Доп. дисциплины.

Магистры весна 2021.

Занятие 2

АЛГОРИТМЫ ТЕОРИИ ЧИСЕЛ

18.03.2021

ВЫЧИСЛЕНИЯ В КОЛЬЦЕ ВЫЧЕТОВ

Сравнения первой степени или линейные уравнения в кольце Z_n, где n – составное число. В теории чисел не принято писать как в алгебре:

Решим сравнение первой степени $ax \equiv b(\operatorname{mod} n)$.

Алгоритм решения сравнения 1 степени

1. Если d = нод(a,n), не делит “b”, то решений нет.
2. Если делит, и $a = a_0 d, b = b_0 d, n = n_0 d$, то мы решаем упрощенное

уравнение $a_0 x \equiv b_0(\operatorname{mod} n_0)$, которое обязательно имеет единственное решение x_0, т.к. нод(a_0,n_0) = 1 и по алгоритму Евклида можно найти обратный к a_0 по модулю n_0. Таким образом, $x_0 = a_0^{-1} \cdot b_0(\operatorname{mod} n_0)$.

Остается найти все решения по mod n. Этих решений ровно d штук, а именно, $x_i = x_0 + i \cdot n_0, i = 0,1,...,d-1$, поскольку при умножении на d добавка всегда будет кратна n.

Алгоритм решения системы линейных уравнений по разным модулям

Чтобы не разводить излишне много обозначений и не рассматривать слишком простой случай, возьмем три уравнения.

Шаг 1. Приведем уравнения системы в «человеческий вид», т.е. если они имели вид $ax \equiv b(\operatorname{mod} n)$, то находим d = нод(a,n), и если d не делит b, то уравнение не имеет решений, а значит и вся система не имеет решений. Поэтому предполагаем, что каждое уравнение имеет решение и все уравнения приведены к виду $x \equiv b(\operatorname{mod} n)$, технические нули в индексах, возникшие в процессе решения, мы опускаем.

Итак, система имеет вид $\begin{cases} x \equiv b_1 (\operatorname{mod} n_1) \\ x \equiv b_2 (\operatorname{mod} n_2) \\ x \equiv b_3 (\operatorname{mod} n_3) \end{cases}$, в общем случае уравнений может быть сколько угодно.

Шаг 2. Из первого уравнения следует, что

$$x = b_1 + n_1 \cdot t \qquad (1),$$

где t – произвольное целое число.

Подставляем это решение во второе уравнение и выделяем новую неизвестную t:

$$b_1 + n_1 t \equiv b_2 (\operatorname{mod} n_2) \Rightarrow n_1 t \equiv b_2 - b_1 (\operatorname{mod} n_2).$$

Шаг 3. Решаем получившееся уравнение и получаем $t \equiv b_2^{'} (\operatorname{mod} n_2^{'})$, штрихи поставлены, потому, что модуль n_2 мог измениться.

Из этого решения, как и на **Шаге 2** получаем

$$t = b_2^{'} + n_2^{'} \cdot s \qquad (2),$$

где s – произвольное целое число. Теперь подставляем (2) в (1):

$$x = b_1 + n_1 \cdot t = b_1 + n_1 \cdot (b_2^{'} + n_2^{'} \cdot s) = (b_1 + n_1 \cdot b_2^{'}) + (n_1 n_2^{'}) \cdot s = b_2^{''} + n_2^{''} \cdot s \qquad (3),$$

Шаг 4. Равенство (3) подставляем в третье уравнение системы и выделяем новую неизвестную s:

$$b_2^{''} + n_2^{''} \cdot s \equiv b_3 (\operatorname{mod} n_3) \Rightarrow n_2^{''} \cdot s \equiv b_3 - b_2^{''} (\operatorname{mod} n_3)$$

Решаем получившееся уравнение и получаем $s \equiv b_3^{'''} (\operatorname{mod} n_3^{'''})$,

Шаг 5. Выписываем ответ. Из последнего уравнения мы получаем, что

$$s = b_3^{'''} + n_3^{'''} \cdot r, \qquad (4)$$

где r – произвольное целое число, и, значит, подставляя в (3) получаем

$$x = b_2^{''} + n_2^{''} \cdot s = b_2^{''} + n_2^{''} \cdot (b_3^{'''} + n_3^{'''} \cdot r) = (b_2^{''} + n_2^{''} \cdot b_3^{'''}) + (n_2^{''} n_3^{'''}) \cdot r$$

Заменяя $b_2^{''} = b_1 + n_1 \cdot b_2^{'}, n_2^{''} = n_1 n_2^{'}$ из (3), окончательно получаем

$$x = (b_1 + n_1 \cdot b_2^{'} + n_1 n_2^{'} \cdot b_3^{'''}) + (n_1 n_2^{'} n_3^{'''}) \cdot r.$$

ОТВЕТ. Решением исходной системы будет число

$$x \equiv b_1 + n_1 \cdot b_2^{'} + n_1 n_2^{'} \cdot b_3^{'''} (\operatorname{mod} n_1 n_2^{'} n_3^{'''})$$

Выглядит довольно сурово. Но у вас будут конкретные числа «и страданья исчезнут куда-то, лишь склоняться над кроватью твоей люди в белых халатах». Слова из знаменитой когда-то песни. Лишь бы палата была не № 6.

РЕШЕНИЕ СИСТЕМ ЛИНЕЙНЫХ УРАВНЕНИЙ В ЦЕЛЫХ ЧИСЛАХ

Здесь есть два алгоритма, один «высоконаучный», когда решается сразу вся система. Но его «рабоче-крестьянский» вариант практичнее.

Ход 1. Мы решаем первое уравнение, и полученное решение подставляем во все остальные уравнения. Если исходно было n неизвестных и “k” уравнений, то после этого остается “k-1” уравнение от “n-1” неизвестной.

Ход 2. Мы выбираем второе уравнение, которое уже имеет “n-1” неизвестное и решаем его, решение подставляем во все оставшиеся и т.д.

На Ходе k. Мы или получаем полное решение или выясняем, что система решений не имеет. Одна сложность, как исходные неизвестные иксы, выразить через конечные, например, цеты.

Пусть нам дана система k линейных уравнений с целыми коэффициентами от n неизвестных, которую нам нужно решить в целых числах:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + ... + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + ... + a_{2n}x_n = b_2 \\ \dots\dots\dots\dots\dots\dots\dots\dots \\ a_{k1}x_1 + a_{k2}x_2 + ... + a_{kn}x_n = b_k \end{cases} \quad \textbf{(1),}$$

Запишем ее в матричном виде $AX = \bar{b}$, где A – матрица коэффициентов при неизвестных, X – столбец неизвестных, $\bar{b}$ - столбец свободных членов.

Алгоритм решения одного линейного уравнения в целых числах

Пусть первое уравнение имеет вид (индексы 1, обозначающие его номер, мы для краткости опустили)

$$a_1x_1 + a_2x_2 + ... + a_nx_n = b_1, \quad \textbf{(2)}$$

Пусть $A_1 = (a_1, a_2, ..., a_n)$ - матрица-строка коэффициентов первого уравнения, $X = (x_1, x_2, ..., x_n)^T$ - столбец неизвестных, тогда в матричном виде уравнение примет очень простой вид $A_1X = b_1$. Одна сложность, что никакой обратной матрицы для матрицы-строки A_1 не существует и «нарисовать» ответ $X = (A_1)^{-1}b_1$, мы не можем.

Алгоритм решения одного линейного уравнения в целых числах сильно напоминает алгоритм нахождения обратной матрицы методом Гаусса.

Разыскивая матрицу, обратную к матрице А, размера $n\times n$, мы дописываем справа единичную матрицу Е размера $n\times n$ и применяем **элементарные преобразования строк** к матрице $B=(A|E)$ размера $n\times 2n$.

После преобразований в левой части получается единичная матрица, тогда как справа появится матрица, обратная к матрице А: $(A|E)\Rightarrow(E|A^{-1})$.

В случае нашего алгоритма единичную матрицу Е размера $n\times n$, мы дописываем снизу от матрицы-строки A_1 и **элементарными преобразованиями столбцов** в первой строке все элементы обнуляем и оставляем один единственный ненулевой элемент «c_1», который, как легко можно будет понять, равен $c_1 = нод(a_1, a_2, ..., a_n)$.

Начинаем применение алгоритма.

Этап первый.

Шаг 1. Выписываем матрицу размера $(n+1)\times n$

$$\begin{pmatrix} A \\ E \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & ... & a_i & ... & a_j & ... & a_n \\ 1 & 0 & ... & 0 & ... & 0 & ... & 0 \\ 0 & 1 & ... & 0 & ... & 0 & ... & 0 \\ ... & ... & ... & ... & ... & ... & ... & ... \\ 0 & 0 & ... & 0 & ... & 0 & 1 & 0 \\ 0 & 0 & ... & 0 & ... & 0 & 0 & 1 \end{pmatrix}$$

Находим в первой строке **самое маленькое по модулю ненулевое число**, допустим это a_i. Пусть, a_j любое другое ненулевое число. Поделим его с остатком на первое и получим $a_j = q_j a_i + r_j, |r_j| < |a_i|$. Теперь умножим i-столбец на «$-q_j$» и прибавим его к j-столбцу. В первой строке j-столбца появится число r_j. Проделаем такую операцию со всеми ненулевыми элементами первой строки.

Шаг 2. Теперь, заменяя a_i, на самый маленький по модулю ненулевой остаток r_j снова выполняем **Шаг 1**.

Шаг 3. В итоге, во всей первой строке останется один единственный ненулевой элемент «c_1», находящийся на некотором месте с номером “s”. Так как при всех вычислениях мы применяли алгоритм Евклида, то этот $c_1 = нод(a_1, a_2, ..., a_n)$.

Шаг 4. Выписываем ответ. После всех преобразований мы получим матрицу

$$\begin{pmatrix} A_1 \\ E \end{pmatrix} \Rightarrow \begin{pmatrix} 0,0,...,0,c_1,0,...,0 \\ C_1 \end{pmatrix} = \begin{pmatrix} 0 & 0 & ... & c & ... & 0 & 0 \\ c_{1,1} & c_{1,2} & ... & c_{1,S} & ... & c_{1,n-1} & c_{1,n} \\ c_{2,1} & c_{2,2} & ... & c_{2,S} & ... & c_{2,n-1} & c_{2,n} \\ ... & ... & ... & ... & ... & ... & ... \\ c_{n-1,1} & c_{n-1,2} & ... & c_{n-1,S} & ... & c_{n-1,n-1} & c_{n-1,n} \\ c_{n,1} & c_{n,2} & ... & c_{n,S} & ... & c_{n,n-1} & c_{n,n} \end{pmatrix}$$

Производя, указанные выше преобразования мы, фактически, умножаем нашу матрицу $\begin{pmatrix} A_1 \\ E \end{pmatrix}$ справа, на некоторую матрицу C_1 размера $n \times n$. Мы следили только за тем, что происходит с первой строкой, т.е. за матрицей-строкой A_1. Но ведь умножается-то вся матрица $\begin{pmatrix} A_1 \\ E \end{pmatrix}$.

Так как снизу расположена единичная матрица, то она превратится после умножения на матрицу C_1, в эту самую матрицу C_1

Таким образом, после преобразований имеем:

$$\begin{pmatrix} A_1 \\ E \end{pmatrix} X = \begin{pmatrix} A_1 \\ E \end{pmatrix} \cdot C_1 \cdot C_1^{-1} \cdot X = \begin{pmatrix} 0,0,...,0,c_1,0,...,0 \\ C_1 \end{pmatrix} \cdot \left(C_1^{-1} \cdot X \right) = \begin{pmatrix} 0,0,...,0,c_1,0,...,0 \\ C_1 \end{pmatrix} \cdot Y$$

Итак, мы имеем новые переменные

$$Y = (y_1, y_2, ..., y_n)^T = C_1^{-1} \cdot X \Rightarrow X = C_1 \cdot Y . \qquad \textbf{(3)}$$

И в этих новых переменных наше исходное уравнение **(2)** выглядит очень просто

$$(0,0,...,0,c_1,0,...,0) \cdot \begin{pmatrix} y_1 \\ ... \\ y_s \\ ... \\ y_n \end{pmatrix} = b_1 \Rightarrow c_1 y_s = b_1 \Rightarrow y_s = \frac{b_1}{c_1}$$, и если c_1 не делит b_1,

то уравнение не имеет решений.

Теперь нужно выразить исходные иксы, через игреки. То есть найти решение уравнения **(2).** Ответ содержится в формуле **(3),** если учесть, что $y_s = \frac{b_1}{c_1}$.

В явном виде ответ дан формулой

$$X=\begin{pmatrix} c_{1,1} & c_{1,2} & \dots & c_{1,S} & \dots & c_{1,n-1} & c_{1,n} \\ c_{2,1} & c_{2,2} & \dots & c_{2,S} & \dots & c_{2,n-1} & c_{2,n} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ c_{n-1,1} & c_{n-1,2} & \dots & c_{n-1,S} & \dots & c_{n-1,n-1} & c_{n-1,n} \\ c_{n,1} & c_{n,2} & \dots & c_{n,S} & \dots & c_{n,n-1} & c_{n,n} \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ \dots \\ y_{s-1} \\ b_1/c_1 \\ y_{s+1} \\ \dots \\ y_n \end{pmatrix}. \qquad \textbf{(4)}$$

Это и есть решение уравнения **(2)** в целых числах, при этом новые переменные игреки, кроме s-того, который равен $y_s=\frac{b_1}{c_1}$, играют роль параметров, и могут принимать любые целые значения.

Или в развернутом виде:

$$x_i=c_{i,1}y_1+c_{i,2}y_2+....+c_{i,S-1}y_{S-1}+c_{i,S}\frac{b_1}{c_1}+c_{i,S+1}y_{S+1}+...c_{i,n}y_n, i=1,2,...,n..$$

Шаг 5. Полученное решение **(4)** подставляем во все уравнения системы **(1)**, приводим подобные относительно игреков и получаем систему из (k-1) одного уравнения от n-1 переменной $\{y_1,y_2,....,y_{S-1},y_{S+1},...,y_n\}$.

Но это долгий способ. Поступаем формально.

$$AX=\overline{b} \Rightarrow A\cdot(C_1Y)=\overline{b} \Rightarrow (A\cdot C_1)Y=\overline{b}.$$

То есть новая матрица системы **(1)** это $A'=A\cdot C_1$ и система имеет вид $A'Y=\overline{b}$.

Этап второй.

Итак, система имеет вид $A'Y=\overline{b}$.

Первым уравнением этой системы будет $c_1y_s=b_1$. Делая во всех уравнениях, начиная со второго, замену $y_s=\frac{b_1}{c_1}$, и перенося возникшие числовые значения в правую часть уравнений, а потом, отбрасывая первое уравнение, получаем систему из k-1 уравнения от n-1 неизвестных (игреков).

Пусть B - это матрица $A'=A\cdot C_1$, у которой вычеркнули первую строку и s-й столбец.

Столбец свободных членов изменился из-за переноса вправо числовых значений, возникших после замены $y_s=\frac{b_1}{c_1}$.

Пусть $Y^{(s)}=(y_1,...,y_{s-1},y_{s+1},...,y_n)^T$ - столбец оставшихся переменных,

$\overline{b}'=(b_2',...,b_s',b_{s+1}',...,b_n')^T$ - столбец новых свободных членов.

Тогда новая система из k-1 уравнения от n-1 неизвестной будет выглядеть так

$$\mathrm{B}\mathrm{Y}^{(s)} = \overline{\mathrm{b}}' \qquad \textbf{(I).}$$

Выбираем из этих k-1 уравнения первое и применяем к нему

Алгоритм решения одного линейного уравнения в целых числах, описанный на Этапе 1

Пусть это уравнение имеет вид (индексы 1, обозначающие его номер, мы для краткости опять опускаем)

$$b_1 y_1 + ... + b_{s-1} y_{s-1} + b_s y_{s+1} ... + b_{n-1} y_n = b_2^{'}, \qquad \textbf{(II)}$$

А в матричном виде $\mathrm{B}_1 \mathrm{Y}^{(s)} = \mathrm{b}_2^{'}$.

Под матрицей-строкой B_1 опять записываем единичную матрицу, на этот раз, размера $(n-1)\times(n-1)$.

Проводим все преобразования Этапа 1 и получаем итоговую матрицу

$\begin{pmatrix} B_1 \\ E \end{pmatrix} \Rightarrow \begin{pmatrix} 0, 0, ..., 0, c_2, 0, ..., 0 \\ C_2 \end{pmatrix}$, где элемент $c_2 = нод(b_1, b_2, ..., b_{n-1})$ находится на некотором месте t. Матрица C_2 (аналог матрицы C_1) имеет размер $(n-1)\times(n-1)$. Новые переменные назовем

$Z = (z_1, z_2, ..., z_{n-1})^T$. Так же как на Этапе 1 получаем $z_t = \frac{b_2^{'}}{c_2}$.

Естественно, если дробь $\frac{b_2^{'}}{c_2}$ - нецелое число, то уравнение **(II)**, а вместе с ним и вся система **(I)** решений не имеет.

Зависимость между переменными, имеет такой же вид, как и на Этапе 1

$$Y^{(s)} = C_2 Z \qquad \textbf{(III)}$$

Это и есть решение уравнения **(II)** в целых числах, при этом новые переменные (зеты), кроме t-го, который равен $z_t = \frac{b_2^{'}}{c_2}$, играют роль параметров, и могут принимать любые целые значения.

В новых переменных система **(I)** принимает вид $(\mathrm{B}C_2)Z = \overline{\mathrm{b}}' \Rightarrow \mathrm{B}^{'}Z = \overline{\mathrm{b}}'$,

Первым уравнением этой системы будет $c_2 z_t = b_2^1$. Делая во всех уравнениях, начиная со второго, замену $z_t = \frac{b_2^{'}}{c_2}$, и перенося возникшие числовые значения в правую часть уравнений, а потом, отбрасывая первое уравнение, получаем систему из k-2 уравнения от n-2 неизвестных (зетов).

Пусть С - это матрица $B' = B \cdot C_2$, у которой вычеркнули первую строку и t-й столбец.

Пусть $Z^{(t)} = (z_1, ..., z_{t-1}, z_{t+1}, ..., z_n)^T$ - столбец оставшихся переменных,

$\bar{b}'' = (b_3'', ..., b_t'', b_{t+1}'', ..., b_n'')^T$ - столбец новых свободных членов.

Тогда новая система из k-2 уравнений от n-2 неизвестных будет выглядеть так

$$CZ^{(t)} = \bar{b}'' \text{ и т.д.}$$

Осталось выразить n исходных переменных X, через n-2 новых $Z^{(t)}$. Основными здесь являются формулы **(4) и (III):**

$X = C_1 \cdot \begin{pmatrix} y_1 \\ ... \\ y_{s-1} \\ b_1/c_1 \\ y_{s+1} \\ ... \\ y_n \end{pmatrix}$, $Y^{(s)} = C_2 Z$**,** и еще нужно учесть, что $z_t = \frac{b_2'}{c_2}$. В итоге получаем:

$$X = C_1 \cdot \begin{pmatrix} y_1 \\ ... \\ y_{s-1} \\ b_1/c_1 \\ y_{s+1} \\ ... \\ y_n \end{pmatrix}, \qquad \begin{pmatrix} y_1 \\ ... \\ y_{s-1} \\ y_{s+1} \\ ... \\ y_n \end{pmatrix} = C_2 \cdot \begin{pmatrix} z_1 \\ ... \\ z_{t-1} \\ b_2'/c_2 \\ z_{t+1} \\ ... \\ z_n \end{pmatrix}$$

. Если обозначить через C_1^s и $C_1^{(s)}$, соответственно s-й столбец матрицы C_1 и саму матрицу C_1 с вычеркнутым s-м столбцом, то как непосредственно проверяется, мы получим:

$$X = C_1 \cdot \begin{pmatrix} y_1 \\ ... \\ y_{s-1} \\ b_1/c_1 \\ y_{s+1} \\ ... \\ y_n \end{pmatrix} = C_1^{(s)} \cdot \begin{pmatrix} y_1 \\ ... \\ y_{s-1} \\ y_{s+1} \\ ... \\ y_n \end{pmatrix} + \frac{b_1}{c_1} \cdot C_1^s = C_1^{(s)} \cdot C_2 \cdot \begin{pmatrix} z_1 \\ ... \\ z_{t-1} \\ b_2'/c_2 \\ z_{t+1} \\ ... \\ z_n \end{pmatrix} + \frac{b_1}{c_1} \cdot C_1^s .$$

Окончательный ответ, где только нужно перемножать матрицы, а не мучительно делать подстановки, раскладывать в сумму и приводить подобные такой:

$$X = (C_1^{(s)} \cdot C_2) \cdot \begin{pmatrix} z_1 \\ \ldots \\ z_{t-1} \\ b_2^{'} / c_2 \\ z_{t+1} \\ \ldots \\ z_n \end{pmatrix} + \frac{b_1}{c_1} \cdot C_1^s$$

Потом, на Этапе 3, у матрицы $C^{''} = C_1^{(s)} \cdot C_2$ нужно будет вычеркнуть t-столбец и умножить его на $\frac{b_2^{'}}{c_2}$ и т.д.