УДК 681.04

Ю.Д. Полисский, канд. техн. наук
Научно-исследовательский ин-т
автоматизации черной металлургии
(Украина, 49000, Днепр, ул. Шевченко, 59,
тел. (056) 7443365, e-mail: polissky477@gmail.com)

О некоторых подходах к выполнению проблемных операций в системе остаточных классов

Изложены новые подходы к выполнению проблемных операций в системе остаточных классов — модульному возведению в степень и дискретному логарифмированию, основанные на табличной реализации данных операций. Выборка данных осуществляется из таблиц, предварительно подготовленных для принятой системы модулей, с последующей обработкой выбранных данных по определенным алгоритмам.

К л ю ч е в ы е с л о в а: остаточные классы, проблемные операции, модули, дискретный логарифм.

Викладено нові підходи до виконання проблемних операцій в системі залишкових класів — модульного зведення в ступінь і дискретного логарифмування, базовані на табличній реалізації даних операцій. Вибірка даних здійснюється з таблиць, попередньо підготовлених для прийнятої системи модулів, з подальшою обробкою обраних даних за певними алгоритмами.

К л ю ч о в і с л о в а: залишкові класи, проблемні операції, модулі, дискретний логарифм.

Согласно результатам исследований при создании цифровых вычислительных структур эффективным является применение новых принципов на основе представления данных в системе остаточных классов (СОК) [1]. В СОК произвольное число N представляется в виде набора наименьших неотрицательных остатков по модулям $m_1, m_2, ..., m_n$, т.е. $N = (\alpha_1, \alpha_2, ..., \alpha_n)$, где $\alpha_i = N \pmod{m_i}$. При этом если числа m_i взаимно простые, то такому представлению соответствует только одно число N из диапазона $[0, M)$, где $M = m_1 m_2 ... m_n$.

Часто используемыми проблемными операциями являются модульное возведение в степень остатков целых чисел и нахождение остатков показателя степени, называемого дискретным логарифмом, по остаткам степени и остаткам основания. Основные алгоритмы реализации данных операций рассмотрены в работах [2—5]. Авторами [5] сделан следующий

вывод: «На основании приведенных методов решения проблемы дискретного логарифмирования нетрудно убедиться, что при решении этих задач используются громоздкие и трудоемкие операции над числами».

Рассмотрим новые подходы к реализации этих операций, основанные на табличной обработке данных, и дадим теоретическое обоснование алгоритмов табличной реализации модульного возведения в степень и дискретного логарифмирования.

Постановка задачи и ее решение. Пусть необходимо определить значение $\tilde{\alpha} = \alpha^r \pmod m$, где $\tilde{\alpha}, \alpha$ — остаток степени, остаток основания; r — остаток показателя степени; m — модуль; $0 \le \tilde{\alpha}, \alpha, r \le m-1$. Модульное возведение в степень представляет собой последовательное повторение операций умножения. В табл. 1 показано возведение в степень по модулю 11.

Рассмотрим циклическую подгруппу по возведению в степень, порожденную числом 7. Первый элемент — $7^0 = 1$; второй элемент — $7^1 = 7^0 * 7 = 7$; третий элемент — $7^2 = 7^1 * 7 = 5$; четвертый, пятый и шестой элементы — соответственно $7^3 = 7^2 * 7 = 2$, $7^4 = 7^3 * 7 = 3$ и $7^5 = 7^4 * 7 = 10$; седьмой, восьмой и девятый элементы — соответственно $7^6 = 7^5 * 7 = 4$, $7^7 = 7^6 * 7 = 6$ и $7^8 = 7^7 * 7 = 9$; десятый элемент — $7^9 = 7^8 * 7 = 8$; одиннадцатый элемент $7^{10} = 7^9 * 7 = 1$ равен первому. Условие цикличности $\alpha^r == km + 1$, где r, k — произвольные целые неотрицательные числа.

Пусть в общем случае $m = t\mu$, остаток по модулю m есть $\alpha = ts$, где μ и s — простые числа, $t > 1$ — произвольное целое неотрицательное число. При

Таблица 1

Остаток основания	Остаток степени при r										Длина цикла
	0	1	2	3	4	5	6	7	8	9	
0	0	0	0	0	0	0	0	0	0	0	1
1	1	1	1	1	1	1	1	1	1	1	1
2	1	2	4	8	5	10	9	7	3	6	10
3	1	3	9	5	4	1	3	9	5	4	5
4	1	4	5	9	3	1	4	5	9	3	5
5	1	5	3	4	9	1	5	3	4	9	10
6	1	6	3	7	9	10	5	8	4	2	10
7	1	7	5	2	3	10	4	6	9	8	10
8	1	8	9	6	4	10	3	2	5	7	10
9	1	9	4	3	5	1	9	4	3	5	5
10	1	10	1	10	1	10	1	10	1	10	2

этом испытуемое число и модуль могут быть как взаимно непростыми числами (т.е. $t > 1$), так и взаимно простыми (т.е. $t = 1$). Перепишем условие цикличности в виде $t(t^{r-1}s^r - k\mu) = 1$. Поскольку в выражении цикличности все числа целые, это условие может быть выполнено при $t = 1$ и $s^r - k\mu = 1$. Выражение $t = 1$ означает, что $m = \mu$, т.е. m — простое число. Следовательно, условие цикличности выполняется только для модулей, являющихся простыми числами.

Из табл. 1 видно, что длина L цикла различна для различных порождающих чисел — остатков оснований от $L_{\min} = 1$ до $L_{\max} = m - 1$. Работа с табл. 1 может быть сведена к простой выборке остатков степени по остаткам основания и показателя степени, если $L_{\max}$ кратна длине каждого из остальных циклов. Условие кратности $L_{\max}$ длине каждого из остальных циклов равносильно условию $(\alpha^{m-1})(\mathrm{mod}\, m) = 1$, т.е. $\alpha^{m-1} = km + 1$, где модуль m — простое число, k — произвольное целое неотрицательное число.

Утверждение. Если условие цикличности выполняется для остатка α основания, т.е. $(\alpha^{m-1})(\mathrm{mod}\, m) = 1$, то оно выполняется и для остатка $\tilde{\alpha} = \alpha + 1$ основания, т.е. $(\alpha + 1)^{m-1}(\mathrm{mod}\, m) = 1$.

Доказательство. Перепишем выражение $(\alpha^{m-1})(\mathrm{mod}\, m) = 1$ в виде $\dfrac{\alpha^m(\mathrm{mod}\, m)}{\alpha(\mathrm{mod}\, m)} = 1$. Пусть $\tilde{\alpha} = \alpha + 1$. Тогда

$$\frac{\tilde{\alpha}^m(\mathrm{mod}\, m)}{\tilde{\alpha}(\mathrm{mod}\, m)} = \frac{(\alpha+1)^m(\mathrm{mod}\, m)}{(\alpha+1)(\mathrm{mod}\, m)}.$$

Бином имеет вид

$$(\alpha+1)^m = \alpha^m + \sum_{s=1}^{m-1}\binom{m}{s}\alpha^{m-s} + 1.$$

Поскольку сумма $\sum_{s=1}^{m-1}\binom{m}{s}\alpha^{m-s}$ кратна модулю, $\left(\sum_{s=1}^{m-1}\binom{m}{s}\alpha^{m-s}\right)(\mathrm{mod}\, m) = 0$ и $(\alpha+1)^m(\mathrm{mod}\, m) = (\alpha^m + 1)(\mathrm{mod}\, m)$. Так как $\alpha^m = \alpha^{m-1}\alpha = (km+1)\alpha$, откуда $(\alpha^m)(\mathrm{mod}\, m) = \alpha$, получаем

$$(\alpha^m + 1)(\mathrm{mod}\, m) = (\alpha+1)(\mathrm{mod}\, m),\ (\alpha+1)^{m-1}(\mathrm{mod}\, m) = \frac{(\alpha+1)^m(\mathrm{mod}\, m)}{(\alpha+1)(\mathrm{mod}\, m)} = 1.$$

Поскольку для остатка $\alpha = 1$ основания это условие выполняется всегда, длина $L_{\max}$ кратна длине каждого из циклов, порожденных остатками основания по данному модулю.

Рассмотрим работу алгоритма в системе модулей $m_1 = 11$, $m_2 = 7$, $m_3 = 13$, $M = m_1 m_2 \ldots m_n = 1001$ на примере возведения числа $N_1 = 12 =$

$= (\alpha_1^1 = 1, \alpha_2^1 = 5, \alpha_3^1 = 12)$ в степень $R = 3 = (3, 3, 3)$ с получением числа $N_2 = (\alpha_1^2, \alpha_2^2, \alpha_3^2)$. В табл. 2 и 3 показано возведение в степень соответственно по модулям 7 и 13.

На первой итерации осуществляется преобразование представления показателя степени — числа $R = 3 = (3,3,3)$ из системы модулей $m_1 = 11$, $m_2 = 7$, $m_3 = 13$ в систему модулей $\hat{m}_1 = (m_1 - 1)$, $\hat{m}_2 = (m_2 - 1)$, ..., $\hat{m}_n = (m_n - 1)$, т.е. $\hat{m}_1 = 10$, $\hat{m}_2 = 6$, $\hat{m}_3 = 12$. В данном случае получаем $\hat{R} = 3 = (3,3,3)$. При этом от $M = m_1 m_2 ... m_n$, или, что то же самое, от M — наименьшего общего кратного модулей $m_1, m_2, ..., m_n$ переходим к $\hat{M} = \langle \hat{m}_1, \hat{m}_2, ..., \hat{m}_n \rangle$, где $\hat{M}$ — наименьшее общее кратное модулей $\hat{m}_1 = (m_1 - 1)$, $\hat{m}_2 = (m_2 - 1)$, ..., $\hat{m}_n = (m_n - 1)$.

На второй итерации по остатку $\alpha_1^1 = 1$ основания и остатку $\hat{r}_1 = 3$ показателя степени выбираем из табл. 1 значение остатка $\alpha_1^2 = 1$ степени, по остатку $\alpha_2^1 = 5$ основания и остатку $\hat{r}_2 = 3$ показателя степени выбираем из табл. 2 значение остатка $\alpha_2^2 = 6$ степени, по остатку $\alpha_3^1 = 12$ основания и остатку $\hat{r}_3 = 3$ показателя степени выбираем из табл. 3 значение остатка $\alpha_3^2 = 12$ степени. В исходной системе модулей $m_1 = 11$, $m_2 = 7$, $m_3 = 13$ получаем

$$N_2 = (\alpha_1^2 = 1, \alpha_2^2 = 6, \alpha_3^2 = 12) = 1728 - 1001 = 727.$$

Пусть по-прежнему $\tilde{\alpha} = \alpha^r \pmod{m}$, где r — остаток показателя степени, называемый дискретным логарифмом элемента $\tilde{\alpha}$ по основанию α; m — модуль; $0 \le \tilde{\alpha}, \alpha, r \le m - 1$. Задача дискретного логарифмирования состоит в отыскании по данным $\tilde{\alpha}$ и α всех дискретных логарифмов $r = r_{\min} + \mu\Delta$, $\mu = 0,1,2,...,\mu_{\max}$, в диапазоне $[0, M)$. Здесь $\mu_{\max}$ определяет максимальный не выходящий за пределы диапазона $[0, M)$ дискретный логарифм $r_{\max}$, Δ — величина шага перехода к соседнему дискретному логарифму. Таким образом, задача сводится к отысканию значений $r_{\max}$ и $\mu_{\max}$.

Таблица 2

Остаток основания	Остаток степени при r						Длина цикла
	0	1	2	3	4	5	
0	0	0	0	0	0	0	1
1	1	1	1	1	1	1	1
2	1	2	4	1	2	4	3
3	1	3	2	6	4	5	6
4	1	4	2	1	4	2	3
5	1	5	4	6	2	3	6
6	1	6	1	6	1	6	2

Выше была рассмотрена работа алгоритма модульного возведения в степень в системе модулей $m_1 = 11, m_2 = 7, m_3 = 13, M = m_1\ m_2 \ldots m_n = 1001$ на примере возведения числа $N_1 = 12 = (\alpha_1^1 = 1, \alpha_2^1 = 5, \alpha_3^1 = 12)$ в степень $R = 3 = (3,3,3)$ с получением результата $N_2 = (\alpha_1^2 = 1, \alpha_2^2 = 6, \alpha_3^2 = 12) = 727$. Рассмотрим алгоритм дискретного логарифмирования в той же системе модулей на примере определения дискретных логарифмов числа $N_2 = 727 = (\alpha_1^2 = 1, \alpha_2^2 = 6, \alpha_3^2 = 12)$ по основанию $N_1 = 12 = (\alpha_1^1 = 1, \alpha_2^1 = 5, \alpha_3^1 = 12)$.

По остатку $\alpha_1^1 = 1$ основания и остатку $\alpha_1^2 = 1$ степени выбираем из табл. 1 остатки показателя степени $\hat{r}_1 = \{0,1,2,3,4,5,6,7,8,9\}$, а по остатку $\alpha_1^1 = 1$ основания выбираем длину $L_1 = 1$ цикла. По остатку $\alpha_2^1 = 5$ основания и остатку $\alpha_2^2 = 6$ степени выбираем из табл. 2 остатки показателя степени $\hat{r}_2 = \{3\}$, а по остатку $\alpha_2^1 = 5$ основания — длину $L_2 = 6$ цикла. По остатку $\alpha_3^1 = 12$ основания и остатку $\alpha_3^2 = 12$ степени выбираем из табл. 3 остатки показателя степени $\hat{r}_3 = \{1,3,5,7,9,11\}$, а по остатку $\alpha_3^1 = 12$ основания — длину $L_3 = 2$ цикла.

Величина Δ шага перехода к соседнему дискретному логарифму $\Delta = 6 = \langle (L_1 = 1)^*(L_2 = 6)^*(L_3 = 2) \rangle$ есть наименьшее общее кратное длин L_1, L_2, L_3 циклов. Поскольку преобразование [6] представления показателя степени в систему модулей $m_1 = 11,\ m_2 = 7,\ m_3 = 13$ выполняется из четной

Таблица 3

Остаток основания	Остаток степени при r												Длина цикла
	0	1	2	3	4	5	6	7	8	9	10	11	
0	0	0	0	0	0	0	0	0	0	0	0	0	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	2	4	8	3	6	12	11	9	5	10	7	12
3	1	3	9	1	3	9	1	3	9	1	3	9	3
4	1	4	3	12	9	10	1	4	3	12	9	10	6
5	1	5	12	8	1	5	12	8	1	5	12	8	4
6	1	6	10	8	9	2	12	7	3	5	4	11	12
7	1	7	10	5	9	11	12	6	3	8	4	2	12
8	1	8	12	5	1	8	12	5	1	8	12	5	4
9	1	9	3	1	9	3	1	9	3	1	9	3	3
10	1	10	9	12	3	4	1	10	9	12	3	4	6
11	1	11	4	5	3	7	12	2	9	8	10	6	12
12	1	12	1	12	1	12	1	12	1	12	1	12	2

системы модулей $\hat{m}_1 = 10$, $\hat{m}_2 = 6$, $\hat{m}_3 = 12$, константы вычитания включают произведения четных модулей. Поэтому остатки показателя степени должны быть либо все четными, либо все нечетными, т.е. в данном случае $\hat{r}_1 = \{1,3,5,7,9\}$, $\hat{r}_2 = \{3\}$, $\hat{r}_3 = \{1,3,5,7,9,11\}$.

Наименьший из дискретных логарифмов $r_{\min}$ можно найти двумя способами. Первый способ заключается в следующем. Поскольку $0 \le r_{\min} \le \Delta$, наименьший дискретный логарифм может быть найден последовательным перебором чисел $0,1,...,\Delta$ одновременно в системах модулей $\hat{m}_1 = 10$, $\hat{m}_2 = 6$, $\hat{m}_3 = 12$ и $m_1 = 11$, $m_2 = 7$, $m_3 = 13$ с проверкой каждого из этих чисел в системе модулей $\hat{m}_1 = 10$, $\hat{m}_2 = 6$, $\hat{m}_3 = 12$ на соответствие табл. 1—3. Первое соответствие определяет $r_{\min} = 3 = (3,3,3)$. Способ обеспечивает хорошее быстродействие при сравнительно небольших значениях Δ.

Второй способ увеличения быстродействия при больших значениях Δ состоит в следующем. Из элементов полученных множеств остатков показателя степени $\hat{r}_1 = \{1,3,5,7,9\}$, $\hat{r}_2 = \{3\}$, $\hat{r}_3 = \{1,3,5,7,9,11\}$ в системе модулей $\hat{m}_1 = 10$, $\hat{m}_2 = 6$, $\hat{m}_3 = 12$ формируем в этой же системе произвольный набор остатков $\hat{r}_3 = \{1,3,5,7,9,11\}$, например $\tilde{r}_v = (\hat{r}_{1,1} = 1, \hat{r}_{2,1} = 3, \hat{r}_{3,3} = 5)$. Данный набор преобразуем из системы модулей $\hat{m}_1 = 10$, $\hat{m}_2 = 6$, $\hat{m}_3 = 12$ в систему модулей $m_1 = 11$, $m_2 = 7$, $m_3 = 13$. Если число $\tilde{r}_v$ находится в диапазоне $[0, \hat{M})$, такое преобразование возможно, и тогда получаем некоторый дискретный логарифм $\tilde{r}_v$ в системе модулей $m_1 = 11$, $m_2 = 7$, $m_3 = 13$.

Если число $\tilde{r}_v$ не находится в диапазоне $[0, \hat{M})$, то преобразование числа $\tilde{r}_v$ из системы модулей $\hat{m}_1 = 10$, $\hat{m}_2 = 6$, $\hat{m}_3 = 12$ в систему модулей $m_1 = 11$, $m_2 = 7$, $m_3 = 13$ невозможно. В этом случае из элементов множеств остатков показателя степени $\hat{r}_1 = \{1,3,5,7,9\}$, $\hat{r}_2 = \{3\}$, $\hat{r}_3 = \{1,3,5,7,9,11\}$ формируем новый набор остатков, например $\tilde{r}_v = (\hat{r}_{1,1} = 1, \hat{r}_{2,1} = 3, \hat{r}_{3,4} = 7)$. Описанная итерация повторяется до получения дискретного логарифма $\tilde{r}_v$ в системе модулей $m_1 = 11$, $m_2 = 7$, $m_3 = 13$. Поскольку значения Δ большие, число таких наборов остатков невелико. В частности, в рассматриваемом примере $\tilde{r}_v = (1,3,5) \to \tilde{r}_v = (1,3,7) \to \tilde{r}_v = (1,3,9)$. Поскольку число $\tilde{r}_v = (1,3,9)$ находится в диапазоне $[0, \hat{M})$, оно преобразуется в $r_v = (10,0,8)$ в системе модулей $m_1 = 11$, $m_2 = 7$, $m_3 = 13$.

Из значения $\tilde{r}_v$ вычитаем значение Δ, т.е. $r_{v-1} = r_v - \Delta = (10,0,8) - (4,1,2)$, и ожидаемую четность r_{v-1} сопоставляем с полученной. Несовпадение четностей свидетельствует о выходе дискретного логарифма за нижнюю границу диапазона $[0, \hat{M})$, следовательно, r_v — наименьший дискретный логарифм. В противном случае итерация выполняется для r_{v-2}: $r_{v-2} = r_{v-1} - \Delta$. Способ обеспечивает хорошее быстродействие при больших значениях Δ.

На основании изложенного представляется целесообразным осуществлять поиск наименьшего $r_{\min}$ одновременно двумя описанными способами. Первый полученный результат в рассмотренном примере: $\tilde{r}_{\min} = (3,3,3)$ в системе модулей $\hat{m}_1 = 10, \hat{m}_2 = 6, \hat{m}_3 = 12$ и $r_{\min} = (3,3,3)$ в системе модулей $m_1 = 11, m_2 = 7, m_3 = 13$.

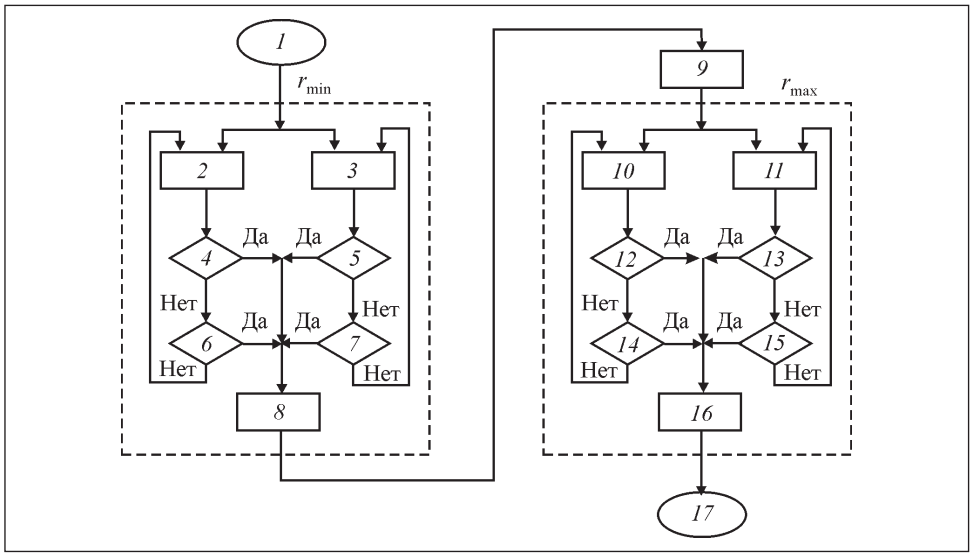
Для нахождения $\mu_{\max}$ понадобится величина $\tilde{r}_{\max}$ в диапазоне $[0, \hat{M})$, представленная в системе модулей $m_1 = 11, m_2 = 7, m_3 = 13$. Для этого определяем $\tilde{r}_{\max} = \tilde{r}_{\min} - \Delta$, переводим $\tilde{r}_{\max}$ в систему модулей $m_1 = 11, m_2 = 7, m_3 = 13$ и находим значение $\bar{r}_{\max}$. В данном примере $\tilde{r}_{\max} = \tilde{r}_{\min} - \Delta = (3,3,3) - (6,0,6) = (7,3,9) = 57$ и $\bar{r}_{\max} = 57 = (2,1,5)$ в системе модулей $m_1 = 11, m_2 = 7, m_3 = 13$. Поскольку система модулей задана, вычисляем предварительно $r_\sigma = \left(\left[\frac{M}{\hat{M}}\right] - 1\right)\hat{M}$, где $\left[\frac{M}{\hat{M}}\right]$ — целая часть от деления M на $\hat{M}$, и $r_u = r_\sigma + \bar{r}_{\max}$.

Аналогично двумя способами можно найти наибольший из дискретных логарифмов $r_{\max}$. Первым способом выполняется последовательный перебор чисел $M, M-1, \ldots, r_u$ одновременно в системах модулей $\hat{m}_1 = 10, \hat{m}_2 = 6, \hat{m}_3 = 12$ и $m_1 = 11, m_2 = 7, m_3 = 13$ с проверкой каждого из этих чисел в системе модулей $\hat{m}_1 = 10, \hat{m}_2 = 6, \hat{m}_3 = 12$ на соответствие табл. 1—3. Первое соответствие определяет $r_{\max}$. Способ обеспечивает хорошее быстродействие при сравнительно больших значениях r_u.

Второй способ увеличения быстродействия при небольших значениях r_u заключается в следующем. К значению r_u добавляем значение Δ, т.е. $r_{u+1} = r_u + \Delta = (7,5,8) + (4,1,2)$, и ожидаемую четность r_{u+1} сопоставляем с полученной. Несовпадение четностей свидетельствует о выходе дискретного логарифма за верхнюю границу диапазона $[0, M)$. Следовательно, r_{u+1} — наибольший дискретный логарифм. В противном случае итерация выполняется для r_{u+2}: $r_{u+2} = r_{u+1} + \Delta$. Способ обеспечивает хорошее быстродействие при малых значениях r_u.

Поскольку первый способ обеспечивает хорошее быстродействие при сравнительно больших значениях r_u, а второй способ — при малых значениях r_u, представляется целесообразным осуществлять поиск наибольшего дискретного логарифма $r_{\max}$ одновременно двумя способами. Первый полученный результат: $r_{\max} = 999 = (9,5,11)$ в системе модулей $m_1 = 11, m_2 = 7, m_3 = 13$.

Блок-схема алгоритма определения $r_{\min}$ и $r_{\max}$ представлена на рисунке. Определив $r_{\max}$ и $r_{\min}$, находим $\mu_{\max} = \frac{r_{\max} - r_{\min}}{\Delta}$. В рассматриваемом примере



Блок-схема алгоритма определения наименьшего r_{min} и наибольшего r_{max} дискретных логарифмов: *1* — начало; *2*, *3* — определение r_{min} первым и вторым способами; *4*, *7* и *12*, *15* — проверка, достигнут ли результат первым способом; *5*, *6* и *13*, *14* — проверка, достигнут ли результат вторым способом; *8* — сохранение результата в качестве r_{min}; *9* — подготовка промежуточных данных для определения r_{max}; *10*, *11*— определение r_{max} первым и вторым способами; *16* — сохранение результата в качестве r_{max}; *17* — конец

$$\mu_{max} = \frac{r_{max} - r_{min}}{\Delta} = \frac{999-3}{6} = \frac{(9,5,11)-(3,3,3)}{(6,6,6)} = \frac{(6,2,8)}{(6,6,6)} = (1,5,10) = 166.$$

В результате получаем

$$r = r_{min} + \mu\Delta,\ \ \mu = 0,1,2,\ ...,\ \mu_{max} = (3,3,3) + \mu\,(6,6,6),$$

$$\mu = (0,1,1),\ (1,1,1),\ (2,2,2),\ ...,\ (1,5,10).$$

Таким образом, предложенные подходы обеспечивают реализацию двух сложных операций — модульного возведения в степень и нахождения дискретного логарифма.

Выводы

Предложенные подходы к модульному возведению в степень и дискретному логарифмированию в системе остаточных классов основаны на выборке значений остатков из предварительно подготовленных таблиц с

последующей их обработкой по определенным алгоритмам. Применение данных подходов целесообразно для получения эффективных решений задач при выполнении проблемных операций.

СПИСОК ЛИТЕРАТУРЫ

1. *Акушский И.Я., Юдицкий Д.И.* Машинная арифметика в остаточных классах. М.: Сов. радио, 1968, 440 с.
2. *Василенко О.Н.* Теоретико-числовые алгоритмы в криптографии. М.: МЦНМО, 2003, 328 с.
3. *Рожков А.В., Ниссенбаум О.В.* Теоретико-числовые методы в криптографии: Учеб. пособие. Тюмень: Изд-во ТюмГУ, 2007, 156 с.
4. *Майзаков М.А.* Разработка модулей автоматической генерации заданий с решениями по теме «Дискретное логарифмирование»: Дипломная работа. Тюмень: Изд-во ТюмГУ, 2010, 54 с.
5. *Червяков Н.И., Бабенко М.Г., Кияшко Е.С., Шульженко К.С.* Решение проблемы дискретного логарифмирования с использованием системы остаточных классов. // http://hpc-education.unn.ru/files/conference_hpc/2013/files/62.pdf.
6. *Полисский Ю.Д.* О преобразовании представлений чисел в остатках из одной системы модулей в другую // Наука та прогрес транспорту. Вісн. Дніпропетровського національного університету залізничного транспорту імені акад. В. Лазаряна, 2016, № 63 (3), с. 130—137.

Поступила 20.02.17;
после доработки 15.05.17

REFERENCES

1. Akushskiy, I.Ya. and Yuditskiy, D.I. (1968), *Mashinnaya arifmetika v ostatochnykh klassakh* [Machine arithmetic in the residual classes], Sovetskoe radio, Moscow, USSR.
2. Vasilenko, O.N. (2003), *Teoretiko-chislovyie algoritmy v kriptografii* [Theoretical-numerical algorithms in cryptography], MTsNMO, Moscow, Russia.
3. Rozhkov, A.V. and Nissenbaum, O.V. (2007), *Teoretiko-chislovyie metody v kriptografii: Uchebnoe posobie* [Theoretical-numerical methods in cryptography: Manual], Izdatelstvo Tyumenskogo gosudarstvennogo universiteta, Tyumen, Russia.
4. Maizakov, M.A. (2010). “Development of modules for automatic generation of tasks with solutions concerning the subject Discrete Logarithm", Graduation Thesis, Tyumen State University, Tyumen, Russia.
5. Chervyakov, N.I., Babenko, M.G., Kiyashko, E.S. and Shulzhenko, K.S. (2013), “Solution of the problem of taking discrete logarithm with the use of the residue class system, available at: // http://hpc-education.unn.ru/files/conference_hpc/2013/files/62.pdf.
6. Polissky, Yu.D. (2016), “On the transformation of representation of numbers in residues from one modular system to another”, Nauka ta progress transportu, *Visnyk Dnipropetrovskogo natsionalnogo universitetu zaliznychnogo transportu imeni akad. V. Lazaryana*, Vol. 3, no. 63, pp. 130-137.

Received 20.02.17;
After revision 15.05.17

Yu.D. Polissky

ON CERTAIN APPROACHES TO IMPLEMENTATION
OF SOME PROBLEM OPERATIONS IN THE RESIDUE CLASS SYSTEM

New approaches to the implementation of the problem operations in the system of residue classes — modular exponentiation and discrete taking the logarithm — have been stated. The approaches are based on tabular implementation of these operations. In this case the data are sampled from the tables, pre-prepared for the taken system of modules, followed by processing of the selected data according to certain algorithms.

K e y w o r d s: residue classes, problem operations, modules, discrete logarithm.

ПОЛИССКИЙ Юрий Давидович, канд. техн. наук, ст. науч. сотр. Научно-исследовательского ин-та автоматизации черной металлургии (г. Днепр). В 1960 г. окончил Днепропетровский металлургический ин-т. Область научных исследований — системы и средства управления.