|  |
| --- |
|  |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** |

**ЛЕКЦИОННЫЕ МАТЕРИАЛЫ**

|  |  |
| --- | --- |
| **Б2.В,ДВ.3.1 «Криптографические методы защиты информации 11504»** | |
| *(наименование дисциплины (модуля) в соответствии с учебным планом)* | |
| Уровень | бакалавриат, специалитет |
|  | *(бакалавриат, магистратура, специалитет)* |
| Форма обучения | очная |
|  | *(очная, очно-заочная, заочная)* |
| Направление(-я)  подготовки | 10.05.02 «Информационная безопасность» |
|  | *(код(-ы) и наименование(-я))* |
|  |  |
| Институт | Кибербезопасности и цифровых технологий |
|  | *(полное и краткое наименование)* |
| Кафедра | Информационное противоборство (КБ-8) |
|  | *(полное и краткое наименование кафедры, реализующей дисциплину (модуль))* |
| Лектор | Доцент Дедов Олег Петрович |
|  | *(сокращенно – ученая степень, ученое звание; полностью – ФИО)* |

|  |  |  |
| --- | --- | --- |
| Используются в данной редакции с учебного года | 2021/2022 | |
|  | *(учебный год цифрами)* | |
| Проверено и согласовано «\_\_\_\_» \_\_\_\_\_\_\_\_20\_\_\_г. |  |  |
|  | *(подпись директора Института/Филиала с расшифровкой)* | |

Москва 2020 г.

|  |  |  |  |
| --- | --- | --- | --- |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **МИРЭА**  Кафедра КБ-8 "Информационное противоборство" | | | |
|  | | |  |
|  | | |  |
|  | | | |
| **ЛЕКЦИЯ № 3** | | | |
| по дисциплине: | | **Б2.В,ДВ.3.1 «Криптографические методы защиты информации»** | |
|  | | (шифр и наименование учебной дисциплины) | |
| по теме: | Модульная арифметика. Арифметические операции в Z7. Свойства оператора mod.Сложение и умножение в Z10.  Аддитивная и мультипликативная инверсия.  Алгоритм Евклида | | |
|  | (наименование темы лекции) | | |
|  | | |
|  | | |  |
| МИРЭА – 2021 г. | | | |

Тема лекции: Модульная арифметика. Основные определения.

Учебные и воспитательные цели:

1.Введение в модульную арифметику. Арифметические операции в Z7.

2.  Свойства оператора mod.Сложение и умножение в Z10.

3.Аддитивная и мультипликативная инверсия.

4.Алгоритм Евклида.

**Время:** 2 часа (90 мин.).

Литература:

а) Основная:

1. Рябко Б.Я.,Фионов А.Н. Криптография в современном мире.-М.: Горячая линия-Телеком, 2018.-300с.:ил.

2. Горбенко А.О., Основы информационной безопасности: введение в профессию.Учебное пособие, СПб: ИЦ «Интермедия», 2016.‒ 224 с.

3. Бутакова Н.Г., Федоров Н.В. Криптографические методы защиты информации. Учебное пособие, СПб: ИЦ «Интермедия», 2016. ‒ 312 с.

4.Хорев А.А., Защита информации от утечки по техническим каналам. Учебник. СПб: ИЦ «Интермедия», 2016. 920 с.

б) Дополнительная литература:

1. Зайцев А.П. и др. Технические средства и методы защиты информации. Уч. пособие. М.: Горячая линия – Телеком. 2009. – 615 с.

2. Романец Ю.В. и др. Защита информации в компьютерных системах и сетях. М.: Радио и связь. 1999. – 376 с.

3. Лозовецкий В.В. Информационная безопасность. М.: Изд. ИУИ. 2011. – 169 с.

Учебно-материальное обеспечение:

Наглядные пособия.

Технические средства обучения: проектор.

Приложения: рисунки, таблицы, слайды.

ПЛАН ЛЕКЦИИ:

**Введение** – до 5 мин.

**Основная часть** (учебные вопросы) – до 80 мин.

1.Модульная арифметика. Арифметические операции в Z7. -25 мин.

2.  Свойства оператора mod.Сложение и умножение в Z10. – 15 мин.

3.Аддитивная и мультипликативная инверсия. 15 мин.

4.Алгоритм Евклида. 25 мин.

Заключение – до 5 мин.

Введение – до 5 мин.

Методические рекомендации:

- показать актуальность темы;

- довести целевую установку через основные положения лекции;

- охарактеризовать место и значение данной темы в курсе;

- описать обстановку, в которой разрабатывалась теоретическая проблема и шла ее практическая реализация;

- дать обзор важнейших источников, монографий, литературы по теме;

- вскрыть особенности изучения студентами материала по рассматриваемой проблеме.

Основная часть – до 80 мин.

Введение.

**Модульная арифметика**

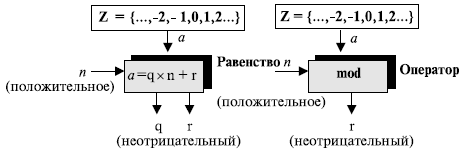
Уравнение деления (  ) имеет два входа ( a и n ) и два выхода ( q и r ). В модульной арифметике мы интересуемся только одним из выходов — остатком r. Мы не заботимся о частном q. Другими словами, когда мы делим a на n, мы интересуемся только тем, что *значение остатка равно* r. Это подразумевает, что мы можем представить изображение вышеупомянутого уравнения как *бинарный оператор* с двумя входами a и n и одним выходом r.

**Операции по модулю**

Вышеупомянутый *бинарный оператор* назван **оператором по модулю** и обозначается как mod. Второй вход ( n ) назван **модулем**. Вывод r назван **вычетом**. [Рисунок 2.9](#image.2.9) показывает отношение деления по сравнению с оператором по модулю.

**Операции по модулю**

Вышеупомянутый *бинарный оператор* назван **оператором по модулю** и обозначается как mod. Второй вход ( n ) назван **модулем**. Вывод r назван **вычетом**. [Рисунок 2.9](#image.2.9) показывает отношение деления по сравнению с оператором по модулю.



**Рис. 2.9.**Соотношение уравнения деления и оператора по модулю

Как показано на [рис. 2.9](#image.2.9), оператор по модулю ( mod ) выбирает целое число ( a ) из множества Z и положительный модуль ( n ). Оператор определяет неотрицательный остаток ( r ).

Мы можем сказать, что

a mod n = r

**Пример 2.14**

Найти результат следующих операций:

a. 27 mod 5

b. 36 mod 12

c. –18 mod 14

d. –7 mod 10

**Решение**

Мы ищем вычет r. Мы можем разделить a на n и найти q и r. Далее можно игнорировать q и сохранить r.

а. Разделим 27 на 5 - результат: r = 2. Это означает, что 27 mod 5 = 2.

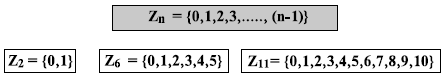
б. Разделим 36 на 12 — результат: r = 0. Это означает, что 36 mod 12 = 0.

в. Разделим (–18) на 14 — результат: r = –4. Однако мы должны прибавить модуль (14), чтобы сделать остаток неотрицательным. Мы имеем r = –4 + 14 = 10. Это означает, что –18 mod 14 = 10.

г. Разделим (–7) на 10 — результат: r = –7. После добавления модуля –7 мы имеем r = 3. Это означает, что –7 mod 10 = 3.

**Система вычетов: Zn**

**Результат операции по модулю n — всегда целое число между 0 и n - 1. Другими словами, результат a mod n — всегда неотрицательное целое число, меньшее, чем n.** Мы можем сказать, что операция по модулю создает набор, который в модульной арифметике можно понимать как **систему наименьших вычетов по модулю n**, или Zn. Однако мы должны помнить, что хотя существует только одно множество целых чисел ( Z ), мы имеем бесконечное число множеств вычетов ( Zn ), но лишь одно для каждого значения n. [Рисунок 2.10](#image.2.10) показывает множество Zn и три множества Z2, Z6 и Z11.



**Рис. 2.10.**Некоторые наборы Zn

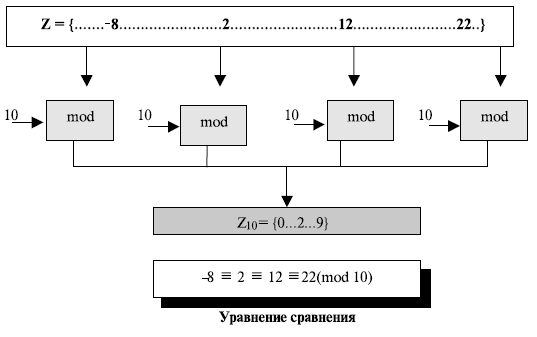
**Сравнения**

В криптографии мы часто используем понятие **сравнения** вместо равенства. Отображение Z в Zn не отображаются "один в один". Бесконечные элементы множества Z могут быть отображены одним элементом Zn. Например, результат 2 mod 10 = 2, 12 mod 10 = 2, 22 mod 10 = 2, и так далее. В модульной арифметике целые числа, подобные 2, 12, и 22, называются сравнимыми по модулю 10 (mod 10). Для того чтобы указать, что два целых числа сравнимы, мы используем **оператор сравнения** (  ). Мы добавляем mod n к правой стороне сравнения, чтобы определить значение модуля и сделать равенство правильным. Например, мы пишем:



[Рисунок 2.11](#image.2.11) показывает принцип сравнения. Мы должны объяснить несколько положений.

a. Оператор сравнения напоминает оператор равенства, но между ними есть различия. Первое: оператор *равенства* отображает элемент Z самого на себя; оператор *сравнения* отображает элемент Z на элемент Zn. Второе: оператор равенства показывает, что наборы слева и справа соответствуют друг другу "один в один", оператор сравнения — "многие — одному".



**Рис. 2.11.**Принцип сравнения

б. Обозначение ( mod n ), которое мы вставляем с правой стороны оператора сравнения, обозначает признак множества ( Zn ). Мы должны добавить это обозначение, чтобы показать, какой модуль используется в отображении. Символ, используемый здесь, не имеет того же самого значения, как бинарный оператор в уравнении деления. Другими словами, символ mod в выражении 12 mod 10 — оператор; а сочетание ( mod 10 ) в сравнении  означает, что набор — Z10.

**Модульная арифметика в Z 7.**

**Сложение a+b по модулю 7.**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **a\b** | **0** | **1** | **2** | **3** | **4** | **5** | **6** |
| **0** | **0** | **1** | **2** | **3** | **4** | **5** | **6** |
| **1** | **1** | **2** | **3** | **4** | **5** | **6** | **0** |
| **2** | **2** | **3** | **4** | **4** | **6** | **0** | **1** |
| **3** | **3** | **4** | **5** | **6** | **0** | **1** | **2** |
| **4** | **4** | **5** | **6** | **0** | **1** | **2** | **3** |
| **5** | **5** | **6** | **0** | **1** | **2** | **3** | **4** |
| **6** | **6** | **0** | **1** | **2** | **3** | **4** | **5** |

**Вычитание a-b по модулю 7.**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **a\b** | **0** | **1** | **2** | **3** | **4** | **5** | **6** |
| **0** | **0** | **6** | **5** | **4** | **3** | **2** | **1** |
| **1** | **1** | **0** | **6** | **5** | **4** | **3** | **2** |
| **2** | **2** | **1** | **0** | **6** | **5** | **4** | **3** |
| **3** | **3** | **2** | **1** | **0** | **6** | **5** | **4** |
| **4** | **4** | **3** | **2** | **1** | **0** | **6** | **5** |
| **5** | **5** | **4** | **3** | **2** | **1** | **0** | **6** |
| **6** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |

**Умножение a\*b по модулю 7.**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **a\b** | **0** | **1** | **2** | **3** | **4** | **5** | **6** |
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** |
| **1** | **0** | **1** | **2** | **3** | **4** | **5** | **6** |
| **2** | **0** | **2** | **4** | **6** | **1** | **3** | **5** |
| **3** | **0** | **3** | **6** | **2** | **5** | **1** | **4** |
| **4** | **0** | **4** | **1** | **5** | **2** | **6** | **3** |
| **5** | **0** | **5** | **3** | **1** | **6** | **4** | **2** |
| **6** | **0** | **6** | **5** | **4** | **3** | **2** | **1** |

**Умножение: 1(скорость) \*3(время)=3( номер позиции нулевой кабинки через 3 мин.) Колесо вращается со скоростью 1\7 оборота в минуту. При вращении со скоростью 5\7 оборота в минуту через 6 минут получим 30: 7= 4(полных оборота) и 2(остаток).**

**Умножение a\*b по модулю 8.**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **a\b** | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** |
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** |
| **1** | **0** | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| **2** | **0** | **2** | **4** | **6** | **0** | **2** | **4** | **6** |
| **3** | **0** | **3** | **6** | **1** | **4** | **7** | **2** | **5** |
| **4** | **0** | **4** | **0** | **4** | **0** | **4** | **0** | **4** |
| **5** | **0** | **5** | **2** | **7** | **4** | **1** | **6** | **3** |
| **6** | **0** | **6** | **4** | **2** | **0** | **6** | **4** | **2** |
| **7** | **0** | **7** | **6** | **5** | **4** | **3** | **2** | **1** |

**Деление a\b .**

**Деление a\b - равносильно умножению a на обратный элемент (обратное число) b, то есть a\b = a\*1\b.**

**Обратный элемент для 5= 1\5, для 6= 1\6 и т.д. Важно отметить, что при умножении числа например 5 на обратное число 1\5 получается 1(единица)**

**То есть 5\*1\5= 1 , 6\*1\6=1.**

**Таблица № 7.Умножение a\*b по модулю 7.(к вычислению обратных чисел).**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **a\b** | **0** | **1** | **2** | **3** | **4** | **5** | **6** |
| **0** | **0** | **0** | **0** | **0** | **0** | **0** | **0** |
| **1** | **0** | **1** | **2** | **3** | **4** | **5** | **6** |
| **2** | **0** | **2** | **4** | **6** | **1** | **3** | **5** |
| **3** | **0** | **3** | **6** | **2** | **5** | **1** | **4** |
| **4** | **0** | **4** | **1** | **5** | **2** | **6** | **3** |
| **5** | **0** | **5** | **3** | **1** | **6** | **4** | **2** |
| **6** | **0** | **6** | **5** | **4** | **3** | **2** | **1** |

**В таблице № 7 мы видим, что число 1 обратно 1 ; 2 обратно 4; 3 обратно 5; 4 обратно 2; 5 обратно 3; 6 обратна 6.**

**Таким образом поделить 3 на 5 это равносильно число 3 умножить на 3**

**3:5= 3\*3=9 , учитывая, что 9=7+2 = 2(mod7). То есть 3:5=3\*3=9= 2 (mod7).**

**Деление a:b по модулю 7.**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **a\b** | **0** | **1** | **2** | **3** | **4** | **5** | **6** |
| **0** | **-** | **0** | **0** | **0** | **0** | **0** | **0** |
| **1** | **-** | **1** | **4** | **5** | **2** | **3** | **6** |
| **2** | **-** | **2** | **1** | **3** | **2** | **6** | **5** |
| **3** | **-** | **3** | **5** | **1** | **6** | **2** | **4** |
| **4** | **-** | **4** | **2** | **6** | **1** | **5** | **3** |
| **5** | **-** | **5** | **6** | **4** | **3** | **1** | **2** |
| **6** | **-** | **6** | **3** | **2** | **5** | **4** | **1** |

**Возведение a в степень b по модулю 7.**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **a\b** | **1** | **2** | **3** | **4** | **5** | **6** |
| **1** | **1** | **1** | **1** | **1** | **1** | **1** |
| **2** | **2** | **4** | **1** | **2** | **4** | **1** |
| **3** | **3** | **2** | **6** | **4** | **5** | **1** |
| **4** | **4** | **2** | **1** | **4** | **2** | **1** |
| **5** | **5** | **4** | **6** | **2** | **3** | **1** |
| **6** | **6** | **1** | **6** | **1** | **6** | **1** |

**Остатки от деления на 7 шестой степени(1,2,3,4,5,6):**

**16**=1=0\*7 +1

2**6=64=9\*7 +1**

**36=729=104\*7+1**

**46= 4096= 585\*7+ 1**

**56=15625= 2232\*7 +1**

**66=46656= 6665\*7+1**[[1]](#endnote-2)

**Система вычетов**

*Система вычетов* [a], или [a]n, — множество целых чисел, сравнимых по модулю n. Другими словами, это набор всех целых чисел, таких, что x = a (mod n). Например, если n = 5, мы имеем множество из пяти элементов [0], [1], [2], [3] и [4], таких как это показано ниже:

[0] = {…., –15, -10, –5, 0, 5, 10, 15, …}

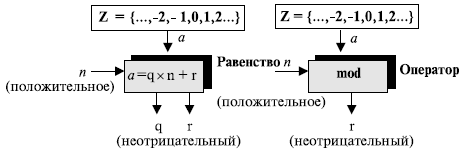
[1] = {…., –14, –9, –4, 1, 6 , 11, 16,…}

[2] = {…., –13, –8, –3, 2, 7, 12, 17,…}

[3] = {...., –12, –7, –2, 3, 8, 13, 18,…}

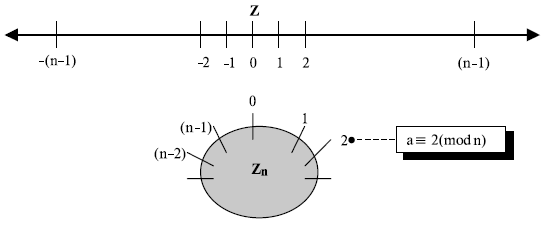
[4] = {…., –11, –6, –1, 4, 9, 14, 19,…}

Целые числа в наборе [0] все дают остаток 0 при делении на 5 (сравнимы по модулю 5 ). Целые числа в наборе [1] все дают остаток 1 при делении на 5 (сравнимы по модулю 5 ), и так далее. В каждом наборе есть один элемент, называемый наименьшим (неотрицательным) вычетом. В наборе [0] это элемент 0 ; в наборе [1] — 1, и так далее. Набор, который показывает все наименьшие вычеты: Z5 = {0, 1, 2, 3, 4}. Другими словами, набор Zn — набор всех **наименьших вычетов по модулю n**.



**Круговая система обозначений**

Понятие "сравнение" может быть лучше раскрыто при использовании круга в качестве модели. Так же, как мы применяем линию, чтобы показать распределение целых чисел в Z, мы можем использовать круг, чтобы показать распределение целых чисел в Zn.



**Рис. 2.12.**Сравнение использования диаграмм для Z и Zn

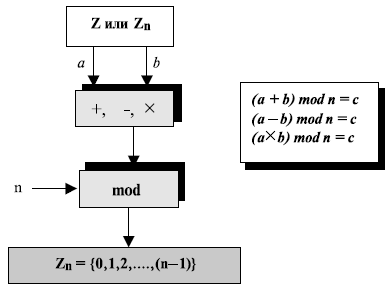
[Рисунок 2.12](#image.2.12) позволяет сравнить два этих подхода. Целые числа от 0 до n–1 расположены равномерно вокруг круга. Все целые числа, *сравнимые по модулю* n, занимают одни и те же точки в круге. Положительные и отрицательные целые числа от Z отображаются в круге одним и тем же способом, соблюдая симметрию между ними.

**Пример 2.15**

Мы пользуемся сравнением по модулю в нашей ежедневной жизни; например, мы применяем часы, чтобы измерить время. Наша система часов использует арифметику по модулю 12. Однако вместо 0 мы берем отсечку 12, так что наша система часов начинается с 0 (или 12 ) и идет до 11. Поскольку наши сутки длятся 24 часа, мы считаем по кругу два раза и обозначаем первое вращение как утро до полудня, а второе — как вечер после полудня.

**Операции в Zn**

Три *бинарных операции* ( *сложение, вычитание* и *умножение* ), которые мы обсуждали для Z, могут также быть определены для набора Zn. Результат, возможно, должен быть отображен в Zn с использованием операции по модулю, как это показано на [рис. 2.13](#image.2.13).



**Рис. 2.13.**Бинарные операции в Zn

Фактически применяются два набора операторов: первый набор — один из *бинарных операторов*  ; второй — операторы по модулю. Мы должны использовать круглые скобки, чтобы подчеркнуть порядок работ. Как показано на [рис. 2.13](#image.2.13), входы ( a и b ) могут быть членами Z или Zn.

**Пример 2.16**

Выполните следующие операторы (поступающие от Zn ):

а. Сложение 7 и 14 в Z15

б. Вычитание 11 из 7 в Z13

в. Умножение 11 на 7 в Z20

**Решение**

Ниже показаны два шага для каждой операции:

(14+7) mod 15 -> (21) mod 15 = 6

(7–11) mod 13 -> (-4) mod 13 = 9

(7x11) mod 20 -> (77) mod 20 = 17

**Пример 2.17**

Выполните следующие операции (поступающие от Zn ):

a. Сложение 17 и 27 в Z14

b. Вычитание 43 из 12 в Z13

c. Умножение 123 на -10 в Z19

**Решение**

Ниже показаны два шага для каждой операции:

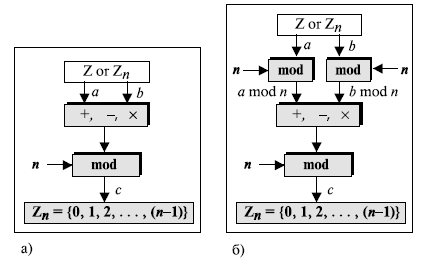
(17 + 27) mod 14 -> (44) mod 14 = 2

(12 – 43) mod 13 -> (–31) mod 13 = 8

((123) x (–10)) mod 19 -> (–1230) mod 19 = 5

**Свойства**

Мы уже упоминали, что два входа для трех *бинарных операторов* в сравнении по модулю могут использовать данные из Z или Zn. Следующие свойства позволяют нам сначала отображать два входа к Zn (если они прибывают от Z ) перед выполнением этих трех *бинарных операторов* . Заинтересованные читатели могут найти доказательства для этих свойств в приложении Q.



**Рис. 2.14.**Свойства оператора mod

**Первое свойство**: (a + b) mod n = [(a mod n) + (b mod n)] mod n

**Второесвойство**: (a – b) mod n = [(a mod n) - (b mod n)] mod n

**Третьесвойство**: (a x b) mod n = [(a mod n) x (b mod n)] mod n

[Рисунок 2.14](#image.2.14) показывает процесс до и после применения указанных выше свойств. Хотя по рисунку видно, что процесс с применением этих свойств более длинен, мы должны помнить, что в криптографии мы имеем дело с очень большими целыми числами. Например, если мы умножаем очень большое целое число на другое очень большое целое число, которое настолько большое, что не может быть записано в компьютере, то применение вышеупомянутых свойств позволяет уменьшить первые два операнда прежде, чем начать умножение. Другими словами, перечисленные свойства позволяют нам работать с меньшими числами. Этот факт станет понятнее при обсуждении экспоненциальных операций в последующих лекциях.

**Пример 2.18**

Следующие примеры показывают приложение вышеупомянутых свойств.

1. 
2. 
3. 

**Пример 2.19**

В арифметике мы часто должны находить остаток от степеней числа 10 при делении на целое число. Например, мы должны найти 10 mod 3, 102 mod 3, 103 mod 3, и так далее. Мы также должны найти 10 mod 7, 102 mod 7, 103 mod 7, и так далее. Третье свойство модульных операторов, упомянутое выше, делает жизнь намного проще.

10n mod x = (10 mod x)n Применение третьего свойства n раз.

Мыимеем

10 mod 3 = 1 -> 10n mod 3 = (10 mod 3)n = 1

10 mod 9 = 1 -> 10n mod 9 = (10 mod 9)n = 1

10 mod 7 = 3 -> 10n mod 7 = (10 mod 7)n = 3n mod 7

**Пример 2.20**

Мы уже говорили, что в арифметике остаток от целого числа, разделенного на 3, такой же, как остаток от деления суммы его десятичных цифр. Другими словами, остаток от деления 6371 равен остатку от деления суммы его цифр (17), на 3. Мы можем доказать, что это утверждение использует свойства модульного оператора. Запишем целое число как сумму его цифр, умноженных на степени 10.

a = an10n +………+ a1101 + a0100

Например: 6371 = 6 x 103 + 3 x 102+ 7 x 101+ 1 x 100

Теперь мы можем применить модульную операцию к двум сторонам равенства и использовать результат предыдущего примера, где остаток 10n mod 3 равен 1.

a mod 3 = (an x 10n +…+ a1 x 101+ a0 x 100) mod 3

= (an x 10n) mod 3 +…+ (a1 x 101) mod 3 + (a0x 100 mod 3) mod 3

= (an mod 3) x (10n mod 3) +…+ (a1 mod 3) x (101 mod 3) +

(a0 mod 3) x (100 mod 3) mod 3

= ((an mod 3) +…+ (a1 mod 3) + (a0 mod 3)) mod 3

= (an +…+ a1 + a0) mod 3

**Инверсии**

Когда мы работаем в модульной арифметике, нам часто нужно найти операцию, которая позволяет вычислить величину, обратную заданному числу. Мы обычно ищем **аддитивную инверсию** (оператор, обратный сложению) или **мультипликативную инверсию** (оператор, обратный умножению).

**Аддитивная инверсия**

В Zn два числа a и b **аддитивно инверсны** друг другу, если b = n – a. Например,



В Zn аддитивная инверсия числу a может быть вычислена как b = n – a. Например, аддитивная инверсия 4 в Z10 равна 10 – 4 = 6.

**В модульной арифметике каждое целое число имеет аддитивную инверсию. Сумма целого числа и его аддитивной инверсии сравнима с** 0 **по модулю**n .

Обратите внимание, что в модульной арифметике каждое число имеет аддитивную инверсию, и эта инверсия уникальна; каждое число имеет одну и только одну аддитивную инверсию. Однако инверсия числа может быть непосредственно тем же самым числом.

**Пример 2.21**

Найдите все взаимно обратные пары по сложению в Z10.

**Решение**

Даны шесть пар аддитивных инверсий — (0, 0), (1, 9), (2, 8), (3, 7), (4, 6) и (5, 5). В этом списке 0 — инверсия самому себе; так же и 5. Обратите внимание: аддитивные инверсии обратны друг другу; если 4 — аддитивная инверсия 6, тогда 6 — также аддитивная инверсия числу 4.

**Мультипликативная инверсия**

В Zn два числа a и b мультипликативно инверсны друг другу, если



Например, если модуль равен 10, то мультипликативная инверсия 3 есть 7. Другими словами, мы имеем .

**В модульной арифметике целое число может или не может иметь мультипликативную инверсию. Целое число и его мультипликативная инверсия сравнимы с**1 **по модулю**n .

Может быть доказано, что a имеет мультипликативную инверсию в Zn, если только НОД(n, a) = 1. В этом случае говорят, что a и n **взаимно простые**.

**Пример 2.22**

Найти мультипликативную инверсию 8 в Z10.

**Решение**

Мультипликативная инверсия не существует, потому что . Другими словами, мы не можем найти число между 0 и 9, такое, что при умножении на 8 результат сравним с 1 по mod 10.

**Пример 2.23**

Найти все мультипликативные инверсии в Z10.

**Решение**

Есть только три пары, удовлетворяющие условиям существования мультипликативной инверсии: (1, 1), (3, 7) и (9, 9). Числа 0, 2, 4, 5, 6 и 8 не имеют мультипликативной инверсии.

Мы можем проверить, что

(1 x 1) mod 10 = 1 (3 x 7) mod 10 = 1 (9 x 9) mod 10 = 1

**Пример 2.24**

Найти все мультипликативные обратные пары в Z11.

**Решение**

Мы имеем следующие пары: (1, 1), (2, 6), (3, 4), (5, 9), (7, 8) и (10, 10). При переходе от Z10 к Z11 число пар увеличивается. При Z11 НОД (11, a) = 1 (взаимно простые) для всех значений a, кроме 0. Это означает, что все целые числа от 1 до 10 имеют мультипликативные инверсии.

**Целое число**a **в** Zn **имеет мультипликативную инверсию тогда и только тогда, если**НОД (n, a) = 1(mod n)

Расширенный *алгоритм Евклида*, который мы обсуждали ранее в этой лекции, может найти мультипликативную инверсию b в Zn, когда даны n и b и инверсия существует. Для этого нам надо заменить первое целое число a на n (модуль). Далее мы можем утверждать, что алгоритм может найти s и t, такие, что . Однако если мультипликативная инверсия b существует, НОД (n, b) должен быть 1. Так что уравнение будет иметь вид

**(s x n) + (b x t) = 1**

Теперь мы применяем операции по модулю к обеим сторонам уравнения. Другими словами, мы отображаем каждую сторону к Zn. Тогда мы будем иметь

(s x n + b x t) mod n =1 mod n

[(s x n) mod n] + [(b x t) mod n] = 1 mod n

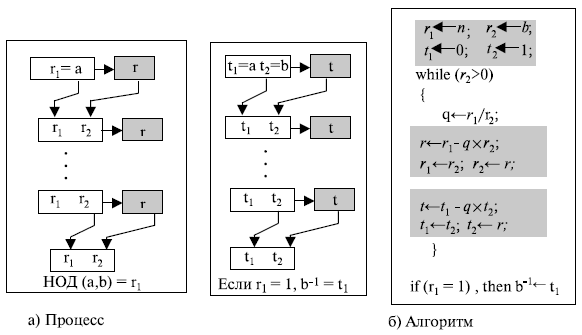
0 + [(b x t) mod n ] = 1

(b x t) mod n =1 -> Это означает, что t – это мультипликативная инверсия b в Zn

Обратите внимание, что  на третьей строке — 0, потому что, если мы делим , частное — s, а остаток — 0.

**Расширенный алгоритм Евклида находит мультипликативные инверсии**b **в** Zn **, когда даны**n **и** b **и** НОД (n, b) = 1 **. Мультипликативная инверсия**b **— это значение**t **, отображенное в**Zn .

[Рисунок 2.15](#image.2.15) показывает, как мы находим мультипликативную инверсию числа, используя расширенный *алгоритм Евклида*.



**Рис. 2.15.**Применение расширенного алгоритма Евклида для поиска мультипликативной инверсии

**Пример 2.25**

Найти мультипликативную инверсию 11 в Z26.

**Решение**

Мы используем таблицу, аналогичную одной из тех, которые мы уже применяли прежде при данных r1 = 26 и r2 = 11. Нас интересует только значение t.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **q** | **r1** | **r2** | **r** | **t1** | **t2** | **t** |
| 2 | 26 | 11 | 4 | 0 | 1 | -2 |
| 2 | 11 | 4 | 3 | 1 | -2 | 5 |
| 1 | 4 | 3 | 1 | -2 | 5 | -7 |
| 3 | 3 | 1 | 0 | 5 | -7 | 26 |
|  | 1 | 0 |  | -7 | 26 |  |

НОД (26, 11) = 1, что означает, что мультипликативная инверсия 11 существует. Расширенный *алгоритм Евклида* дает t1 = (–7).

Мультипликативная инверсия равна (–7) mod 26 = 19. Другими словами, 11 и 19 — мультипликативная инверсия в Z26. Мы можем видеть, что .

**Пример 2.26**

Найти мультипликативную инверсию 23 в Z100.

**Решение**

Мы используем таблицу, подобную той, которую применяли до этого при r1 = 100 и r2 = 23. Нас интересует только значение t.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **q** | **r1** | **r2** | **r** | **t1** | **t2** | **t** |
| 4 | 100 | 23 | 8 | 0 | 1 | -4 |
| 2 | 23 | 8 | 7 | 1 | -4 | 19 |
| 1 | 8 | 7 | 1 | -4 | 9 | -13 |
| 7 | 7 | 1 | 0 | 9 | -13 | 100 |
|  | 1 | 0 |  | -13 | 100 |  |

НОД (100, 23) = 1, что означает, что инверсия 23 существует. Расширенный Евклидов алгоритм дает t1 =-13. Инверсия — (–13) mod 100 = 87. Другими словами, 13 и 87 — мультипликативные инверсии в Z100. Мы можем видеть, что .

**Пример 2.27**

Найти мультипликативную инверсию 12 в Z26.

**Решение**

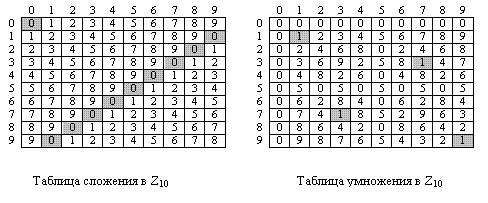
Мы используем таблицу, подобную той, которую мы применяли раньше при r1 = 26 и r2 = 12.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **q** | **r1** | **r2** | **r** | **t1** | **t2** | **t** |
| 2 | 26 | 12 | 2 | 0 | 1 |  |
| 6 | 12 | 2 | 0 | 1 | -2 |  |
|  | 2 | 0 |  | -2 | 13 |  |

, что означает отсутвствие для числа 12 мультипликативной инверсии в Z26

**Сложение и умножение таблиц**

[Рисунок 2.16](#image.2.16) показывает две таблицы для сложения и умножения. При сложении таблиц каждое целое число имеет аддитивную инверсию. Обратные пары могут быть найдены, если результат их сложения — ноль. Мы имеем (0, 0), (1, 9), (2, 8), (3, 7), (4, 6) и (5, 5). При умножении таблиц мы получаем только три мультипликативных пары (1, 1), (3, 7) и (9, 9). Пары могут быть найдены, когда результат умножения равен 1. Обе таблицы симметричны по диагонали, от левой вершины к нижней вершине справа. При этом можно обнаружить свойства коммутативности для сложения и умножения ( a+b = b+a и  ). Таблица сложения также показывает, что каждый ряд или колонка может поменяться с другим рядом или колонкой. Для таблицы умножения это неверно.



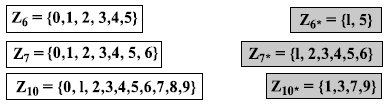
**Рис. 2.16.**Таблицы сложения и умножения для Z10

**Различные множества для сложения и умножения**

В криптографии мы часто работаем с инверсиями. Если отправитель посылает целое число (например, ключ для шифрования слова), приемник применяет инверсию этого целого числа (например, ключ декодирования). Если это действие (алгоритм шифрования/декодирования) является сложением, множество Zn может быть использовано как множество возможных ключей, потому что каждое целое число в этом множестве имеет аддитивную инверсию. С другой стороны, если действие (алгоритм шифрования/декодирования) — умножение, Zn не может быть множеством возможных ключей, потому что только некоторые члены этого множества имеют мультипликативную инверсию. Нам нужно другое множество, которое является подмножеством Zn и включает в себя только целые числа, и при этом в Zn они имеют уникальную мультипликативную инверсию. Это множество обозначается Zn\*. [Рисунок 2.17](#image.2.17) показывает некоторые случаи двух множеств. Обратите внимание, что множество Zn\*может быть получено из таблицы умножения типа показанной на [рис. 2.16](#image.2.16).

Каждый член Zn имеет аддитивную инверсию, но только некоторые члены имеют мультипликативную инверсию. Каждый член Zn\* имеет мультипликативную инверсию, но только некоторые члены множества имеют аддитивную инверсию.

**Мы должны использовать**Zn **, когда необходимы аддитивные инверсии; мы должны использовать**Zn\* **, когда необходимы мультипликативные инверсии**.



**Рис. 2.17.**Некоторые множества Zn и Zn\*

**Еще два множества**

Криптография часто использует еще два множества: Zp, и Zp\*. Модули в этих двух множествах — простые числа. Простые числа будут обсуждаться в следующих лекциях; пока можно сказать, что простое число имеет только два делителя: целое число 1 и само себя.

Множество Zp — то же самое, что и Zn, за исключением того, что n — *простое число*. Zp содержит все целые числа от 0 до p – 1. Каждый элемент в Zp имеет аддитивную инверсию; каждый элемент кроме 0 имеет мультипликативную инверсию.

Множество Zp\* — то же самое, что Zn\*, за исключением того, что Zp\* содержит все целые числа от 1 до p – 1. Каждый элемент в Zp имеет аддитивную и мультипликативную инверсии. Zp\* очень хороший кандидат, когда мы нуждаемся во множестве, которое поддерживает аддитивную и мультипликативную инверсии.

Ниже показаны два множества, когда p = 13.

Z13 = {0,1,2,3,4,5,6,7,8,9,10,11,12},

Z13\* = {1,2,3,4,5,6,7,8,9,10,11,12},

1. [↑](#endnote-ref-2)