We assume $n \geq m$. Write $n_1 = n$, $m_1 = m$ and divide the larger by the smaller:

$$n_1 = d_1 \cdot m_1 + r_1$$

where r_1 is the remainder.
Of course $r_1 < m_1$. So now set $n_2 = m_1$, $m_2 = r_1$, and write

$$n_2 = d_2 \cdot m_2 + r_2.$$

Set $n_3 = m_2, m_3 = r_2$ and continue in this way until we find $n_k = d_k \cdot m_k$ without remainder.
We claim that $m_k = GCD(m, n)$.
First: m_k divides $n_k = m_{k-1}$; but

$$m_k = r_{k-1} = n_{k-1} - d_{k-1}m_{k-1} = n_{k-1} - d_{k-1}n_k.$$

So m_k divides n_{k-1}. By induction we see m_k divides all the n_i and m_i, hence divides m and n.

Euclidean algorithm, part 1

We assume $n \geq m$. Write $n_1 = n$, $m_1 = m$ and divide the larger by the smaller:

$$n_1 = d_1 \cdot m_1 + r_1$$

where r_1 is the remainder.
Of course $r_1 < m_1$. So now set $n_2 = m_1, m_2 = r_1$, and write

$$n_2 = d_2 \cdot m_2 + r_2.$$

Set $n_3 = m_2, m_3 = r_2$ and continue in this way until we find $n_k = d_k \cdot m_k$ without remainder.
We claim that $m_k = GCD(m, n)$.
First: m_k divides $n_k = m_{k-1}$; but

$$m_k = r_{k-1} = n_{k-1} - d_{k-1}m_{k-1} = n_{k-1} - d_{k-1}n_k.$$

So m_k divides n_{k-1}. By induction we see m_k divides all the n_i and m_i, hence divides m and n.

Euclidean algorithm, part 2

To show that m_k is the GCD, we need to show that if a is any divisor of m and n then a divides m_k. For this we show that there are integers α, β such that

$$m_k = \alpha \cdot n + \beta \cdot m.$$

This is also proved by induction: we show that every m_i and n_j is a linear combination of n and m with integer coefficients.

$$m_2 = r_1 = n - d_1 \cdot m$$

$$m_3 = r_2 = m - d_2 \cdot m_2 = m - d_2 \cdot (n - d_1 \cdot m);$$

and so on.

If a divides n and m then a divides $\alpha \cdot n + \beta \cdot m = m_k$.

Euclidean algorithm, part 2

To show that m_k is the GCD, we need to show that if a is any divisor of m and n then a divides m_k. For this we show that there are integers α, β such that

$$m_k = \alpha \cdot n + \beta \cdot m.$$

This is also proved by induction: we show that every m_i and n_j is a linear combination of n and m with integer coefficients.

$$m_2 = r_1 = n - d_1 \cdot m$$

$$m_3 = r_2 = m - d_2 \cdot m_2 = m - d_2 \cdot (n - d_1 \cdot m);$$

and so on.
If a divides n and m then a divides $\alpha \cdot n + \beta \cdot m = m_k$.

Euclidean algorithm, part 2

To show that m_k is the GCD, we need to show that if a is any divisor of m and n then a divides m_k. For this we show that there are integers α, β such that

$$m_k = \alpha \cdot n + \beta \cdot m.$$

This is also proved by induction: we show that every m_i and n_j is a linear combination of n and m with integer coefficients.

$$m_2 = r_1 = n - d_1 \cdot m$$

$$m_3 = r_2 = m - d_2 \cdot m_2 = m - d_2 \cdot (n - d_1 \cdot m);$$

and so on.

If a divides n and m then a divides $\alpha \cdot n + \beta \cdot m = m_k$.

Euclidean algorithm, part 2

To show that m_k is the GCD, we need to show that if a is any divisor of m and n then a divides m_k. For this we show that there are integers α, β such that

$$m_k = \alpha \cdot n + \beta \cdot m.$$

This is also proved by induction: we show that every m_i and n_j is a linear combination of n and m with integer coefficients.

$$m_2 = r_1 = n - d_1 \cdot m$$

$$m_3 = r_2 = m - d_2 \cdot m_2 = m - d_2 \cdot (n - d_1 \cdot m);$$

and so on.
If a divides n and m then a divides $\alpha \cdot n + \beta \cdot m = m_k$.

Euclidean algorithm, example

We compute $GCD(88, 24)$:

$$88 = 3 \cdot 24 + 16.$$

$$24 = 1 \cdot 16 + 8.$$

$$16 = 2 \cdot 8 + 0.$$

Hence $8 = GCD(88, 24)$.

Bezout's theorem

Theorem

Suppose $GCD(m, n) = 1$. *Then there are* $\alpha, \beta \in \mathbb{Z}$ *such that*

$$\alpha m + \beta n = 1.$$

This is just a special case of the Euclidean algorithm.

Bezout's theorem

Theorem

Suppose $GCD(m, n) = 1$*. Then there are* $\alpha, \beta \in \mathbb{Z}$ *such that*

$$\alpha m + \beta n = 1.$$

This is just a special case of the Euclidean algorithm.

Gauss's lemma

Theorem (Gauss lemma)

Suppose $a, b, c \in \mathbb{Z}$.
Suppose $a | b \cdot c$ but $GCD(a, c) = 1$. Then a divides b.
In particular, if p is prime and divides $b \cdot c$, then either p divides b or p divides c.

The proof is as follows: By Bezout, there are α, β in $\mathbb{Z}$ such that $\alpha a + \beta c = 1$. Multiply both sides by b:

$$\alpha \cdot ab + \beta \cdot bc = b.$$

a divides $\alpha \cdot ab$, a divides $\beta \cdot bc \Rightarrow a$ divides $\alpha \cdot ab + \beta \cdot bc = b$.

Gauss's lemma

Theorem (Gauss lemma)

Suppose $a, b, c \in \mathbb{Z}$.
Suppose $a | b \cdot c$ but $GCD(a, c) = 1$. Then a divides b.
In particular, if p is prime and divides $b \cdot c$, then either p divides b or p divides c.

The proof is as follows: By Bezout, there are α, β in $\mathbb{Z}$ such that $\alpha a + \beta c = 1$. Multiply both sides by b:

$$\alpha \cdot ab + \beta \cdot bc = b.$$

a divides $\alpha \cdot ab$, a divides $\beta \cdot bc \Rightarrow a$ divides $\alpha \cdot ab + \beta \cdot bc = b$.

Least common multiple

We define

$$LCM(m,n) = \frac{m \cdot n}{GCD(m,n)}.$$

Exercise: Show that $LCM(m,n)$, defined in this way, is the least common multiple of m and n.

Least common multiple

We define

$$LCM(m,n) = \frac{m \cdot n}{GCD(m,n)}.$$

Exercise: Show that $LCM(m,n)$, defined in this way, is the least common multiple of m and n.

Congruences

Let $n > 1$ be an integer. We define an equivalence relation $\sim_n$ on $\mathbb{Z}$: write

$$a \sim_n b$$

if n divides $a - b$. More commonly, we write $a \equiv b \pmod{n}$.

- Reflexive: for any a, $n \mid (a - a)$, so $a \sim_n a$.
- Symmetric: if $n \mid (a - b)$ then $n \mid (b - a)$.
- Transitive: if $n \mid (a - b)$ and $n \mid (b - c)$ then n divides $(a - b) + (b - c) = a - c$.

The set of equivalence classes $\mathbb{Z}/\sim_n$ – also called *congruence classes*, or *residue classes* – is denoted $\mathbb{Z}_n$ (later C_n).
If $a \in \mathbb{N}$, write $a = d \cdot n + r$; then $r \in \{0, 1, \ldots n - 1\}$, so $a \sim_n r$.
Thus $|\mathbb{Z}_n| = n$ (Check that this works also for negative a.)

Congruences

Let $n > 1$ be an integer. We define an equivalence relation $\sim_n$ on $\mathbb{Z}$: write

$$a \sim_n b$$

if n divides $a - b$. More commonly, we write $a \equiv b \pmod{n}$.

- Reflexive: for any a, $n \mid (a - a)$, so $a \sim_n a$.
- Symmetric: if $n \mid (a - b)$ then $n \mid (b - a)$.
- Transitive: if $n \mid (a - b)$ and $n \mid (b - c)$ then n divides $(a - b) + (b - c) = a - c$.

The set of equivalence classes $\mathbb{Z}/\sim_n$ – also called *congruence classes*, or *residue classes* – is denoted $\mathbb{Z}_n$ (later C_n).
If $a \in \mathbb{N}$, write $a = d \cdot n + r$; then $r \in \{0, 1, \ldots n - 1\}$, so $a \sim_n r$.
Thus $|\mathbb{Z}_n| = n$ (Check that this works also for negative a.)

Congruences

Let $n > 1$ be an integer. We define an equivalence relation $\sim_n$ on $\mathbb{Z}$: write

$$a \sim_n b$$

if n divides $a - b$. More commonly, we write $a \equiv b \pmod{n}$.

- Reflexive: for any a, $n \mid (a - a)$, so $a \sim_n a$.
- Symmetric: if $n \mid (a - b)$ then $n \mid (b - a)$.
- Transitive: if $n \mid (a - b)$ and $n \mid (b - c)$ then n divides $(a - b) + (b - c) = a - c$.

The set of equivalence classes $\mathbb{Z}/\sim_n$ – also called *congruence classes*, or *residue classes* – is denoted $\mathbb{Z}_n$ (later C_n).

If $a \in \mathbb{N}$, write $a = d \cdot n + r$; then $r \in \{0, 1, \ldots n - 1\}$, so $a \sim_n r$.

Thus $|\mathbb{Z}_n| = n$ (Check that this works also for negative a.)

Congruences

Let $n > 1$ be an integer. We define an equivalence relation $\sim_n$ on $\mathbb{Z}$: write

$$a \sim_n b$$

if n divides $a - b$. More commonly, we write $a \equiv b \pmod{n}$.

- Reflexive: for any a, $n \mid (a - a)$, so $a \sim_n a$.
- Symmetric: if $n \mid (a - b)$ then $n \mid (b - a)$.
- Transitive: if $n \mid (a - b)$ and $n \mid (b - c)$ then n divides $(a - b) + (b - c) = a - c$.

The set of equivalence classes $\mathbb{Z}/\sim_n$ – also called *congruence classes*, or *residue classes* – is denoted $\mathbb{Z}_n$ (later C_n).
If $a \in \mathbb{N}$, write $a = d \cdot n + r$; then $r \in \{0, 1, \ldots n - 1\}$, so $a \sim_n r$.
Thus $|\mathbb{Z}_n| = n$ (Check that this works also for negative a.)

Congruences

Let $n > 1$ be an integer. We define an equivalence relation $\sim_n$ on $\mathbb{Z}$: write

$$a \sim_n b$$

if n divides $a - b$. More commonly, we write $a \equiv b \pmod{n}$.

- Reflexive: for any a, $n \mid (a - a)$, so $a \sim_n a$.
- Symmetric: if $n \mid (a - b)$ then $n \mid (b - a)$.
- Transitive: if $n \mid (a - b)$ and $n \mid (b - c)$ then n divides $(a - b) + (b - c) = a - c$.

The set of equivalence classes $\mathbb{Z}/\sim_n$ – also called *congruence classes*, or *residue classes* – is denoted $\mathbb{Z}_n$ (later C_n).

If $a \in \mathbb{N}$, write $a = d \cdot n + r$; then $r \in \{0, 1, \ldots n - 1\}$, so $a \sim_n r$. Thus $|\mathbb{Z}_n| = n$ (Check that this works also for negative a.)

Congruences

Let $n > 1$ be an integer. We define an equivalence relation $\sim_n$ on $\mathbb{Z}$: write

$$a \sim_n b$$

if n divides $a - b$. More commonly, we write $a \equiv b \pmod{n}$.

- Reflexive: for any a, $n \mid (a - a)$, so $a \sim_n a$.
- Symmetric: if $n \mid (a - b)$ then $n \mid (b - a)$.
- Transitive: if $n \mid (a - b)$ and $n \mid (b - c)$ then n divides $(a - b) + (b - c) = a - c$.

The set of equivalence classes $\mathbb{Z}/\sim_n$ – also called *congruence classes*, or *residue classes* – is denoted $\mathbb{Z}_n$ (later C_n).
If $a \in \mathbb{N}$, write $a = d \cdot n + r$; then $r \in \{0, 1, \ldots n - 1\}$, so $a \sim_n r$.
Thus $|\mathbb{Z}_n| = n$ (Check that this works also for negative a.)

Residue classes, examples

Example

For $n = 2$, there are two residue classes: the set of odd or even numbers.

Example

For $n = 10$, any integer a is in the residue class of its last digit:

$$197865493 \equiv 3 \pmod{10}.$$

Example

For $n = 12$, congruence mod 12 is the basis of telling time on a clock.

Residue classes, examples

Example

For $n = 2$, there are two residue classes: the set of odd or even numbers.

Example

For $n = 10$, any integer a is in the residue class of its last digit:

$$197865493 \equiv 3 \pmod{10}.$$

Example

For $n = 12$, congruence mod 12 is the basis of telling time on a clock.

Residue classes, examples

Example

For $n = 2$, there are two residue classes: the set of odd or even numbers.

Example

For $n = 10$, any integer a is in the residue class of its last digit:

$$197865493 \equiv 3 \pmod{10}.$$

Example

For $n = 12$, congruence mod 12 is the basis of telling time on a clock.

A word problem

At $3:00$ I take a bus to Denver. The trip takes 42 hours and the time is 2 hours earlier. What time is it when I arrive?

Answer: $3 + 42 - 2 \equiv 7 \pmod{12}$. So it is $7:00$.

This is a calculation in *arithmetic modulo* 12.

A word problem

At $3:00$ I take a bus to Denver. The trip takes 42 hours and the time is 2 hours earlier. What time is it when I arrive?

Answer: $3 + 42 - 2 \equiv 7 \pmod{12}$. So it is $7:00$.

This is a calculation in *arithmetic modulo* 12.

A word problem

At $3:00$ I take a bus to Denver. The trip takes 42 hours and the time is 2 hours earlier. What time is it when I arrive?

Answer: $3 + 42 - 2 \equiv 7 \pmod{12}$. So it is $7:00$.

This is a calculation in *arithmetic modulo* 12.

Arithmetic modulo n

We know there is a function from $\mathbb{Z}$ to the set of equivalence classes

$$r_n : \mathbb{Z} \to \mathbb{Z}_n = \mathbb{Z}/\sim_n .$$

For any $a \in \mathbb{Z}$, we write $[a]_n = r_n(a)$ for the equivalence class in $\mathbb{Z}_n$ containing a.

Now we can define

$$[a]_n + [b]_n = [a+b]_n; [a]_n \cdot [b]_n = [a \cdot b]_n.$$

Thus for example

$$[3]_{12} + [42]_{12} - [2]_{12} = [43]_{12}.$$

Not practical for telling time!

Arithmetic modulo n

We know there is a function from $\mathbb{Z}$ to the set of equivalence classes

$$r_n : \mathbb{Z} \to \mathbb{Z}_n = \mathbb{Z}/ \sim_n .$$

For any $a \in \mathbb{Z}$, we write $[a]_n = r_n(a)$ for the equivalence class in $\mathbb{Z}_n$ containing a.

Now we can define

$$[a]_n + [b]_n = [a+b]_n; [a]_n \cdot [b]_n = [a \cdot b]_n.$$

Thus for example

$$[3]_{12} + [42]_{12} - [2]_{12} = [43]_{12}.$$

Not practical for telling time!

Arithmetic modulo n

We know there is a function from $\mathbb{Z}$ to the set of equivalence classes

$$r_n : \mathbb{Z} \to \mathbb{Z}_n = \mathbb{Z}/\sim_n .$$

For any $a \in \mathbb{Z}$, we write $[a]_n = r_n(a)$ for the equivalence class in $\mathbb{Z}_n$ containing a.
Now we can define

$$[a]_n + [b]_n = [a+b]_n; [a]_n \cdot [b]_n = [a \cdot b]_n.$$

Thus for example

$$[3]_{12} + [42]_{12} - [2]_{12} = [43]_{12}.$$

Not practical for telling time!

Arithmetic modulo n

We know there is a function from $\mathbb{Z}$ to the set of equivalence classes

$$r_n : \mathbb{Z} \to \mathbb{Z}_n = \mathbb{Z}/\sim_n .$$

For any $a \in \mathbb{Z}$, we write $[a]_n = r_n(a)$ for the equivalence class in $\mathbb{Z}_n$ containing a.
Now we can define

$$[a]_n + [b]_n = [a+b]_n; [a]_n \cdot [b]_n = [a \cdot b]_n.$$

Thus for example

$$[3]_{12} + [42]_{12} - [2]_{12} = [43]_{12}.$$

Not practical for telling time!

Arithmetic modulo n is well defined

Suppose $[a]_n = [a']_n, [b]_n = [b']_n$. We need to show that

$$[ab]_n = [a'b']_n, [a+b]_n = [a'+b']_n.$$

Check for multiplication (more difficult)

$$[a]_n = [a']_n \Rightarrow n \mid (a - a') \Rightarrow (a - a') = dn$$

So $a = a' + dn; b = b' + en$,
So

$$ab = (a' + dn)(b' + en) = a'b' + n(db' + ea' + den) \equiv a'b' \pmod{n}.$$

Arithmetic modulo n is well defined

Suppose $[a]_n = [a']_n, [b]_n = [b']_n$. We need to show that

$$[ab]_n = [a'b']_n, [a+b]_n = [a'+b']_n.$$

Check for multiplication (more difficult)

$$[a]_n = [a']_n \Rightarrow n \mid (a-a') \Rightarrow (a-a') = dn$$

So $a = a' + dn; b = b' + en,$
So

$$ab = (a'+dn)(b'+en) = a'b' + n(db' + ea' + den) \equiv a'b' \pmod{n}.$$

Arithmetic modulo n is well defined

Suppose $[a]_n = [a']_n, [b]_n = [b']_n$. We need to show that

$$[ab]_n = [a'b']_n, [a+b]_n = [a'+b']_n.$$

Check for multiplication (more difficult)

$$[a]_n = [a']_n \Rightarrow n \mid (a-a') \Rightarrow (a-a') = dn$$

So $a = a' + dn; b = b' + en$,
So

$$ab = (a'+dn)(b'+en) = a'b' + n(db' + ea' + den) \equiv a'b' \pmod{n}.$$

Arithmetic modulo n with representatives

We choose one representative in each residue class, usually

$$\{[0]_n, [1]_n, \dots, [n-1]_n\}.$$

Then to compute $[a]_n + [b]_n$, when $0 \leq a, b < n$

- if $a + b < n$ then $[a]_n + [b]_n = [a + b]_n$ is the chosen representative;
- if $n < a + b < 2n$ then $[a]_n + [b]_n = [a + b - n]_n$.

For multiplication, you have $ab = dn + r$ with $0 \leq r < n$ the remainder, so

$$[a]_n[b]_n = [r]_n.$$

Arithmetic modulo n with representatives

We choose one representative in each residue class, usually

$$\{[0]_n, [1]_n, \ldots, [n-1]_n\}.$$

Then to compute $[a]_n + [b]_n$, when $0 \leq a, b < n$

- if $a + b < n$ then $[a]_n + [b]_n = [a+b]_n$ is the chosen representative;
- if $n < a + b < 2n$ then $[a]_n + [b]_n = [a+b-n]_n$.

For multiplication, you have $ab = dn + r$ with $0 \leq r < n$ the remainder, so

$$[a]_n[b]_n = [r]_n.$$

Arithmetic modulo n with representatives

We choose one representative in each residue class, usually

$$\{[0]_n, [1]_n, \dots, [n-1]_n\}.$$

Then to compute $[a]_n + [b]_n$, when $0 \leq a, b < n$

- if $a + b < n$ then $[a]_n + [b]_n = [a+b]_n$ is the chosen representative;
- if $n < a + b < 2n$ then $[a]_n + [b]_n = [a+b-n]_n$.

For multiplication, you have $ab = dn + r$ with $0 \leq r < n$ the remainder, so

$$[a]_n[b]_n = [r]_n.$$

Arithmetic modulo n with representatives

We choose one representative in each residue class, usually

$$\{[0]_n, [1]_n, \ldots, [n-1]_n\}.$$

Then to compute $[a]_n + [b]_n$, when $0 \leq a, b < n$

- if $a + b < n$ then $[a]_n + [b]_n = [a+b]_n$ is the chosen representative;
- if $n < a + b < 2n$ then $[a]_n + [b]_n = [a+b-n]_n$.

For multiplication, you have $ab = dn + r$ with $0 \leq r < n$ the remainder, so

$$[a]_n[b]_n = [r]_n.$$

Arithmetic modulo n with representatives

We choose one representative in each residue class, usually

$$\{[0]_n, [1]_n, \ldots, [n-1]_n\}.$$

Then to compute $[a]_n + [b]_n$, when $0 \leq a, b < n$

- if $a + b < n$ then $[a]_n + [b]_n = [a + b]_n$ is the chosen representative;
- if $n < a + b < 2n$ then $[a]_n + [b]_n = [a + b - n]_n$.

For multiplication, you have $ab = dn + r$ with $0 \leq r < n$ the remainder, so

$$[a]_n[b]_n = [r]_n.$$

A corollary to Bezout's theorem

Recall that if $GCD(a,n) = 1$ then there are integers α, β such that

$$\alpha \cdot a + \beta \cdot n = 1.$$

Thus

$$[\alpha]_n \cdot [a]_n = [1]_n - [\beta \cdot n]_n = [1]_n.$$

In other words, if $(a,n) = 1$ then $[a]_n$ has a *multiplicative inverse* in $\mathbb{Z}_n$.

A corollary to Bezout's theorem

Recall that if $GCD(a, n) = 1$ then there are integers α, β such that

$$\alpha \cdot a + \beta \cdot n = 1.$$

Thus

$$[\alpha]_n \cdot [a]_n = [1]_n - [\beta \cdot n]_n = [1]_n.$$

In other words, if $(a, n) = 1$ then $[a]_n$ has a *multiplicative inverse* in $\mathbb{Z}_n$.

A corollary to Bezout's theorem

Recall that if $GCD(a, n) = 1$ then there are integers α, β such that

$$\alpha \cdot a + \beta \cdot n = 1.$$

Thus

$$[\alpha]_n \cdot [a]_n = [1]_n - [\beta \cdot n]_n = [1]_n.$$

In other words, if $(a, n) = 1$ then $[a]_n$ has a *multiplicative inverse* in $\mathbb{Z}_n$.

Definition of a group

The set $\mathbb{Z}_n$ with addition is the simplest example of a finite group.

Definition

A *binary operation* on a set G is a function $m : G \times G \to G$.

Definition

A *group* is a set G with a binary operation m, where we write $m(g,h) = gh = g \cdot h$, an element $e \in G$, and a function

$$\iota : G \to G, \text{ written } \iota(g) = g^{-1},$$

satisfying these axioms:

- Associativity: $\forall g_1, g_2, g_3 \in G$, $(g_1 g_2) g_3 = g_1(g_2 g_3)$;
- Identity: $\forall g \in G$, $eg = g$;
- Inverse: $\forall g \in G$, $g^{-1}g = e$.

Definition of a group

The set $\mathbb{Z}_n$ with addition is the simplest example of a finite group.

Definition

A *binary operation* on a set G is a function $m : G \times G \to G$.

Definition

A *group* is a set G with a binary operation m, where we write $m(g,h) = gh = g \cdot h$, an element $e \in G$, and a function

$$\iota : G \to G, \text{ written } \iota(g) = g^{-1},$$

satisfying these axioms:

- Associativity: $\forall g_1, g_2, g_3 \in G$, $(g_1 g_2) g_3 = g_1(g_2 g_3)$;
- Identity: $\forall g \in G$, $eg = g$;
- Inverse: $\forall g \in G$, $g^{-1} g = e$.

Definition of a group

The set $\mathbb{Z}_n$ with addition is the simplest example of a finite group.

Definition

A *binary operation* on a set G is a function $m : G \times G \to G$.

Definition

A *group* is a set G with a binary operation m, where we write $m(g, h) = gh = g \cdot h$, an element $e \in G$, and a function

$$\iota : G \to G, \text{ written } \iota(g) = g^{-1},$$

satisfying these axioms:

- Associativity: $\forall g_1, g_2, g_3 \in G$, $(g_1 g_2) g_3 = g_1 (g_2 g_3)$;
- Identity: $\forall g \in G$, $eg = g$;
- Inverse: $\forall g \in G$, $g^{-1} g = e$.

Definition of a group

The set $\mathbb{Z}_n$ with addition is the simplest example of a finite group.

Definition

A *binary operation* on a set G is a function $m : G \times G \to G$.

Definition

A *group* is a set G with a binary operation m, where we write $m(g,h) = gh = g \cdot h$, an element $e \in G$, and a function

$$\iota : G \to G, \text{ written } \iota(g) = g^{-1},$$

satisfying these axioms:

- Associativity: $\forall g_1, g_2, g_3 \in G$, $(g_1 g_2) g_3 = g_1 (g_2 g_3)$;
- Identity: $\forall g \in G$, $eg = g$;
- Inverse: $\forall g \in G$, $g^{-1} g = e$.

Definition of a group

The set $\mathbb{Z}_n$ with addition is the simplest example of a finite group.

Definition

A *binary operation* on a set G is a function $m : G \times G \to G$.

Definition

A *group* is a set G with a binary operation m, where we write $m(g,h) = gh = g \cdot h$, an element $e \in G$, and a function

$$\iota : G \to G, \text{ written } \iota(g) = g^{-1},$$

satisfying these axioms:

- Associativity: $\forall g_1, g_2, g_3 \in G$, $(g_1 g_2) g_3 = g_1 (g_2 g_3)$;
- Identity: $\forall g \in G$, $eg = g$;
- Inverse: $\forall g \in G$, $g^{-1} g = e$.

Elementary properties

For all $g \in G$, $gg^{-1} = e$.

Proof: Let $h = gg^{-1}$. We write

$$g^{-1}h = g^{-1}(gg^{-1}) = (g^{-1}g)g^{-1} \text{ [associative law]}$$

$$g^{-1}h = e \cdot g^{-1} \text{ [inverse]}$$

$$(*) \qquad g^{-1}h = g^{-1}. \text{ [identity]}$$

So

$$h = eh = ((g^{-1})^{-1} \cdot g^{-1}) \cdot h \text{ [identity and inverse]}$$

$$h = (g^{-1})^{-1}(g^{-1} \cdot h) \text{ [associative law]}$$

$$h = (g^{-1})^{-1}g^{-1} \text{ [by (*)]}$$

$$h = e \text{ [inverse]}$$

Elementary properties

For all $g \in G$, $gg^{-1} = e$.
Proof: Let $h = gg^{-1}$. We write

$$g^{-1}h = g^{-1}(gg^{-1}) = (g^{-1}g)g^{-1} \text{ [associative law]}$$

$$g^{-1}h = e \cdot g^{-1} \text{ [inverse]}$$

$$(*) \qquad g^{-1}h = g^{-1}. \text{ [identity]}$$

So

$$h = eh = ((g^{-1})^{-1} \cdot g^{-1}) \cdot h \text{ [identity and inverse]}$$

$$h = (g^{-1})^{-1}(g^{-1} \cdot h) \text{ [associative law]}$$

$$h = (g^{-1})^{-1}g^{-1} \text{ [by (*)]}$$

$$h = e \text{ [inverse]}$$

Elementary properties

For all $g \in G$, $gg^{-1} = e$.

Proof: Let $h = gg^{-1}$. We write

$$g^{-1}h = g^{-1}(gg^{-1}) = (g^{-1}g)g^{-1} \text{ [associative law]}$$

$$g^{-1}h = e \cdot g^{-1} \text{ [inverse]}$$

$$(*) \qquad g^{-1}h = g^{-1}. \text{ [identity]}$$

So

$$h = eh = ((g^{-1})^{-1} \cdot g^{-1}) \cdot h \text{ [identity and inverse]}$$

$$h = (g^{-1})^{-1}(g^{-1} \cdot h) \text{ [associative law]}$$

$$h = (g^{-1})^{-1}g^{-1} \text{ [by (*)]}$$

$$h = e \text{ [inverse]}$$

Elementary properties

For all $g \in G$, $gg^{-1} = e$.
Proof: Let $h = gg^{-1}$. We write

$$g^{-1}h = g^{-1}(gg^{-1}) = (g^{-1}g)g^{-1} \text{ [associative law]}$$

$$g^{-1}h = e \cdot g^{-1} \text{ [inverse]}$$

$$(*) \qquad g^{-1}h = g^{-1}. \text{ [identity]}$$

So

$$h = eh = ((g^{-1})^{-1} \cdot g^{-1}) \cdot h \text{ [identity and inverse]}$$

$$h = (g^{-1})^{-1}(g^{-1} \cdot h) \text{ [associative law]}$$

$$h = (g^{-1})^{-1}g^{-1} \text{ [by (*)]}$$

$$h = e \text{ [inverse]}$$

Elementary properties

For all $g \in G$, $gg^{-1} = e$.

Proof: Let $h = gg^{-1}$. We write

$$g^{-1}h = g^{-1}(gg^{-1}) = (g^{-1}g)g^{-1} \text{ [associative law]}$$

$$g^{-1}h = e \cdot g^{-1} \text{ [inverse]}$$

$$(*) \qquad g^{-1}h = g^{-1}. \text{ [identity]}$$

So

$$h = eh = ((g^{-1})^{-1} \cdot g^{-1}) \cdot h \text{ [identity and inverse]}$$

$$h = (g^{-1})^{-1}(g^{-1} \cdot h) \text{ [associative law]}$$

$$h = (g^{-1})^{-1}g^{-1} \text{ [by (*)]}$$

$$h = e \text{ [inverse]}$$

Elementary properties

Similarly, the identity axiom states $eg = g$, but in fact

$$\forall g, ge = g.$$

Indeed,

$$ge = g(g^{-1}g) = (gg^{-1})g \text{ [associative law]}$$

$$ge = eg \text{ [as we just showed]}$$

$$ge = g \text{ [by the identity axiom]}$$

Elementary properties

Similarly, the identity axiom states $eg = g$, but in fact

$$\forall g, ge = g.$$

Indeed,

$$ge = g(g^{-1}g) = (gg^{-1})g \text{ [associative law]}$$

$$ge = eg \text{ [as we just showed]}$$

$$ge = g \text{ [by the identity axiom]}$$

Elementary properties, exercises

Exercise

(1) Show that, for any g, e is the unique element such that $eg = g$.
(2) Show that, for any g, there is a unique element j such that $gj = e$ (and thus $j = g^{-1}$.

Commutative groups

Definition

The group G is *commutative* if, for all $g, h \in G$, $gh = hg$.

Familiar examples: $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{R}$ are commutative groups under the addition law.

Theorem

The set $\mathbb{Z}_n$ with addition is a group.

Proof.

Associativity of addition in $\mathbb{Z}_n$ follows from that in $\mathbb{Z}$:

$$([a]_n + [b]_n) + [c]_n = [a+b]_n + [c]_n = [(a+b)+c]_n = [a+(b+c)]_n \ldots$$

The element $[0]_n$ is the identity; the inverse of $[a]_n$ is $[-a]_n$. □

Commutative groups

Definition

The group G is *commutative* if, for all $g, h \in G$, $gh = hg$.

Familiar examples: $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{R}$ are commutative groups under the addition law.

Theorem

The set $\mathbb{Z}_n$ with addition is a group.

Proof.

Associativity of addition in $\mathbb{Z}_n$ follows from that in $\mathbb{Z}$:

$$([a]_n+[b]_n)+[c]_n = [a+b]_n+[c]_n = [(a+b)+c]_n = [a+(b+c)]_n \ldots$$

The element $[0]_n$ is the identity; the inverse of $[a]_n$ is $[-a]_n$. □

Commutative groups

Definition

The group G is *commutative* if, for all $g,h \in G$, $gh = hg$.

Familiar examples: $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{R}$ are commutative groups under the addition law.

Theorem

The set $\mathbb{Z}_n$ with addition is a group.

Proof.

Associativity of addition in $\mathbb{Z}_n$ follows from that in $\mathbb{Z}$:

$$([a]_n+[b]_n)+[c]_n = [a+b]_n+[c]_n = [(a+b)+c]_n = [a+(b+c)]_n \ldots$$

The element $[0]_n$ is the identity; the inverse of $[a]_n$ is $[-a]_n$. □

Commutative groups

Definition

The group G is *commutative* if, for all $g, h \in G$, $gh = hg$.

Familiar examples: $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{R}$ are commutative groups under the addition law.

Theorem

The set $\mathbb{Z}_n$ with addition is a group.

Proof.

Associativity of addition in $\mathbb{Z}_n$ follows from that in $\mathbb{Z}$:

$$([a]_n + [b]_n) + [c]_n = [a+b]_n + [c]_n = [(a+b)+c]_n = [a+(b+c)]_n \ldots$$

The element $[0]_n$ is the identity; the inverse of $[a]_n$ is $[-a]_n$. $\square$

More examples

The set $\mathbb{Q}^\times = \mathbb{Q} \setminus \{0\}$ has multiplicative inverses. So $m(a, b) = a \cdot b$ is a group law on $\mathbb{Q}^\times$.

Similarly for $\mathbb{R}^\times = \mathbb{R} \setminus \{0\}$.

The set $\mathbb{Z} \setminus \{0\}$ is **not a group** under multiplication; any element $a > 1$ has no multiplicative inverse in $\mathbb{Z} \setminus \{0\}$.

More examples

The set $\mathbb{Q}^\times = \mathbb{Q} \setminus \{0\}$ has multiplicative inverses. So $m(a, b) = a \cdot b$ is a group law on $\mathbb{Q}^\times$.
Similarly for $\mathbb{R}^\times = \mathbb{R} \setminus \{0\}$.
The set $\mathbb{Z} \setminus \{0\}$ is **not a group** under multiplication; any element $a > 1$ has no multiplicative inverse in $\mathbb{Z} \setminus \{0\}$.

Cyclic groups

For any $m \in \mathbb{N}$, $g \in G$, we write $g^m = g \cdot g \cdot g \cdots g$ (m times). We write $g^0 = e$, $g^{-m} = (g^m)^{-1}$.

Definition

A group G is *cyclic* if there is an element $g \in G$, called a *cyclic generator*, such that every $h \in G$ is of the form g^m for some $m \in \mathbb{Z}$.

Example

The additive group $\mathbb{Z}$ is cyclic; the elements 1 and -1 are both cycllic generators.

Example

The group $\mathbb{Z}_n$ is cyclic with generator $[1]_n$.

Cayley tables

The multiplication table for a group is called a *Cayley table*. Here is the Cayley table for a group with 4 elements.

	e	a	b	c
e	e	a	b	c
a	a	e	c	b
b	b	c	e	a
c	c	b	a	e

You can check that this group satisfies all three axioms. It is the simplest group that is not cyclic and is called the *Klein group*, written K_4.

Some Cayley tables for $\mathbb{Z}_2$ and $\mathbb{Z}_3$ (on the board).

Cayley tables

The multiplication table for a group is called a *Cayley table*. Here is the Cayley table for a group with 4 elements.

	e	a	b	c
e	e	a	b	c
a	a	e	c	b
b	b	c	e	a
c	c	b	a	e

You can check that this group satisfies all three axioms. It is the simplest group that is not cyclic and is called the *Klein group*, written K_4.

Some Cayley tables for $\mathbb{Z}_2$ and $\mathbb{Z}_3$ (on the board).

Cayley tables

The multiplication table for a group is called a *Cayley table*. Here is the Cayley table for a group with 4 elements.

	e	a	b	c
e	e	a	b	c
a	a	e	c	b
b	b	c	e	a
c	c	b	a	e

You can check that this group satisfies all three axioms. It is the simplest group that is not cyclic and is called the *Klein group*, written K_4.

Some Cayley tables for $\mathbb{Z}_2$ and $\mathbb{Z}_3$ (on the board).