УДК 519.1

Э.И. ВАТУТИН

О ПЕРЕЧИСЛЕНИИ ЦИКЛИЧЕСКИХ ЛАТИНСКИХ КВАДРАТОВ И РАСЧЕТЕ ЗНАЧЕНИЯ ФУНКЦИИ ЭЙЛЕРА С ИХ ИСПОЛЬЗОВАНИЕМ

В статье приводится описание вычислительных экспериментов, направленных на подсчет числа циклических латинских (ЛК) и циклических диагональных латинских квадратов (ДЛК) порядка *N*. Полученные числовые ряды для нормализованных по первой строке циклических ЛК и ДЛК являются известными, они присутствуют в онлайн-энциклопедии целочисленных рядов (OEIS) под номерами A000010 и A123565, что позволило установить связь данного типа квадратов с другими известными объектами в области комбинаторики и теории чисел, в том числе с функцией Эйлера. Подсчет числа циклических ЛК и ДЛК общего вида позволил найти новые числовые ряды, которые были добавлены в OEIS под номерами A338522 и A338562. Циклические квадраты по определению порождаются соответствующими им циклическими перестановками. Указанное определение может быть расширено до перестановок, элементы которых образуют цикл длины *N*, а соответствующие им перестановочные квадраты имеют свойства, схожие с циклическими квадратами. С использованием нормализованных по первой строке циклических ЛК возможно разработать новый алгоритм вычисления функции Эйлера $\varphi(N)$ за время $t \simeq O(N^2)$ с затратами памяти $m \simeq O(1)$. Данный алгоритм при последовательной программной реализации проигрывает известным алгоритмам на базе факторизации аргумента функции Эйлера, однако он обладает хорошим потенциалом для распараллеливания и не включает в своем составе операций умножения и деления.

Ключевые слова: *комбинаторика, латинские квадраты, циклические латинские квадраты, числовые последовательности, OEIS, функция Эйлера.*

Введение

Одним из известных типов комбинаторных объектов являются латинские квадраты (ЛК), исследованию теоретических свойств и возможности практического применения которых посвящено большое количество научных работ [1, 2]. Латинским квадратом общего вида порядка *N* называется квадратная таблица размером $N \times N$ ячеек, заполненная элементами некоторого алфавита *U* мощности $|U| = N$ (для определенности, числами $0, 1, ..., N-1$) таким образом, что значения во всех строках и столбцах квадрата различны. Для диагональных латинских квадратов (ДЛК), являющихся подмножеством ЛК, вводится дополнительное ограничение на уникальность значений на главной и побочной диагоналях (другими словами, главная и побочная диагонали ДЛК являются трансверсалями). Нормализованными по первой строке называются квадраты, у которых первая строка зафиксирована (например, по возрастанию значений элементов) и не меняется в рамках соответствующего класса изоморфизма. Существенный фундаментальный интерес представляют квадраты специального вида, которые зачастую обладают интересными свойствами, не присущими квадратам общего вида (например, экстремальным числом трансверсалей, интеркалятов, ортогональных квадратов, мощностью клики из попарно-ортогональных квадратов с их участием и т.п.).

Одним из таких типов являются циклические квадраты [3, 4], в которых каждая $(i+1)$-я строка получается из *i*-й путем ее циклического сдвига на *d* позиций. Получаемый таким образом циклический квадрат именуется в некоторых источниках дважды циклическим ввиду того, что аналогичное свойство, присущее строкам, прослеживается и у его столбцов (см. пример на рис. 1). Циклические квадраты тесно связаны с пандиагональными квадратами, в которых, по определению, все ломаные диагонали и антидиагонали являются трансверсалями. При этом *i*-й ломаной диагональю называется множество ячеек квадрата $\left\{a_{i,(i+x) \bmod N}\right\}, x = \overline{0, N-1}$, соответствующая ломаная диагональ «параллельна» главной диагонали, а *i*-й ломаной антидиагональю – множество ячеек $\left\{a_{i,(N-i+x) \bmod N}\right\}, x = \overline{0, N-1}$, соответственно антидиагональ «параллельна» побочной диагонали (см. рис. 2). Более строго, все корректные циклические квадраты являются пандиагональными (данное утверждение, известное ранее, было проверено в ходе полного перебора всех циклических ЛК порядков $N \le 25$). Обратное утверждение в общем случае неверно.

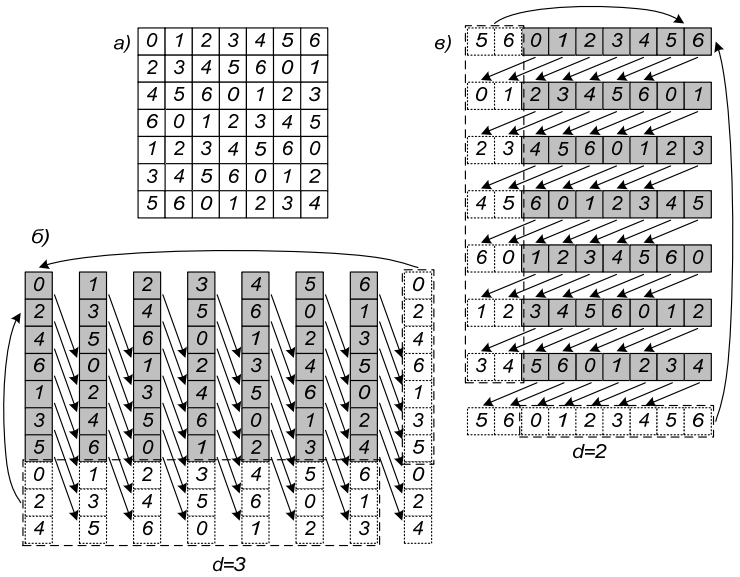

Рис. 1. Пример ДЛК (а), являющегося циклическим по вертикали – каждый столбец $C_{(i+1) \bmod N}$ получается путем циклического сдвига предыдущего столбца C_i на $d = 3$ позиции (б) и циклическим по горизонтали – каждая строка $R_{(i+1) \bmod N}$ получается из предыдущей строки R_i путем ее циклического сдвига на $d = 2$ позиции (в)

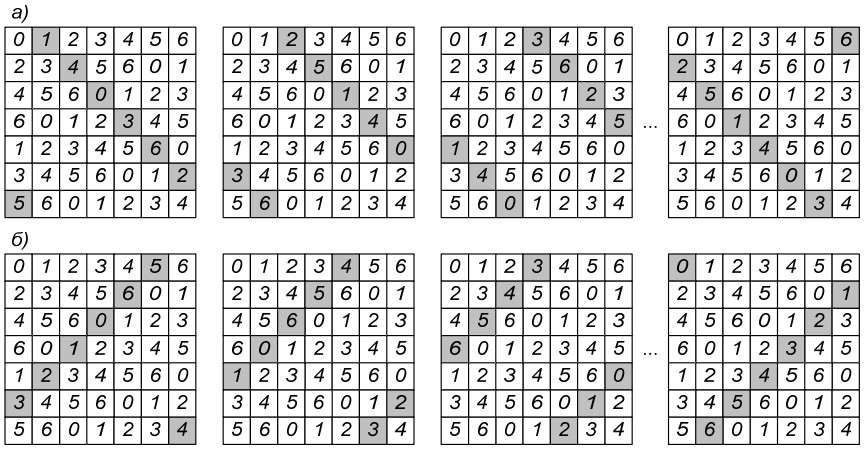

Рис. 2. Иллюстрация того, что все ломаные диагонали (а) и антидиагонали (б) циклического ДЛК (рис. 1, а) являются трансверсалями, т.е. он по определению является пандиагональным

Подсчет числа циклических ЛК и ДЛК в ходе вычислительного эксперимента

Не все циклические ЛК и ДЛК, получаемые путем циклического сдвига предыдущей строки на *d* позиций, являются корректными ввиду возможности появления дублирующихся значений в столбцах (нарушение определения ЛК) и на диагоналях (нарушение определения ДЛК). Данная особенность проиллюстрирована на рис. 3 для циклических квадратов порядка $N = 7$ и всех возможных значений $d = \overline{0, 6}$.

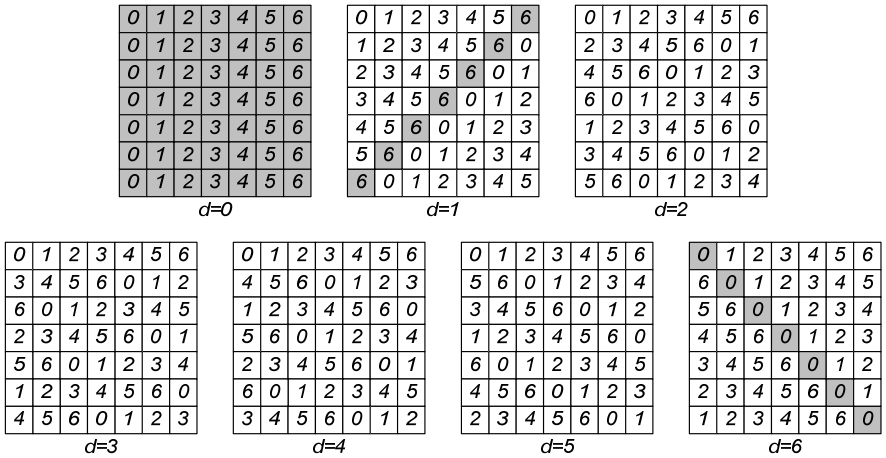

Рис. 3. Пример построения всех возможных циклических ЛК порядка $N = 7$: из полученного множества квадратов 6 являются корректными ЛК, 4 – корректными ДЛК, дубли значения элементов квадратов выделены серым

Для подсчета числа циклических ЛК и ДЛК заданного порядка N был организован вычислительный эксперимент, в ходе которого формировались все возможные циклические квадраты для всех $d = \overline{0, N-1}$, определялась их корректность с позиции соответствия определениям ЛК и ДЛК и производился подсчет полученного числа корректных квадратов каждого из типов. В результате были получены следующие числовые ряды:

1, 1, 2, 2, 4, 2, 6, 4, 6, 4, 10, 4, 12, 6, 8, 8, 16, 6, 18, 8, 12, 10, 22, 8, 20, …

и

1, 0, 0, 0, 2, 0, 4, 0, 0, 0, 8, 0, 10, 0, 0, 0, 14, 0, 16, 0, 0, 0, 20, 0, 10, …

Оба ряда являются известными, присутствуют в OEIS [5] под номерами A000010 и A123565 и напрямую не связаны с ЛК. Первый ряд представляет собой значение функции Эйлера (англ. Euler totient function) [6], второй – число таких положительных $k \le N$, что одновременно k, $k-1$ и $k+1$ являются взаимно простыми с N. Установленная связь с ЛК является нетривиальным эмпирическим следствием организованного вычислительного эксперимента, соответствующее описание числовых рядов в OEIS было расширено путем добавления информации о циклических ЛК и ДЛК.

Умножая приведенные выше значения на $N!$, можно получить соответственно общее число циклических ЛК и циклических ДЛК, соответствующие числовые ряды

1, 2, 12, 48, 480, 1440, 30240, 161280, 2177280, 14515200, 399168000, 1916006400, 74724249600, 523069747200, 10461394944000, 167382319104000, 5690998849536000, 38414242234368000, 2189611807358976000, …

и

1, 0, 0, 0, 240, 0, 20160, 0, 0, 0, 319334400, 0, 62270208000, 0, 0, 0, 4979623993344000, 0, 1946321606541312000, 0, 0, 0, …

На момент организации вычислительного эксперимента данные числовые ряды были неизвестны, они прошли апробацию и были добавлены в OEIS под номерами A338522 и A338562.

Обобщение определения циклических ЛК до перестановочных ЛК

Приведенное выше понятие циклических ЛК можно обобщить. По определению:

$$R_{(i+1) \bmod N} = P(R_i), \tag{1}$$

где P – некоторая циклическая перестановка. Например, для $d = 1$ $P = [1, 2, 3, 4, ..., N-1, 0]$, для $d = 2$ $P = [2, 3, 4, ..., N-1, 0, 1]$ и т.д. Несложно убедиться в том, что число подобных перестановок равно N и соответствующее число циклических квадратов не превосходит N (точнее, $N-1$, т.к. для $d = 0$ ЛК будет корректным только в вырожденном случае $N = 1$, для всех остальных размерностей будут присутствовать дублирования одних и тех же значений в столбцах, что иллюстрирует верхний левый квадрат на рис. 3). Можно найти и другие перестановки P, не являющиеся

циклическими и порождающие корректные ЛК и ДЛК для выбранного порядка *N*. Зависимости числа таких нормализованных ЛК и ДЛК, которые будем называть перестановочными (не путать со строчно-перестановочными ортогональными ДЛК, поиск которых производился в проекте RakeSearch [7]), получены в ходе вычислительного эксперимента и приведены ниже:

1, 1, 2, 6, 24, 120, 720, 5040, 40320, 362880, 3628800, 39916800, …

и

1, 0, 0, 0, 2, 0, 4, 0, 0, 0, 8, 0, …

Так из первого числового ряда следует, что для порядка *N* число перестановочных ЛК, нормализованных по первой строке, равно $(N-1)!$ (последовательность A000142 в OEIS), что является неожиданным следствием организованного вычислительного эксперимента. Второй числовой ряд представляет собой уже упомянутую выше последовательность A123565, число соответствующих нормализованных перестановочных ДЛК не отличается от числа нормализованных циклических ДЛК. Примеры нормализованных перестановочных ЛК порядка 6 и соответствующих им порождающих перестановок *P* приведены на рис. 4.

0	1	2	3	4	5
1	2	3	4	5	0
2	3	4	5	0	1
3	4	5	0	1	2
4	5	0	1	2	3
5	0	1	2	3	4

P = [1,2,3,4,5,0]

0	1	2	3	4	5
1	2	3	5	0	4
2	3	5	4	1	0
3	5	4	0	2	1
5	4	0	1	3	2
4	0	1	2	5	3

P = [1,2,3,5,0,4]

0	1	2	3	4	5
4	5	0	2	1	3
1	3	4	0	5	2
5	2	1	4	3	0
3	0	5	1	2	4
2	4	3	5	0	1

P = [4,5,0,2,1,3]

0	1	2	3	4	5
5	4	3	1	0	2
2	0	1	4	5	3
3	5	4	0	2	1
1	2	0	5	3	4
4	3	5	2	1	0

P = [5,4,3,1,0,2]

Рис. 4. Примеры нормализованных перестановочных ЛК порядка 6 и соответствующих им порождающих перестановок *P*

У всех перестановок *P*, порождающих соответствующие им перестановочные ЛК, структура мультимножества длин циклов одна и та же – $\{N\}$, т.е. все элементы перестановок входят в один цикл максимальной длины, что и определяет число таких перестановок, равное числу возможных циклов – $(N-1)!$.

Вычисление функции Эйлера с использованием циклических латинских квадратов

Выше было установлено, что число нормализованных по первой строке циклических ЛК порядка *N* равно значению функции Эйлера $\varphi(N)$. С учетом установленной особенности ее значение можно вычислять путем подсчета числа циклических ЛК, нормализованных по первой строке.

На практике для вычисления ее значения используют либо подсчет числа *k*, $1 \le k < N$, взаимно простых с *N* (расчет по определению), либо разложение *N* на простые сомножители по основной теореме арифметики

$$n = p_1^{\alpha_1} \cdot p_2^{\alpha_2} \cdot \ldots \cdot p_m^{\alpha_m},$$

где $p_1, p_2, ..., p_m$ – простые числа, возводимые в некоторую степень $\alpha_1, \alpha_2, ..., \alpha_m$, с последующим вычислением искомого значения функции Эйлера с использованием ее свойства мультипликативности:

$$\varphi(n) = n\left(1-\frac{1}{p_1}\right)\left(1-\frac{1}{p_2}\right)\cdot\ldots\cdot\left(1-\frac{1}{p_m}\right). \qquad (2)$$

Например, $6 = 2^1 \cdot 3^1$, поэтому $\varphi(6) = 6\left(1-\frac{1}{2}\right)\left(1-\frac{1}{3}\right) = 2$.

Первый способ наиболее эффективно реализуется с использованием алгоритма Евклида, второй требует факторизации числа *N*. При практической программной реализации первый способ оказывается медленнее, поэтому обычно на практике применяется второй. Оба способа требуют операций деления или остатка от деления, что не всегда удобно (особенно при реализации операций длинной арифметики или в виде специализированного вычислителя), алгоритм Евклида не распараллеливается. Указанных недостатков лишен способ вычисления через циклические ЛК, нормализованные по первой строке.

Простейшей практической реализацией метода определения функции Эйлера через циклические ЛК порядка N является следующая: для каждого смещения d, $0 < d < N$, формируется нормализованный по первой строке ЛК, который проверяется на корректность, число корректных ЛК представляет собой искомое значение функции Эйлера $\varphi(N)$. Некорректность может возникать в некоторых случаях из-за дублирования значений в столбцах формируемого циклического ЛК (см. рис.). Например, для $N = 6$ есть как корректные ЛК, так и нет, что проиллюстрировано на рис. 5.

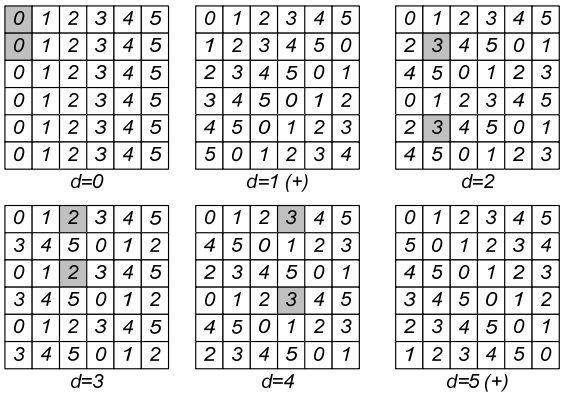

Рис. 5. Пример формирования нормализованных циклических ЛК порядка $N = 6$, число корректных ЛК совпадает со значением функции Эйлера $\varphi(6) = 2$; некоторые дубли значений в столбцах некорректных ЛК выделены серым; значения d, для которых получены корректные ЛК, отмечены символом «+»

Несложно показать, что корректные ЛК соответствуют только таким значениям d, которые являются взаимно простыми с N, т.е. значение функции Эйлера в результате подсчета нормализованных по первой строке циклических ЛК получается по определению.

Оценим вычислительную сложность данного алгоритма проверки. Проверить требуется $N-1$ смещение d. При каждой проверке на корректность формируемого ЛК производится определение $N-1$ строки ЛК (первая фиксирована и не меняется) по N элементов каждая, сама проверка реализуется путем заполнения одномерного булева массива с уже использованными в столбце значениями за время, не зависящее от N. Таким образом, временная сложность алгоритма составляет величину порядка $t \simeq O(N^3)$. На хранение формируемого ЛК требуется N^2 ячеек памяти, на хранение вектора использованных в столбце булевых значений – N ячеек, для N столбцов – суммарно N^2 ячеек, т.е. емкостная сложность алгоритма составляет $m \simeq O(N^2)$.

Алгоритм допускает ряд оптимизаций. Прежде всего, вместо хранения всех строк формируемого ЛК можно хранить только текущую. При этом множества использованных в столбцах значений элементов хранить все равно необходимо, затраты памяти снижаются с $\underbrace{N^2}_{\substack{\text{строки}\\ \text{ЛК}}} + \underbrace{N^2}_{\substack{\text{множества}\\ \text{использованных}\\ \text{значений в}\\ \text{столбцах ЛК}}} = 2N^2$ до $\underbrace{N}_{\substack{\text{текущая}\\ \text{строка}\\ \text{ЛК}}} + \underbrace{N^2}_{\substack{\text{множества}\\ \text{использованных}\\ \text{значений в}\\ \text{столбцах ЛК}}}$, асимптотическая емкостная сложность при этом не снижается, оставаясь квадратичной.

Проверку корректности формируемого квадрата можно организовать не по строкам, а по столбцам один за одним. Точнее, достаточно проверить дублирование значений только по одному столбцу, для определенности – по первому. Несложно показать, что в формируемом квадрате первый столбец будет состоять из значений

$$
\begin{aligned}
&v_0 = 0,\\
&v_1 = 0 + d \pmod N,\\
&v_2 = 0 + 2d \pmod N,\\
&\ldots\\
&v_i = 0 + id \pmod N,\\
&\ldots\\
&v_{N-1} = 0 + (N-1)d \pmod N.
\end{aligned}
\tag{3}
$$

Например, для приведенного выше примера (рис. 5) для $N = 6$ и $d = 5$ получаем:

$$
\begin{aligned}
&v_0 = 0,\\
&v_1 = 0 + 5 \equiv 5 \pmod 6,\\
&v_2 = 0 + 10 \equiv 4 \pmod 6,\\
&v_3 = 0 + 15 \equiv 3 \pmod 6,\\
&v_4 = 0 + 20 \equiv 2 \pmod 6,\\
&v_5 = 0 + 25 \equiv 1 \pmod 6.
\end{aligned}
$$

j-й столбец формируемого ЛК *L* при этом будет образован значениями

$$L_{ij} = v_i = j + id \pmod N.$$

Цель выполняемой проверки на корректность формируемого ЛК – убедиться в том, что все значения в столбце различны. Несложно показать, что если значения различны в одном столбце, то они будут различны и во всех остальных (с точностью до циклического сдвига значений v_i в рассматриваемой паре столбцов). Таким образом, проверять можно только один из столбцов, что и было отмечено выше. Математически проверка сводится к тому, чтобы среди значений (3) отсутствовали повторы, в случае чего значения v_i элементов столбца формируют множество $\{0, 1, 2, ..., N-1\}$. Таким образом, упрощенный вариант алгоритма сводится к следующему: для всех смещений $0 < d < N$ необходимо убедиться в неповторяемости значений $v_i = id \pmod N, i = \overline{0, N-1}$. Несложно заметить, что в данной реализации алгоритма $t \simeq O(N^2)$ (в *N* раз меньше, чем в предыдущем варианте реализации путем полного построения ЛК), а для проверки неповторяемости нужен линейный массив из *N* булевых элементов, т.е. $m \simeq O(N)$ (аналогично в *N* раз меньше).

Практическая реализация проверки на базе формул (2) не включает в своем составе операций деления и умножения, что допускает ее эффективное практическое воплощение (программное или аппаратное). Кажущееся противоречие, связанное с необходимостью вычисления произведения id с последующим нахождением его остатка от деления на *N*, при практической реализации с легкостью обходится с использованием кода вроде

```
curr_value = 0;

for (i = 1; i < N; i++) {
    curr_value += d;
    if (curr_value >= N)
        curr_value -= N;

    // Проверка уникальности значения curr_value
    ...
}
```

Несложно показать, что повторы можно искать не во всех значениях v_i, вместо этого ожидая только повтора значения 0, который встретится через *N* итераций в случае взаимной простоты *N* и *d*, либо раньше в случае, если указанная пара чисел имеет общие делители. Это позволяет отка-

заться от использования массива для проверки дублирования значений v_i, сократив емкостную сложность алгоритма до $m \simeq O(1)$.

При практической реализации алгоритма можно учесть свойства симметрии. Очевидно, что сдвиг строк циклического квадрата вправо на *d* элементов и влево на *d* элементов будет давать один и тот же результат с позиции свойств получаемого квадрата (один квадрат из другого можно получить путем отражения по горизонтали с последующей нормализацией по первой строке). С учетом данной особенности проверять можно не все *d*, а только в диапазоне $2 \le d \le \left\lfloor \frac{N-1}{2} \right\rfloor$, где $\lfloor x \rfloor$ – обозначение операции округления вниз (усечения), что не влияет на временную асимптотику алгоритма, сокращая необходимые затраты вычислительного времени при его практической реализации приблизительно в 2 раза (случай $d = 1$ можно не проверять, т.к. для любого *N* получаемый ЛК будет являться корректным ввиду того, что значения *N* и 1 взаимно просты). Если для некоторого *d* квадрат получился корректным, то и для $d' = N - d \equiv -d \pmod N$ он также будет корректным. Для четных $N > 2$ при этом остается непроверенным центральное значение $d = \frac{N}{2}$, но его можно не проверять, т.к. квадрат для него корректным не будет ввиду того, что значения $\frac{N}{2}$ и *N* не взаимно просты и имеют общий делитель, равный $\frac{N}{2}$. Учет этой особенности при программной реализации позволяет снизить время работы приблизительно в 1,8–1,9 раза (см. табл.).

С целью практической апробации предложенного способа вычисления значения функции Эйлера с использованием циклических латинских квадратов была разработана программная реализация всех рассмотренных выше методов и алгоритмов. Искомые значения функции Эйлера $\varphi(N)$ совпали для всех способов ее вычисления, что подтверждает корректность рассмотренных выше теоретических выкладок, результаты измерения времени приведены в таблице.

Таблица. Результаты времени вычисления значения функции Эйлера, тактов процессора Core i7 4770 для однопоточной программной реализации (t_1 – время расчета на базе формулы (2), для факторизации использован простейший алгоритм перебора делителей до $\sqrt{N}$; t_2 – время расчета по определению с использованием алгоритма Евклида; t_3 – время расчета путем полного построения циклического ЛК с проверкой его корректности; t_4 – время расчета путем поиска дублирования нуля в первом столбце ЛК; t_5 – время расчета путем поиска дублирования нуля в первом столбце ЛК с учетом симметрии), наилучшие значения выделены серым.

N	t_1	t_2	t_3	t_4	t_5
1	221	94	227	175	287
2	54	166	79	154	66
3	166	605	284	163	75
4	217	314	568	257	76
5	211	613	858	475	311
6	166	456	1007	583	266
7	221	846	1236	749	299
8	239	843	1438	783	308
9	229	873	1991	810	553
10	200	822	2019	916	447
20	305	1913	6519	2370	1481
30	580	3602	12497	4820	2880
40	241	4497	22573	7700	4929
50	565	6495	80645	12095	7477

Полученные в ходе вычислительного эксперимента значения позволяют сделать вывод о том, что при практической программной реализации для малых значений аргументов время вычисле-

ния на базе циклических ЛК сопоставимо с временем работы программной реализации на базе факторизации. При б**о**льших значениях аргумента функции Эйлера реализация на базе факторизации является более быстрой.

Оценим потенциал предложенного способа вычисления значения функции Эйлера через подсчет циклических латинских квадратов при реализации операций на базе длинной арифметики. Например, в составе алгоритма RSA-1024 [8], криптостойкость которого базируется на сложности факторизации большого составного числа и, соответственно, вычислительной сложности определения значения функции Эйлера (с учетом ее свойства мультипликативности), используется работа с $n=1024$ разрядными двоичными числами (308 цифр в десятичной форме записи), а для их обработки используется длинная арифметика. В данном случае число двоичных разрядов – n, значение аргумента функции Эйлера (с точностью до порядка) – $N \simeq 2^n$. В результате обзора литературы было установлено, что асипмтотические временн**ы**е сложности реализации арифметических операций с использованием длинной арифметики составляют следующие значения:

- сложение – $O(n)$;
- умножение:
  - $O\left(n^{1,59}\right)$ по методу Карацюбы [9];
  - $O\left(n^{1,4}\right)$ по методу Тоома-Кука [10];
  - $O(n \cdot \log n \cdot \log\log n)$ через дискретное преобразование Фурье по методу Шёнхаге-Штрассена [11] (на практике эффективно для существенно больших чисел, чем в рассмотренном выше примере RSA-1024);
- деление – $O\left(n^{1,4}\right)$ по методу Бурникеля-Циглера [12], значение экспоненты определяется алгоритмом умножения Тоома.

Таким образом, вычисление функции Эйлера через факторизацию потребует приблизительно $\sqrt{N} \cdot n^{1,4} = \sqrt{2^n} \cdot n^{1,4} = 2^{\frac{n}{2}} \cdot n^{1,4}$ шагов (с точностью до константы), а через циклические ЛК – $N^2 \cdot n = 2^{2n} \cdot n$. Выигрыш во временной сложности в соотношении n («длинные» сложения) по сравнению с $n^{1,4}$ («длинные» деления) компенсируется проигрышем в сравнении значений величин $2^{\frac{n}{2}}$ и 2^{2n}.

Заключение

Таким образом, в статье произведено перечисление и подсчет циклических латинских квадратов различного типа и их обобщений на перестановочные квадраты, получены новые числовые ряды и установлена взаимосвязь числа нормализованных по первой строке циклических ЛК порядка N с функцией Эйлера $\varphi(N)$. На основании результатов вычислительных экспериментов, эмпирически подтверждающих установленную взаимосвязь, предложен новый способ вычисления значения функции Эйлера через циклические ЛК. В разработанной последовательной программной реализации он не является более эффективным в вычислительном плане по сравнению с известным способом на базе факторизации аргумента и формулы (2) как для классической реализации, так и для реализации на базе длинной арифметики, однако к его достоинствам, открывающим путь для его дальнейшей доработки, можно отнести хорошую распараллеливаемость и простоту реализации ввиду отсутствия необходимости выполнения операций умножения и деления.

Автор статьи выражает благодарность citerra [Russia Team] с интернет-портала BOINC.ru и Andrew Howroyd за ряд ценных замечаний в процессе разработки алгоритмов, модификации описания числовых рядов в OEIS и подготовки статьи к публикации.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Colbourn C.J., Dinitz J.H. Handbook of Combinatorial Designs, Second Edition. Chapman & Hall/CRC, 2006. 1016 p.
2. Keedwell A.D., Dénes J. Latin Squares and their Applications. Elsevier, 2015. 438 p. DOI: 10.1016/C2014-0-03412-0.

3. Atkin A.O.L., Hay L., Larson R.G. Enumeration and construction of pandiagonal Latin squares of prime order // Computers & Mathematics with Applications. Vol. 9. Iss. 2. 1983. pp. 267–292. DOI: 10.1016/0898-1221(83)90130-X.
4. Dabbaghian V., Wu T. Constructing non-cyclic pandiagonal Latin squares of prime orders // Journal of Discrete Algorithms. Vol. 30. 2015. pp. 70–77.
5. Sloane N.J.A. The on-line encyclopedia of integer sequences // https://oeis.org/
6. Sándor J., Mitrinovic D.S., Crstici B. Handbook of Number Theory I. Springer Netherlands, 2006. 622 p.
7. Manzyuk M., Nikitina N., Vatutin E. Start-up and the Results of the Volunteer Computing Project RakeSearch // Communications in Computer and Information Science book series. Vol. 1129. Springer, 2019. pp. 725–734. DOI: 10.1007/978-3-030-36592-9_59.
8. Rivest R., Shamir A., Adleman L. A method for obtaining digital signatures and public-key cryptosystems // Commun. ACM — NYC. ACM, 1978. Vol. 21, Iss. 2. pp. 120–126. DOI: 10.1145/359340.359342.
9. Карацуба А., Офман Ю. Умножение многозначных чисел на автоматах // Доклады Академии Наук СССР. 1962. Т. 145, № 2. С. 293–294.
10. Knuth D. The Art of Computer Programming. Volume 2 (3rd Ed.): Seminumerical Algorithms. Addison-Wesley, 1997. DOI: 10.5555/270146.
11. Schönhage A., Strassen V. Schnelle Multiplikation großer Zahlen // Computing. 1971. № 7. pp. 281–292.
12. Burnikel C., Ziegler J. Fast Recursive Division // Research report. Max-Planck-Institut fur Informatik, Saarbrucken, 1998. 29 p.

Ватутин Эдуард Игоревич
Доцент каф. Вычислительной техники,
Юго-Западный государственный университет
305040, Россия, г. Курск, ул. 50 лет Октября, д. 94,
ORCID 0000-0002-7362-7387
Тел.: +7-4712-22-26-65
Эл. почта: evatutin@rambler.ru

E.I. VATUTIN

ENUMERATING CYCLIC LATIN SQUARES AND EULER TOTIENT FUNCTION CALCULATIG USING THEM

The article describes computational experiments aimed to enumerating the cyclic Latin squares (LS) and cyclic diagonal Latin squares (DLS) of order *N*. The obtained numerical series for the normalized by the first row cyclic LS and DLS are known, they are present in the Online Encyclopedia of Integer Series (OEIS) under the numbers A000010 and A123565, which made it possible to establish a connection of this type of squares with other well-known objects in the field of combinatorics and number theory, including the Euler totient function. Enumerating the number of cyclic LS and DLS of general form made it possible to find new numerical series, that was added to OEIS under the numbers A338522 and A338562. Cyclic squares, by definition, are generated by the corresponding cyclic permutations. This definition can be extended to permutations, the elements of which form a arbitrary cycle of length *N*, and the corresponding permutation squares have properties similar to cyclic squares. Using normalized to the first row cyclic LS it is possible to develop a new algorithm for calculating the Euler totient function $\varphi(N)$ with time $t \simeq O(N^2)$ and memory $m \simeq O(1)$ asymptotic complexities. This algorithm loses to the known algorithms based on the factorization of the argument of the Euler totient function for a sequential software implementation, however, it has good potential for parallelization and does not include multiplication and division operations.

Keywords: *combinatorics, Latin squares, cyclic Latin squares, integer sequences, OEIS, Euler totient function.*

REFERENCES

1. Colbourn C.J., Dinitz J.H. Handbook of Combinatorial Designs, Second Edition. Chapman & Hall/CRC, 2006. 1016 p.
2. Keedwell A.D., Dénes J. Latin Squares and their Applications. Elsevier, 2015. 438 p. DOI: 10.1016/C2014-0-03412-0.
3. Atkin A.O.L., Hay L., Larson R.G. Enumeration and construction of pandiagonal Latin squares of prime order // Computers & Mathematics with Applications. Vol. 9. Iss. 2. 1983. pp. 267–292. DOI: 10.1016/0898-1221(83)90130-X.
4. Dabbaghian V., Wu T. Constructing non-cyclic pandiagonal Latin squares of prime orders // Journal of Discrete Algorithms. Vol. 30. 2015. pp. 70–77.
5. Sloane N.J.A. The on-line encyclopedia of integer sequences // https://oeis.org/
6. Sándor J., Mitrinovic D.S., Crstici B. Handbook of Number Theory I. Springer Netherlands, 2006. 622 p.
7. Manzyuk M., Nikitina N., Vatutin E. Start-up and the Results of the Volunteer Computing Project RakeSearch // Communications in Computer and Information Science book series. Vol. 1129. Springer, 2019. pp. 725–734. DOI: 10.1007/978-3-030-36592-9_59.

8. Rivest R., Shamir A., Adleman L. A method for obtaining digital signatures and public-key cryptosystems // Commun. ACM — NYC. ACM, 1978. Vol. 21, Iss. 2. pp. 120–126. DOI: 10.1145/359340.359342.
9. Karatsuba A., Ofman Yu. Multiplying multidigit numbers on automatic machines // Proceedings of RAS. 1962. Vol. 145, No. 2. pp. 293–294. (in Russian).
10. Knuth D. The Art of Computer Programming. Volume 2 (3rd Ed.): Seminumerical Algorithms. Addison-Wesley, 1997. DOI: 10.5555/270146.
11. Schönhage A., Strassen V. Schnelle Multiplikation großer Zahlen // Computing. 1971. № 7. pp. 281–292.
12. Burnikel C., Ziegler J. Fast Recursive Division // Research report. Max-Planck-Institut fur Informatik, Saarbrucken, 1998. 29 p.

Eduard I. Vatutin
Docent of Department of Computing Techniques,
Southwest State University
305040, Russia, Kursk, 50 let Oktyabrya st., 94,
ORCID 0000-0002-7362-7387
Phone: +7-4712-22-26-65,
E-mail: evatutin@rambler.ru