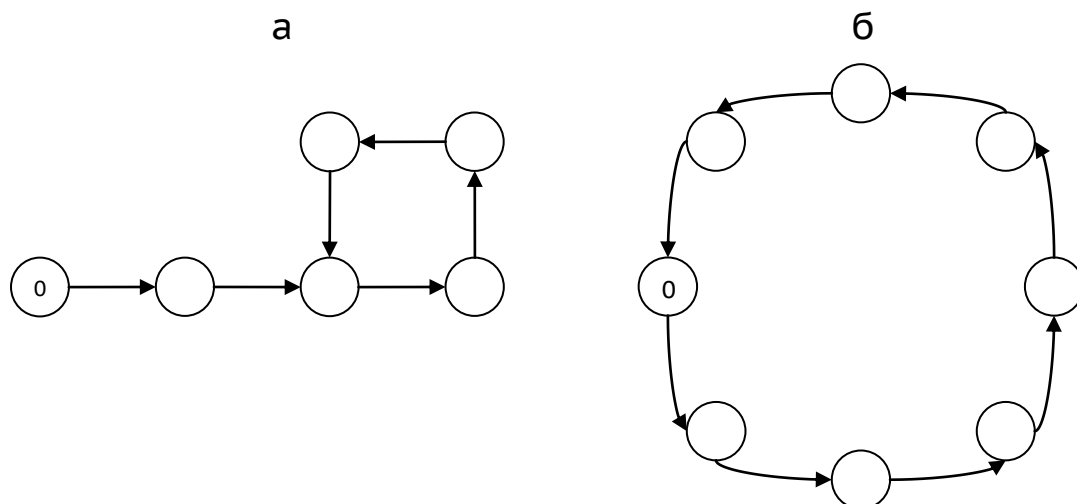


## Оглавление

1. Введение .....	2
2. Решение уравнений .....	4
2.1. Свойства конкатенации .....	4
2.2. Алгоритмы решения уравнений .....	5
3. Решение систем .....	7
3.1. Свойства возведения в степень .....	7
3.2. Решение уравнений вида $X * Y = A$ .....	7
3.3. Совмещение решений отдельных уравнений .....	8
4. Пример работы алгоритма .....	10
5. Направления дальнейшей работы .....	11

## 1. Введение

*Унарный конечный автомат* — конечный автомат, входной алфавит которого состоит из одного символа. Другими словами, в унарном автомате при переходе между состояниями не может быть ветвлений. Так как кол-во состояний конечно, «топологически» такой автомат может принимать только одну форму, кольца с «хвостом»:



**Рис. 1. Вид унарных конечных автоматов: а) кольцо с «хвостом»; б) вырожденная форма — простое кольцо. Нулем отмечено начальное состояние. Все стрелки означают переход по единственному символу. Какие-то состояния могут быть допускающими.**

Также, принимаемые унарным автоматом строки можно отождествить с их длинами, ведь все они состоят только из одного символа.

С кольцами без «хвостов» удобно работать в алгебраических терминах: рассматривать допускающие состояния как вычеты по модулю длины кольца  $N$ . Тогда множество принимаемых автоматом длин равняется множеству допускающих состояний – вычетов по модулю  $N$ . Далее в зависимости от контекста «кольцо» может означать конкретный автомат или кольцо вычетов.

Автомат будет обозначаться так:  $A = \{a_1, \dots, a_k\} \% N$ , где  $a_1, \dots, a_k$  — допускающие вычеты,  $N$  — модуль. Множество принимаемых длин обозначается  $L(A)$ ,

$$a \in L(A) \Leftrightarrow a \in \{a_1, \dots, a_k\}(\text{mod } N)$$

*Конкатенация 2 автоматов* — автомат, принимающий язык, слова которого могут быть получены конкатенацией слов из языков первого и второго исходных автоматов соответственно. Обозначение:  $C = A * B$ . При конкатенации унарных автоматов вместо «слов» и их «конкатенации» можно сразу рассматривать принимаемые длины и их суммы.

«Возведение» автомата в степень  $m$  — автомат, принимающий строки, полученные повторением принимаемых исходным автоматом строк  $m$  раз. Для колец:

$$A^m = \{ma_1, \dots, ma_k\} \% mN$$

**Цель этой курсовой работы** — изучение свойств операций конкатенации и возведения в степень применительно к унарным конечным автоматам и составление алгоритмов для приближенного решения уравнений вида  $X * A = B$  и систем вида

$$\begin{cases} X^{m_1} * Y^{m_2} = A \\ X^{m_3} * Y^{m_4} = B \end{cases}$$

где  $X, Y$  — неизвестные унарные конечные автоматы,  $A$  и  $B$  — кольца.

Решение таких систем (а также систем с большим количеством уравнений) может помочь оптимизировать работу виртуальной машины языка РЕФАЛ. (...)

Ввиду практического смысла (оптимизация, сокращение перебора) уравнений и систем над автоматами важно получить не точное решение, а автомат, язык которого *включает* язык точного решения. Такое послабление будет использоваться чтобы упростить алгоритмы и оставаться в множестве колец без «хвостов».

С другой стороны, нет смысла различать кольца, принимающие один язык. Поэтому далее на алгоритмическом уровне под «кольцами» (автоматами) понимаются классы эквивалентности (здесь и далее имеются ввиду классы эквивалентности по равенству языка):  $[A] = \{B | L(A) = L(B)\}$ , в т. ч. для колец:

$$A_{min} = \{a_1, \dots, a_k\} \% N, [A_{min}] = \{\{a_1 + i_1N, \dots, a_k + i_kN\} \% mN | m \in \mathbb{N}, 0 \leq i_* < m\}$$

На практике полезно сводить кольца к минимальным представителям их класса эквивалентности. Эта операция и результат называются *упрощением кольца*.

## 2. Решение уравнений

### 2.1. Свойства конкатенации

Операция конкатенации унарных автоматов коммутативна, потому что ее «база» — сложение целых чисел (длин) — коммутативная операция. Следовательно решения уравнений  $X * A = B$  и  $A * X = B$  совпадают.

Если  $A$  — унарный конечный автомат,  $A^*$  — обозначение для кольца, получаемого механическим «отбрасыванием» «хвоста», начальным состоянием становится точка входа в кольцо.  $A^*$  называется *собственным кольцом автомата  $A$* .

*Подкольцо кольца  $A$*  — кольцо, язык которого — подмножество языка кольца  $A$ .

*Свойство.* Результат конкатенации колец длин  $N_1$  и  $N_2$  — кольцо с «хвостом» длины не больше  $\text{lcm}(N_1, N_2)$  и модулем  $\text{gcd}(N_1, N_2)$ , его язык — подмножество языка его же собственного кольца.

*Доказательство.* Исходя из определения легко заметить, что конкатенация допускает «начальные» длины, полученные попарными сложениями базовых представителей вычетов исходных колец, которые могут произвольно удлиняться 2 модулями. Такое удлинение является подмножеством, а начиная с  $\text{lcm}(N_1, N_2)$  совпадает с удлинением одним модулем:  $\text{gcd}(N_1, N_2)$ :

$$\begin{aligned} A_1 &= \{a_1^1, \dots, a_1^{k_1}\} \% N_1, A_2 = \{a_2^1, \dots, a_2^{k_2}\} \% N_2, A_1 * A_2 = A_3 \Rightarrow \\ L(A_3) &= \{a_1^1 + a_2^1 + i_1^1 N_1 + j_1^1 N_2, \dots, a_1^{k_1} + a_2^{k_2} + i_1^{k_1} N_1 + j_1^{k_2} N_2, \dots, a_1^{k_1} + a_2^{k_2} + i_{k_1}^{k_1} N_1 + j_{k_1}^{k_2} N_2\} \subseteq \\ &\subseteq \{a_1^1 + a_2^1 + \beta_1^1 \text{gcd}(N_1, N_2), \dots, a_1^{k_1} + a_2^{k_2} + \beta_1^{k_2} \text{gcd}(N_1, N_2), \dots, a_1^{k_1} + a_2^{k_2} + \beta_{k_1}^{k_2} \text{gcd}(N_1, N_2)\} \Rightarrow \\ A_3^* &= \{a_1^1 + a_2^1, \dots, a_1^{k_1} + a_2^{k_2}\} \% \text{gcd}(N_1, N_2), \text{ где } i_*, j_*, \beta_* \in \mathbb{N} \cup \{0\}. \end{aligned}$$

*Следствие:* конкатенация колец с взаимнопростыми модулями ( $A$ ) — цепочка с петлей на конце, т. е.  $A^* = \{\bar{0}\} \% 1$ .

Если  $A = \{a_1, \dots, a_k\} \% mN, m \in \mathbb{N}$ , то кольцо  $\text{reduction}(A, N) := \{a_1, \dots, a_k\} \% N$  называется *сокращением  $A$  по модулю  $N$  (в  $m$  раз)*.  $A$  является подкольцом любого своего сокращения.

*Следствие.* Если  $(A * B)^* = C = \{c_1, \dots, c_k\} \% N$ , то на место  $A$  и  $B$  можно подставить любого представителя классов эквивалентности  $[A]_N := \{R \mid \text{reduction}(R, N) = \text{reduction}(A, N)\}$  и  $[B]_N$  соответственно без изменения результата.

Свойство показывает, что операция конкатенации и решение уравнения (следовательно и системы) могут выводить за пределы множества простых колец. Нахождение точных решений выходит за рамки этой курсовой работы. Далее под «решением» уравнения  $X * A = B$  понимается нахождение множества  $S = \{C^* \mid C * A = B\}$  — собственных колец точных решений. Это имеет смысл, потому что кольцевая структура стабильна:  $\forall C \subseteq S: (C * A)^* = B^*$ . Языки колец из  $S$  не включают языки точных решений, поэтому формально для вывода предназначены автоматы с «хвостом» некоторой длины, в котором каждое состояние — допускающее, и найденными кольцами.

## 2.2. Алгоритмы решения уравнений

*Утверждение.* Если у уравнения  $X * A = B$  есть решения, то это кольцо (максимальное решение) такой же длины, как и  $B$ , и, возможно, некоторые его подкольца.

*Доказательство.* Пусть  $B$  — минимальный представитель своего класса эквивалентности, длины  $N$ . Из свойства конкатенации следует: чтобы у уравнения были решения, длина кольца  $A$  и всех решений должна делиться на  $N$ . Пусть решением является кольцо  $C = \{c_1, \dots, c_k\} \% mN$ . Но тогда  $C' = \text{reduction}(C, N) = \{c_1, \dots, c_k\} \% N$  — тоже решение, а  $C$  — его подкольцо. Если  $C_1, \dots, C_p$  — решения модуля  $N$ , решением является и объединение их вычетов по модулю  $N$ , подкольца колец  $C_1, \dots, C_p$  и они сами также являются подкольцами объединения. Следовательно, любое решение является подкольцом единственного максимального решения.

*Следствие.* Отдельные вычеты по модулю  $N$ , рассматриваемые как кольца, при конкатенации с кольцом  $A$  могут давать подкольцо  $B$ . Если объединение языков таких подколец равно языку  $B$ , максимальное решение равно объединению исходных вычетов по модулю  $N$ .

Алгоритм нахождения максимального решения уравнения основан на следствии из утверждения о максимальном решении. На вход подается 2 кольца: левый или правый множитель и результат конкатенации. Выводится максимальное решение уравнения или сообщение о том, что решений нет.

```

1  MAXIMUM_SOLUTION ( $A = \{a_1, \dots, a_{k1}\} \% M, B = \{b_1, \dots, b_{k2}\} \% N$ ):
2  Упростить кольца  $A$  и  $B$  (обозначения  $a_*, b_*, N, M$  сохраняются для удобства)
3  if  $B = \{\bar{0}\} \% 1$  return  $X = \{\bar{0}\} \% 1$ 
4  if  $M$  не делит  $N$  Решения нет
5   $solution \leftarrow \emptyset$ 
6   $remainder \leftarrow \{b_1, \dots, b_{k2}\}$ 
7  for  $i \leftarrow 0..N - 1$ :
8      if  $\{a_1 + i, \dots, a_{k1} + i\} \subseteq \{b_1, \dots, b_{k2}\}(\text{mod } N)$ :
9           $solution \leftarrow solution \cup \{i\}$ 
10          $remainder \leftarrow remainder \setminus \{a_1 + i, \dots, a_{k1} + i\}(\text{mod } N)$ 
11 if  $remainder = \emptyset$  return  $X = solution \% N$ 
12 else Решения нет

```

Для решения систем помимо максимальных решений уравнений требуются условно «минимальные»: такие кольца модуля  $N$ , что ни одно из подколец, полученных исключением 1 вычета из списка допускающих, не является решением уравнения. Задача поиска минимальных решений не проще известной NP-сложной задачи о поиске минимального подпокрытия конечного множества.

```

1  MINIMUM_SOLUTIONS ( $A = \{a_1, \dots, a_{k1}\} \% M, B = \{b_1, \dots, b_{k2}\} \% N$ ):
2  Упростить кольца  $A$  и  $B$ 
3  if не существует  $maximum \leftarrow \text{MAXIMUM\_SOLUTION}(A, B)$ :
4      return  $\emptyset$ 

```

```

5  // maximum =  $\{m_1, \dots, m_l\} \% N$ 
6   $S \leftarrow \emptyset$ 
7   $U \leftarrow \{b_1, \dots, b_{k_2}\}$ 
8   $C \leftarrow \{\text{множество вычетов } (\{m_i\} \% N * A)^* \mid 1 \leq i \leq l\}$ 
9   $\text{cover}(s) := \bigcup_{m \in s} C_m$ 
10 for  $\text{subset} \subseteq \{m_1, \dots, m_l\}$ :
11     if  $\text{cover}(\text{subset}) = U$  and  $\forall \text{less} \subset \text{subset} \text{ cover}(\text{less}) \neq U$ :
12          $S \leftarrow S \cup \{\text{subset} \% N\}$ 
13 return  $S$ 

```

При реализации алгоритмов благодаря 2-му следствию из свойства конкатенации можно заменить кольцо  $A$  его сокращением по модулю  $N$  и работать не с автоматами, а напрямую с кольцом вычетов по модулю  $N$  и множествами на нем.

### 3. Решение систем

План решения систем вида  $\begin{cases} X^{m_1} * Y^{m_2} = A = \{a_1, \dots, a_{k_1}\} \% N_A \\ X^{m_3} * Y^{m_4} = B = \{b_1, \dots, b_{k_2}\} \% N_B \end{cases}$ :

1. Решение уравнений  $X'_1 * Y'_1 = A$  и  $X'_2 * Y'_2 = B$ , т. е. нахождение множеств  $S'_1$  и  $S'_2$ :  
 $\forall R \in S'_{1,2}$  существуют решения уравнений  $R * Y'_1 = A$  и  $R * Y'_2 = B$  соответственно.
2.  $S_1 = \{X: \text{reduction}(X^{m_1}, N_A) = R, \text{ если существует } R \in S'_1\}$ , аналогично для  $S_2$ .
3. «Совмещение» решений 1-го и 2-го уравнений (подробнее ниже).

#### 3.1. Свойства возведения в степень

Все степени  $m_i$  действуют аналогично, поэтому в этом разделе любое неизвестное кольцо на любой позиции будет обозначаться буквой  $X$ ,  $m$  — степень, в которую возводится  $X$  на этой позиции,  $N$  — длина соотв. конкатенации.

При решении каждого уравнения с 2 неизвестными разумно сразу ввести ограничение: надо, чтобы найденные кольца  $R$  длины  $N$  могли быть получены возведением некоторого кольца  $X$  в степень  $m$ :  $R = \text{reduction}(X^m, N)$ . Это ограничение задается подмножеством вычетов в  $\mathbb{Z}_N$ , которые могут присутствовать в  $R$ :

$$\text{allowed\_residues}(N, m) := \{i \mid \gcd(N, m) \mid i\}$$

В этом случае кольца  $X$  будут иметь модуль  $N / \gcd(N, m)$ . Возведение во взаимнопростую с модулем степень  $m$  приводит к «растяжению» и перестановке вычетов друг относительно друга.

Это словесное описание требуется оформить в виде вспомогательной функции, переводящей кольцо из «локальной» формы (решение уравнения) в «глобальную», общую для системы:

- 1  $\text{LOG}(R = \{r_1, \dots, r_k\} \% N, m)$ :
- 2  $M \leftarrow N / \gcd(N, m)$
- 3 **return**  $X = \{x_i: 0 \leq x_i < M, x_i * m = r_i(\text{mod } N) \mid 0 \leq i \leq k\}$
- 4 //  $x_i$  гарантированно существуют,
- 5 // если  $\{r_1, \dots, r_k\} \subseteq \text{allowed\_residues}(N, m)$

Изложенная выше схема следует из свойств кольца вычетов и его мультипликативной группы.

#### 3.2. Решение уравнений вида $X * Y = A$

Алгоритм реализует «спецификацию» из 1-го пункта в плане решения системы. Он опирается на утверждение о максимальном решении и его следствия.

- 1  $\text{UNBOUND\_SOLUTIONS}(A = \{a_1, \dots, a_{k_1}\} \% N_A, m_1, m_2)$ :
- 2  $\text{possible\_lefts} \leftarrow \emptyset$
- 3 **for**  $\text{left} \in \text{allowed\_residues}(N_A, m_1)$ :
- 4     **if** существует  $\text{right} \leftarrow \text{MAXIMUM\_SOLUTION}(\text{left} \% N_A, A)$
- 5     **and** множество вычетов  $\text{right} \subseteq \text{allowed\_residues}(N_A, m_2)$ :
- 6          $\text{possible\_lefts} \leftarrow \text{possible\_lefts} \cup \{\text{left} \% N_A\}$

7 **return** *possible\_lefts*

### 3.3. Совмещение решений отдельных уравнений

*Утверждение.* Если  $A = \{a_1, \dots, a_l\} \% pM$  и  $qM = kN$ , то сокращение можно «проносить» через возведение в степень, т. е.

$$\text{reduction}(A^q, N) = \text{reduction}(\text{reduction}(A, M)^q, N)$$

*Доказательство:*

$$\begin{aligned} \text{reduction}(A^q, N) &= \text{reduction}(\{qa_1, \dots, qa_l\} \% qpM, N) = \{qa_1, \dots, qa_l\} \% N = \\ &= \text{reduction}(\{qa_1, \dots, qa_l\} \% qM, N) = \text{reduction}((\{a_1, \dots, a_l\} \% M)^q, N) = \\ &= \text{reduction}(\text{reduction}(A, M)^q, N). \end{aligned}$$

Если  $A = \{a_1, \dots, a_k\} \% N$ , то кольцо  $mA = \{a_1 + i_1N, \dots, a_k + i_kN | 0 \leq i_* < m\} \% mN$  называется *удлинением кольца  $A$  в  $m$  раз*.

*Утверждение.* Если для

$$X_1 = \{\dots\} \% N_1, X_2 = \{\dots\} \% N_2 \exists X_0: \text{reduction}(X_0, N_1) = X_1, \text{reduction}(X_0, N_2) = X_2(*),$$

то множество

$$X_0 = \text{intersection}(X_1, X_2) :=$$

$$\{\text{мн. вычетов } (\text{lcm}(N_1, N_2)/N_2)X_1 \cap \text{мн. вычетов } (\text{lcm}(N_1, N_2)/N_1)X_2\} \% \text{lcm}(N_1, N_2).$$

Если же  $\text{lcm}(N_1, N_2) = 1$ , построенное так кольцо  $X_0$  удовлетворяет условиям (\*). Обе части утверждения — следствия свойств прямого произведения колец вычетов.

Понадобится функция, определяющая, совместимы ли 2 кольца.

```

1  COMPATIBLE ( $X_1 = \{\dots\} \% N_1, X_2 = \{\dots\} \% N_2$ ):
2  ins ← intersection( $X_1, X_2$ )
3  if reduction(ins,  $N_1$ ) =  $X_1$  and reduction(ins,  $N_2$ ) =  $X_2$ :
4      return true
5  else return false

```

Пусть уже получены и переведены в «глобальную» форму наборы решений  $S_1$  и  $S_2$  (см. 2-й пункт плана решения системы). Требуется найти такие пары

$$(X_1 = \{\dots\} \% N_1 \in S_1, X_2 = \{\dots\} \% N_2 \in S_2),$$

что

$$\begin{aligned} \exists X_0, Y_0, Y_1 = \{\dots\} \% N_3, Y_2 = \{\dots\} \% N_4: \text{reduction}(X_0, N_1) = X_1, \text{reduction}(X_0, N_2) = X_2, \\ X_1^{m_1} * Y_1^{m_2} = A, X_2^{m_3} * Y_2^{m_4} = B, \text{reduction}(Y_0, N_3) = Y_1, \text{reduction}(Y_0, N_4) = Y_2. \end{aligned}$$

Тогда пары  $(X_0, Y_0)$  будут решениями всей системы: это следствие утверждения о возведении в степень и сокращении.

Утверждение о пересечении колец дает простой способ проверить, совместимы ли левые решения. Если совместимы, нужно подобрать соотв. пару совместимых правых решений. По аналогии с утверждением о максимальном решении уравнения с одним неизвестным, будет существовать единственная «максимальная» пара, потому что если существует 2 совместимые пары, объединением вычетов колец из 1-го и 2-го уравнения получается также совместимая пара.



Но максимальная правая пара не обязательно состоит из максимальных решений «полуопределенных» уравнений (левые решения зафиксированы в пределах шага). Например, если  $X_1 = \{\dots\} \% N_1$ ,  $X_2 = \{\dots\} \% N_2$ ,  $Y_1 = \{\dots\} \% N_3$  и  $Y_2 = \{\dots\} \% N_4$  — максимальные решения уравнений  $X_1 * Y = A$  и  $X_2 * Y = B$  соответственно,  $X_1$  и  $X_2$  совместимы, а  $Y_1$  и  $Y_2$  — нет, т. е. мн. вычетов  $\text{reduction}(\text{intersection}(Y_1, Y_2), N_3) \subset \text{мн. вычетов } Y_1$ , или аналогично с  $Y_2$ , пока

мн. вычетов  $\text{reduction}(\text{intersection}(Y_1, Y_2), N_3) \supseteq \text{мн. вычетов } \forall R \in \text{MINIMUM\_SOLUTIONS}(X_1, A)$  можно принять  $\text{reduction}(\text{intersection}(Y_1, Y_2), N_3)$  за «новый»  $Y_1$  и попробовать совместить  $Y_1$  и  $Y_2$  снова.

Рассматривать подкольца минимальных правых решений смысла нет, потому что если не совместимы сокращения по модулям  $N_3$  и  $N_4$ , тем более не будут совместимы исходные кольца. Поэтому если на каком-то шаге  $Y_1$  или  $Y_2$  меньше любого минимального решения своего уравнения, этап можно завершить и совмещение  $X_1$  и  $X_2$  в ответ не включается.

Все готово, чтобы записать итоговый алгоритм решения систем.

```

1  SOLVE_SYSTEM ( $A = \{a_1, \dots, a_{k1}\} \% M$ ,  $B = \{b_1, \dots, b_{k2}\} \% N$ ,  $m1, m2, m3, m4$ ):
2   $solutions \leftarrow \emptyset$ 
3   $S_1 \leftarrow \text{map}(\text{UNBOUND\_SOLUTIONS}(A, m1, m2), \text{LOG}(*, m1))$ 
4   $S_2 \leftarrow \text{map}(\text{UNBOUND\_SOLUTIONS}(B, m3, m4), \text{LOG}(*, m3))$ 
5   $N_3 \leftarrow M / \text{gcd}(M, m2)$ 
6   $N_4 \leftarrow N / \text{gcd}(N, m4)$ 
7  for  $X_1 \in S_1$ ,  $X_2 \in S_2$ , if COMPATIBLE( $X_1, X_2$ ):
8       $Y_1 \leftarrow \text{LOG}(\text{MAXIMUM\_SOLUTION}(X_1, A), m2)$ 
9       $Y_2 \leftarrow \text{LOG}(\text{MAXIMUM\_SOLUTION}(X_2, B), m4)$ 
10      $M_1 \leftarrow \text{map}(\text{MINIMUM\_SOLUTIONS}(X_1, A), \text{LOG}(*, m2))$ 
11      $M_2 \leftarrow \text{map}(\text{MINIMUM\_SOLUTIONS}(X_2, B), \text{LOG}(*, m4))$ 
12     while not COMPATIBLE( $Y_1, Y_2$ ) and  $M_1 \neq \emptyset$  and  $M_2 \neq \emptyset$ :
13          $ins \leftarrow \text{intersection}(Y_1, Y_2)$ 
14         if мн. вычетов  $\text{reduction}(ins, N_3) \subset \text{мн. вычетов } Y_1$ :
15              $Y_1 \leftarrow \text{reduction}(ins, N_3)$ 
16         if мн. вычетов  $\text{reduction}(ins, N_4) \subset \text{мн. вычетов } Y_2$ :
17              $Y_2 \leftarrow \text{reduction}(ins, N_4)$ 
18          $\text{filter}(M_1, \text{мн. вычетов } min \in M_1 \subseteq \text{мн. вычетов } Y_1)$ 
19          $\text{filter}(M_2, \text{мн. вычетов } min \in M_2 \subseteq \text{мн. вычетов } Y_2)$ 
20     if COMPATIBLE( $Y_1, Y_2$ ):
21          $solutions \leftarrow solutions \cup \{(\text{intersection}(X_1, X_2), \text{intersection}(Y_1, Y_2))\}$ 
22 return  $solutions$ 

```

#### 4. Пример работы алгоритма

С помощью алгоритма можно решить систему

$$\begin{cases} X^1 * Y^1 = \{\bar{1}\} \% 3 \\ X^2 * Y^1 = \{\bar{5}\} \% 6 \end{cases} \quad (1)$$

В 1-м уравнении единичные степени идентичное действие функции LOG. Левым решением может быть любое кольцо модуля 3 с 1 вычетом:

$$S_1 = \{\{\bar{0}\} \% 3, \{\bar{1}\} \% 3, \{\bar{2}\} \% 3\}.$$

Во 2-м уравнении степень левого кольца сокращает кол-во возможных решений вдвое,  $\text{allowed\_residues}(6, 2) := \{\bar{0}, \bar{2}, \bar{4}\} \% 6$ , но на уровне системы множество  $S_2$  совпадает со множеством  $S_1$ .

В данном случае совпадают только равные левые решения. Если  $X_1 = X_2 = \{\bar{0}\} \% 3$ , то соответствующие правые максимальные (и единственные) решения совпадают с результатами конкатенации:  $Y_1 = \{\bar{1}\} \% 3, Y_2 = \{\bar{5}\} \% 6$ .

$$\begin{aligned} \text{intersection}(Y_1, Y_2) &= \{\text{мн. вычетов } 2Y_1 \cap \text{мн. вычетов } Y_2\} \% 6 = \\ &= \{\{\bar{1}, \bar{4}\} \cap \{\bar{5}\}\} \% 6 = \emptyset \% 6, \end{aligned}$$

поэтому правые решения не совпадают.

Если  $X_1 = X_2 = \{\bar{1}\} \% 3$ , то  $(X_2)^2 = \{\bar{2}\} \% 6$ ,  $\Rightarrow Y_1 = \{\bar{0}\} \% 3, Y_2 = \{\bar{3}\} \% 6$  — максимальные правые решения.

$\text{intersection}(Y_1, Y_2) = \{\{\bar{0}, \bar{3}\} \cap \{\bar{3}\}\} \% 6 = \{\bar{3}\} \% 6, \text{reduction}(\{\bar{3}\} \% 6, 3) = \{\bar{3}\} \% 3 = Y_1$ , «сокращение» кольца модуля 6 до 3 — пустое, таким образом пара  $(X = \{\bar{1}\} \% 3, \{\bar{3}\} \% 6)$  является решением системы.

Наконец, если  $X_1 = X_2 = \{\bar{2}\} \% 3$ , тогда  $(X_2)^2 = \{\bar{4}\} \% 6$ , соответственно  $Y_1 = \{\bar{2}\} \% 3, Y_2 = \{\bar{1}\} \% 6$  — не совпадающие правые решения, поэтому левого решения  $\{\bar{2}\} \% 3$  в ответе не будет.

Итог: «максимальным» (также пары, состоящие из любых подколец, но в контексте практического смысла решения подобных систем это не интересно) решением системы (1) является пара  $(X = \{\bar{1}\} \% 3, \{\bar{3}\} \% 6)$ . В конечном счете результатом будут 2 унарных автомата с «хвостами» длины 6, в которых каждое состояние — допускающее, и приведенными выше собственными кольцами.

## 5. Направления дальнейшей работы

Совмещение решений очень просто обобщается на случай множества (больше 2) уравнений вместе с функциями `intersection()` и `COMPARIBLE`. А пункты 1 и 2 в плане решения систем касаются каждого уравнения по отдельности, поэтому остаются независимыми. То есть, можно быстро доработать алгоритм для решения систем из многих уравнений.

«Приближение» ответов к точным решениям не кажется перспективным. Во-первых, точных решений систем в подавляющем большинстве случаев просто не существует, потому что в этом случае совпадение «хвостов» точных левых или правых решений (причем не в конечных алгебрах, а на  $\mathbb{Z}$ ) разных уравнений будет чистой случайностью.

Во-вторых, предполагаемые текущим алгоритмом «хвосты», состоящие из допускающих состояний, не имеют тенденцию к удлинению, конкретнее, по длине они равны максимальному хвосту среди предполагаемых решениями отдельных уравнений (см. параграф 2.1).

В-третьих, специфика практического применения алгоритма в большом количестве уравнений и множестве простых результатов конкатенации, вплоть до  $\{0\} \% 1$ . Учитывая предыдущий аргумент, в результате «лишних» принимаемых длин может быть пренебрежимо мало по сравнению с количеством строк «в основных» кольцах.

Нахождение менее тривиальных свойств конкатенации и возведения в степень применительно к кольцевой структуре может помочь существенно оптимизировать работу алгоритма, в т. ч. снизить максимальную асимптотическую сложность.

С одной стороны, при определенных входных данных (конкатенациях и степенях) некоторые блоки алгоритма могут гарантированно работать заранее известным образом. Например, в приведенном примере в строках 10 и 11 функции `SOLVE_SYSTEM` всегда возвращалось множество «минимальных» решений, состоящее из одного элемента — вычисленного буквально на предыдущем шаге максимального решения. Соответственно, «фильтрация» в строках 18 и 19, да и вообще весь цикл (строки 12–19) оказывались ненужными.

Кроме того, свойства кольцевой структуры, возможно, позволяют сократить полный перебор подмножеств в строке 3 функции `UNBOUND_SOLUTIONS` и возможных левых решений в строке 7 функции `SOLVE_SYSTEM`.