WEB3CLUBS FOUNDATION LIMITED

Course Instructor: DR. Cyprian Omukhwaya Sakwa
PHONE: +254723584205   Email: cypriansakwa@gmail.com

Foundational Mathematics for Web3 Builders

Implemented in RUST

Lecture 46

August 26, 2024

Direct product of cyclic groups

- The direct product of cyclic groups is an essential concept in group theory, particularly in cryptography and other mathematical fields.
- In cryptography, the direct product of cyclic groups is used to create more complex groups for cryptographic protocols.
- For example, some cryptosystems employ product groups to integrate the security features of several smaller groups.
- The structure of the direct product of cyclic groups has the potential to impact the security of cryptographic algorithms.
- Understanding whether a product is cyclic or not can influence the design of secure protocols.

23.1 Properties of Direct Products of Cyclic Groups

1. If G_1 and G_2 are cyclic groups with orders n_1 and n_2, respectively, the order of the direct product $G_1 \times G_2$ is $n_1 \times n_2$.
2. Let G and H be cyclic groups with $|G| = n$ and $|H| = m$. Then $G \times H$ is cyclic if and only if m and n are relatively prime positive integers.

   If m and n are not relatively prime positive integers, the direct product $G \times H$ is not cyclic but still forms a group with interesting structural properties.
3. The direct product can be extended to more than two cyclic groups. If $G_1, G_2, \cdots, G_k$ are cyclic groups with orders $n_1, n_2, \cdots, n_k$ the direct product $G_1 \times G_2 \times \cdots \times G_k$ is cyclic if and only if the orders $n_1, n_2, \cdots, n_k$ are pairwise coprime.

Example 93

By determining a generator if it exists, show whether or not the following product groups are cyclic.

a) $\mathbb{Z}_2 \times \mathbb{Z}_3$ b) $\mathbb{Z}_2 \times \mathbb{Z}_2$

Solution

a) $\mathbb{Z}_2 \times \mathbb{Z}_3 = \{(0,0),\ (0,1),\ (0,2),\ (1,0),\ (1,1),\ (1,2)\}$. We check to see whether any of the six elements of $\mathbb{Z}_2 \times \mathbb{Z}_3$ generates it. Indeed $(0,0)$, $(0,1)$, $(0,2)$ and $(1,0)$ cannot generate $\mathbb{Z}_2 \times \mathbb{Z}_3$. We check the $(1,1)$ and $(1,2)$. Clearly,

$$(1,1)^1 = (1,1)$$

$$(1,1)^2 = (1,1)(1,1) = (0,2)$$

$$(1,1)^3 = (0,2)(1,1) = (1,0)$$

Solution (conti...)

$$(1,1)^4 = (1,0)(1,1) = (0,1)$$

$$(1,1)^5 = (0,1)(1,1) = (1,2)$$

$$(1,1)^6 = (1,2)(1,1) = (0,0).$$

Thus $\langle(1,1)\rangle = \{(0,0),\ (0,1),\ (0,2),\ (1,0),\ (1,1),\ (1,2)\} = \mathbb{Z}_2 \times \mathbb{Z}_3$. Therefore $\mathbb{Z}_2 \times \mathbb{Z}_3$ is cyclic since it is generated by $(1,1)$.

Does $(1,2)$ also generate $\mathbb{Z}_2 \times \mathbb{Z}_3$?
Note that the order of $1 \in \mathbb{Z}_2$ is 2 while the order of $2 \in \mathbb{Z}_3$ is 3 implying that $|(1,2)| = \text{lcm}(2,3) = 6$. So $(1,2)$ generates $\mathbb{Z}_2 \times \mathbb{Z}_3$.

Solution (conti...)

b) $\mathbb{Z}_2 \times \mathbb{Z}_2 = \{(0,0), (0,1), (1,0), (1,1)\}$. Note that $(0,0)$, $(0,1)$ and $(1,0)$ cannot generate $\mathbb{Z}_2 \times \mathbb{Z}_2$. Let us check $(1, 1.)$

$$(1,1)^1 = (1,1)$$

$$(1,1)^2 = (1,1)(1,1) = (0,0)$$

$$(1,1)^3 = (0,0)(1,1) = (1,1)$$

Thus $\langle(1,1)\rangle = \{(0,0),(1,1)\} \neq \mathbb{Z}_2 \times \mathbb{Z}_2$. Thus $(1,1)$ does not generate $\mathbb{Z}_2 \times \mathbb{Z}_2$ meaning that $\mathbb{Z}_2 \times \mathbb{Z}_2$ is not cyclic.

Notice that in the example above, $\mathbb{Z}_2 \times \mathbb{Z}_3$ was cyclic because the orders 2 and 3 are relatively prime. That is, gcd$(2,3) = 1$. However, $\mathbb{Z}_2 \times \mathbb{Z}_2$ was not cyclic because gcd$(2,2) = 2 \neq 1$. That is, we can simply apply $\gcd$ of orders of groups to determine if the direct product group is cyclic or not.

Example 94

Show that the group $\mathbb{Z}_5 \times \mathbb{Z}_7^*$ is cyclic.

Solution

$|\mathbb{Z}_5| = 5$ and $|\mathbb{Z}_7^*| = 6$.

Therefore, $|\mathbb{Z}_5 \times \mathbb{Z}_7^*| =$ lcm$(5,6) = 30$.
The generators of $\mathbb{Z}_5 = \{1,2,3,4\}$
The generators of $\mathbb{Z}_7^* = \{3,5\}$
So, the generators of $\mathbb{Z}_5 \times \mathbb{Z}_7^*$ are (1, 5), (1, 3), (2, 5), (2, 3), (3, 5), (3, 3), (4, 5), (4, 3)

Solution (conti...)

Let us take $(1, 5)$ for instance,

We have $30(1, 5) = \left(30 \cdot 1 \bmod 5, 5^{30} \bmod 7\right) = (0, 1)$ as required.

For fun, we find $\mathbb{Z}_5 \times \mathbb{Z}_7^* =$(0, 1) (0, 2),(0, 3),(0, 4),(0, 5),(0, 6),(1, 1),(1, 2),(1, 3),(1, 4),(1, 5),(1, 6),(2, 1),(2, 2),(2, 3),(2, 4),(2, 5),(2, 6),(3, 1), (3, 2),(3, 3),(3, 4),(3, 5),(3, 6),(4, 1),(4, 2),(4, 3),(4, 4), (4, 5),(4, 6).

Example 95

Show that $\mathbb{Z}_{11} \times \mathbb{Z}_{12}$ is cyclic.

Solution

Generators of $\mathbb{Z}_{11} = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$

Generators of $\mathbb{Z}_{12} = \{1, 5, 7, 11\}$

Thus, the generators of $\mathbb{Z}_{11} \times \mathbb{Z}_{12}$ are (1, 1), (1, 5), (1, 7), (1, 11), (2, 1), (2, 5), (2, 7), (2, 11), (3, 1), (3, 5), (3, 7), (3, 11), (4, 1), (4, 5), (4, 7), (4, 11), (5, 1), (5, 5), (5, 7), (5, 11), (6, 1), (6, 5), (6, 7), (6, 11), (7, 1), (7, 5), (7, 7), (7, 11), (8, 1), (8, 5), (8, 7), (8, 11), (9, 1), (9, 5), (9, 7), (9, 11), (10, 1), (10, 5), (10, 7), (10, 11).

Providing just one of these is enough.

Let us pick $(1, 1)$ for instance

The order of $\mathbb{Z}_{11} \times \mathbb{Z}_{12}$ is 132.

Thus, $132(1, 1) = (0, 0)$ as required.

- The following program contains a Rust implementation for finding generators of the direct product of two cyclic groups $\mathbb{Z}_n \times \mathbb{Z}_m$.
- The primary use is to identify pairs of elements that generate the entire group, which can be useful in various mathematical and cryptographic applications.
- A cyclic group $\mathbb{Z}_n$ is a group of integers modulo n, with the group operation being addition modulo n. $\mathbb{Z}_n - \{0, 1, 2, \ldots, n-1\}$
- The direct product $\mathbb{Z}_n \times \mathbb{Z}_m$ is the set of ordered pairs of elements from these two groups, with the group operation being component-wise addition.
- In this program, we define a structure 'ZnZm' to represent the direct product of two cyclic groups $\mathbb{Z}_n$ and $\mathbb{Z}_m$, and implement methods to find all generators of this product group.

```rust
struct ZnZm {
        n: u32,
        m: u32,
}

impl ZnZm {
        fn new(n: u32, m: u32) -> Self {
                ZnZm { n, m }
        }

        fn find_generators(&self) -> Vec<(u32, u32)> {
                let mut generators = Vec::new();

                for i in 0..self.n {
                        for j in 0..self.m {
                                if self.is_generator(i, j) {
                                        generators.push((i, j));
                                }
                        }
                }

                generators
        }

        fn is_generator(&self, a: u32, b: u32) -> bool {
                let mut visited = vec![vec![false; self.m as usize]; self.n as usize];
                let mut count = 0;

                let mut x = 0;
                let mut y = 0;

                loop {
```

0

(1,1)

$\mathbb{Z}_{11} \times \mathbb{Z}_{12}$

11×12

```
                        if visited[x as usize][y as usize] {
                                break;
                        }

                        visited[x as usize][y as usize] = true;
                        count += 1;

                        x = (x + a) % self.n;
                        y = (y + b) % self.m;
                }

                count == (self.n * self.m)
        }
}

fn main() {
        let group = ZnZm::new(11, 12);
        let generators = group.find_generators();

        println!("Generators␣of␣Z11␣x␣Z12:␣{:?}", generators);
}
```

0

2

1 2

(1, 2)

x y

(0,0)

Understanding the Rust code

1. We define a structure ‘ZnZm‘ to represent the direct product of two cyclic groups $\mathbb{Z}_n$ and $\mathbb{Z}_m$. The struct ZnZm contains two fields:

- n: u32: The order of the first cyclic group $\mathbb{Z}_n$.
- m: u32: The order of the second cyclic group $\mathbb{Z}_m$.

(2) Implementation of ZnZm: The implementation block for ZnZm contains three methods:

- Constructor: new $(n : u32, m : u32) - >$ Self
  - This function creates a new instance of the ZnZm struct with the provided values for n and m.
  - Example: ZnZm::new(11, 12) creates a direct product group $\mathbb{Z}_{11} \times \mathbb{Z}_{12}$.

Understanding the Rust code (conti...)

- Finding Generators: find_generators(&self) $- > \text{Vec} < (u32, u32) >$
  - This method finds and returns a vector of all generators of the group $\mathbb{Z}_n \times \mathbb{Z}_m$.
  - It iterates over all possible pairs (i, j) where i ranges from 0 to $n-1$ and j ranges from 0 to $m-1$.
  - For each pair (i, j), it checks whether this pair is a generator using the is_generator method.
- Checking Generators: is_generator(&self, $a : u32, b : u32) - > bool$
  - This method checks if the pair (a, b) is a generator of the group $\mathbb{Z}_n \times \mathbb{Z}_m$.

Understanding the Rust code (conti...)

- A generator must visit every element in the group exactly once before repeating.
- The method uses a loop to simulate the group operation starting from $(0, 0)$ and moving to the next element by adding a to the first component and b to the second component (modulo n and m, respectively).
- If the loop visits all $n \cdot m$ elements exactly once before repeating, the pair (a, b) is a generator.

(3) main Function

- The main function creates an instance of the ZnZm struct representing $\mathbb{Z}_n \times \mathbb{Z}_m$.
- It then calls find_generators to find all generators of this group and prints them out.