ПРОБЛЕМЫ ПЕРЕДАЧИ ИНФОРМАЦИИ

Том 54 **2018** **Вып. 4**

УДК 621.391.15 : 519.72

© 2018 г. **И.С. Сергеев**

О СЛОЖНОСТИ ФИБОНАЧЧИЕВА КОДИРОВАНИЯ[1]

Показано, что перевод n-разрядного числа из двоичного в фибоначчиево представление и обратно может быть реализован булевыми схемами сложности $O(M(n)\log n)$, где $M(n)$ – сложность целочисленного умножения. Для более общего случая r-фибоначчиевых представлений полученные оценки сложности имеют вид $2^{O(\sqrt{\log n})}n$.

§ 1. Введение

Последовательность чисел Фибоначчи r-го порядка определяется соотношением

$$F_n^{(r)} = F_{n-1}^{(r)} + \ldots + F_{n-r}^{(r)}$$

и начальными условиями

$$F_{-r+2}^{(r)} = \ldots = F_{-1}^{(r)} = F_0^{(r)} = 0, \quad F_1^{(r)} = 1.$$

Назовем *r-фибоначчиевым представлением* неотрицательного числа $A \in \mathbb{Z}$ булев вектор (или строку) $[a_n, \ldots, a_2]$, такой что

$$A = \sum_{i=2}^{n} a_i F_i^{(r)}, \tag{1}$$

где $a_i \in \{0, 1\}$. Если при этом $a_{i+1} \ldots a_{i+r} = 0$ для всех $i \geqslant 1$, то представление (1) называется *каноническим*. Легко проверить, что каноническое представление существует, и оно единственно. Множество r-фибоначчиевых представлений совпадает с множеством булевых строк, в которых нет r подряд идущих единиц. Далее, если не оговаривается иное, под r-фибоначчиевыми представлениями будем понимать канонические представления.

Для стандартных чисел Фибоначчи вместо $F_k^{(2)}$ будем использовать обозначение F_k, а связанное с ними представление называть просто фибоначчиевым.

Фибоначчиевы представления принято связывать с именем Э. Цекендорфа [1], а r-фибоначчиевы представления, видимо, впервые рассмотрены в работе [2], где изучены их основные свойства, а также установлена связь с кодированием чисел двоичными строками, в которых запрещены серии из $r+1$ нулей или единиц (назовем их r-последовательностями).

Действительно, любой r-последовательности $a^{s_1}b^{s_2}a^{s_3}\ldots$ соответствует слово $(s_1, s_2, s_3, \ldots)$ в алфавите $\{1, \ldots, r\}$. При этом множество последовательностей длины n отображается во множество слов веса $\sum s_i = n$. Прообраз каждого слова –

[1] Работа выполнена при частичной финансовой поддержке Российского фонда фундаментальных исследований (номер проекта 17-01-00485а).

две взаимно инверсные последовательности (случаи $a = 0$ и $a = 1$). Слову веса n можно поставить в соответствие r-фибоначчиеву последовательность по правилу $s_1 \to 1^{s_1-1}$, $s_i \to 01^{s_i-1}$, $i \geqslant 2$, и это соответствие взаимно однозначно. Таким образом, каждой r-фибоначчиевой последовательности длины $n-1$ соответствуют две r-последовательности длины n.

Еще одно представление суммами из обобщенных чисел Фибоначчи предложено и изучено в [3]. Последовательность $F_k^{\langle q \rangle}$ задается соотношением

$$F_n^{\langle q \rangle} = F_{n-1}^{\langle q \rangle} + F_{n-q}^{\langle q \rangle}$$

и начальными условиями

$$F_{3-q}^{\langle q \rangle} = \ldots = F_1^{\langle q \rangle} = F_2^{\langle q \rangle} = 1.$$

По аналогии с (1) определим *$\langle q \rangle$-фибоначчиево представление*

$$A = \sum_{i=2}^{n} a_i F_i^{\langle q \rangle}, \tag{2}$$

где $a_i \in \{0, 1\}$. Если $a_{i+1} + \ldots + a_{i+q} \leqslant 1$ для всех $i \geqslant 1$, то представление (2) называется каноническим. Как и в случае r-фибоначчиевых представлений, каноническое $\langle q \rangle$-фибоначчиево представление существует и единственно. Множество таких представлений совпадает с множеством всех булевых строк, в которых единицы отделены друг от друга как минимум $q-1$ нулями.

С обобщенными фибоначчиевыми представлениями также связаны представления в некоторых системах счисления с иррациональными основаниями. В частности, система счисления с основанием $\varphi = \dfrac{1+\sqrt{5}}{2}$ была предложена в [4]. Каноническая запись целого числа в такой системе так же, как и в фибоначчиевой, не содержит повторяющихся единиц, но сама запись длиннее.

Для кодов из обобщенных фибоначчиевых представлений предлагались разнообразные приложения, среди них: асинхронная последовательная передача информации [2], передача строк переменной длины [5], хранение и поиск архивированных данных [6]. Фибоначчиевы коды являются частным случаем RLL- и MTR-кодов[2], используемых для записи информации на некоторые виды накопителей [7].

Для большинства потенциальных приложений имеет значение наличие эффективных алгоритмов кодирования и декодирования, т.е. перехода от двоичного представления чисел к фибоначчиевым и обратно. К этому вопросу примыкает вопрос об эффективном выполнении арифметических действий с числами в фибоначчиевых представлениях, в первую очередь, сложения.

Только сравнительно недавно в работе [8] были построены булевы схемы[3] для сложения и вычитания n-разрядных чисел в фибоначчиевом представлении линейной сложности и глубины $O(\log n)$. Впрочем, фактически сумматор с такими характеристиками был описан еще в [10]. Иногда выгодно отказаться от канонического представления. В работе [11] показано, что в фибоначчиевом представлении, допускающем любые коэффициенты из $\{0, \pm 1\}$, сложения и вычитания выполняются схемами глубины $O(1)$. Из таких сумматоров строятся схемы сложения N чисел

[2] RLL – runlength limited; MTR – maximum transition run.

[3] Определения булевых схем, их сложности и глубины см., например, в [9]. Приводимые далее оценки сложности справедливы и для ряда других моделей вычислений: некоторых видов программ, машин Тьюринга и т.п.

линейной сложности и глубины $O(\log N)$. Для перехода к каноническому представлению можно использовать[4] схему из [8].

Сложность тривиальных схем для перехода между любым фибоначчиевым и двоичным представлениями – квадратична. Схемы перехода между фибоначчиевым и двоичным представлениями n-разрядных чисел, предложенные в [8], имеют сложность $O(n^2)$ и глубину $O(\log n)$. Схемы составляются либо из сумматоров-компрессоров, складывающих числа Фибоначчи в двоичном представлении, либо из неканонических сумматоров [11], складывающих степени двойки в фибоначчиевом представлении. Во втором случае можно также использовать и компрессоры.

Далее будет показано, что оценка сложности переходов между двоичным и фибоначчиевым представлениями может быть понижена до $O(M(n)\log n)$, где $M(n)$ – верхняя оценка сложности умножения n-разрядных двоичных чисел, удовлетворяющая условию монотонного неубывания функции $M(n)/n$. Напомним, что наилучшие известные оценки сложности умножения имеют вид[5] $M(n) = n\,2^{O(\log^* n)}\log n$ (см. [12]). Кроме того, мы покажем, что переходы между двоичным и r-фибоначчиевым представлениями реализуются схемами сложности $2^{O(\sqrt{\log n})}n$. С такой же сложностью выполняются переходы между двоичным и $\langle 3\rangle$-фибоначчиевым представлениями.

Оговоримся, что предпринятое исследование сложности преобразований сугубо теоретическое. На практике кодирование, как правило, выполняется с блоками сравнительно небольшого размера n. В этой ситуации применение методов, эффективных при $n \to \infty$, часто оказывается неуместным.

§ 2. Операции в фибоначчиевом представлении

Излагаемый далее метод выполнения перехода к фибоначчиеву представлению и обратно аналогичен методу Шёнхаге для переходов между системами счисления с разными основаниями (см., например, [13]). Метод основан на идее "деления пополам" – разбить число на две части, обработать каждую из них отдельно и затем объединить результаты.

Теорема 1. *Перевод n-разрядного числа из двоичного представления в фибоначчиево выполняется схемой сложности $O(M(n)\log n)$.*

Доказательство. Воспользуемся известной формулой для произведения числа Фибоначчи и числа Люка. Числа Люка определяются как

$$L_m = F_{m+1} + F_{m-1} = \varphi_+^m + \varphi_-^m,$$

[4] Иначе схему глубины $O(\log N)$ можно построить в виде дерева из схем-компрессоров. Опишем компрессор, который сводит сложение четырех чисел в неканоническом фибоначчиевом представлении (1), не запрещающем серии из единиц, к сложению трех чисел и имеет линейную сложность и глубину $O(1)$.

Введем обозначение $f_k = (F_{2k+1}, F_{2k})^T$. Удобно разбить фибоначчиево представление (1) на блоки длины 2: $A = \sum v_k f_k$, где $v_k = (a_{2k+1}, a_{2k})$.

При покомпонентном суммировании четырех чисел получается выражение вида $\sum v_k f_k$, в котором компоненты векторов v_k принимают значения от 0 до 4. Легко проверить, что для любого такого вектора v существует представление

$$v f_k = v_1 f_{k+1} + v_0 f_k + v_{-1} f_{k-1},$$

где $v_i \in \{0,1\}^2$. Преобразованные по этой формуле блоки суммы можно распределить между тремя числами, предварительно переписав возникающее при $k = 1$ слагаемое $(x, y)f_0$ в виде $(0, x)f_1$.

[5] Через $\log^* n$ обозначается сверхлогарифм – число, определяемое из условия $\lfloor \underbrace{\log_2 \ldots \log_2}_{\log^* n} n \rfloor = 1$.

где $\varphi_\pm = \dfrac{1 \pm \sqrt{5}}{2}$. При любых m, k справедливо

$$F_k L_m = F_{m+k} - (-1)^k F_{m-k}. \tag{3}$$

Опишем схему преобразования $2n$-разрядного двоичного числа A при $n \geqslant n_0$. Пусть $L_m \geqslant 2^{n+1} > L_{m-1}$. Разделим A на L_m с остатком:

$$A = A_1 L_m + A_0.$$

Построим фибоначчиевы представления для чисел A_1 и A_0 (соответственно, $(n-1)$- и $(n+2)$-разрядных).

Для объединения результатов воспользуемся формулой (3): применяя к компонентам записи числа A_1 правило (3), выражаем число $A_1 L_m$ разностью двух чисел в фибоначчиевом представлении. Слагаемое составляют компоненты из правой части (3) с положительными коэффициентами, вычитаемое – с отрицательными. Заметим, что $A_1 < 2^{n-1} < F_{m-1}$, поэтому в формулах (3) не участвуют числа Фибоначчи с номерами, меньшими 2.

Теперь для получения фибоначчиевой записи числа A достаточно выполнить сложение и вычитание (воспользуемся методом [8]).

Поскольку деление с остатком имеет сложность $O(M(n))$ (см., например, [9, 13]), то для сложности $T(2n)$ всей схемы справедливо рекуррентное соотношение

$$T(2n) \leqslant T(n+2) + T(n-1) + O(M(n)),$$

разрешаемое стандартным образом как $T(n) = O(M(n) \log n)$. ▲

Если алгоритм умножения имеет сложность $M_0(n)$ и глубину $D(n)$, то деление может быть реализовано схемой сложности $O(M_0(n))$ и глубины $O(D(n) \log n)$. Тогда переход к фибоначчиеву представлению выполняется схемой сложности $O(M_0(n) \log n)$ и глубины $O(D(n) \log^2 n)$. На самом деле, если $D(n) = O(\log n)$, то деление может быть выполнено (см. [14]) схемой сложности $O(M_0(n))$ и глубины $O(\log n \log\log n)$. Как следствие, оценка глубины схемы перехода понижается до $O(\log^2 n \log\log n)$.

Теорема 2. *Перевод n-разрядного числа из фибоначчиева представления в двоичное выполняется схемой сложности $O(M(n) \log n)$.*

Доказательство. При переходе к двоичному представлению порядок действий обратный. Выбираем параметр m, примерно равный половине длины фибоначчиева представления числа. Выражая старшие компоненты по формуле (3), получаем

$$A = A_1 L_m + B, \quad B = A_1' - A_1'' + A_0,$$

где слагаемые A_1' и A_1'' включают в себя крайние правые компоненты из (3). Далее вычисляем фибоначчиево представление числа B. Применяем алгоритм построения двоичного представления чисел A_1 и B – они примерно вдвое короче исходного числа. Для завершения вычислений остается умножить и сложить двоичные числа. ▲

При использовании алгоритма умножения сложности $M_0(n)$ и глубины $D(n)$ получаем схему перехода сложности $O(M_0(n) \log n)$ и глубины $O(D(n) \log n)$. Каноничности фибоначчиевых представлений на промежуточных шагах алгоритма можно не требовать.

§ 3. Операции в r-фибоначчиевом представлении

Для записи обобщенных чисел Фибоначчи известны формулы вида

$$F_n^{(r)} = c_1\xi_{r,1}^n + \ldots + c_r\xi_{r,r}^n, \tag{4}$$

где $\xi_{r,1}, \ldots, \xi_{r,r}$ – различные корни многочлена $x^r - x^{r-1} - \ldots - x - 1$, а c_i – некоторые постоянные. Установлено, что многочлен имеет один вещественный корень (под ним далее будем понимать $\xi = \xi_{r,1}$) из интервала $(1; 2)$, а остальные корни по модулю меньше 1. В частности,

$$F_n^{(r)} = c_1\xi^n + \varepsilon_n, \tag{5}$$

где $c_1 < 1$ и $|\varepsilon_n| < 1/2$. Подробнее см., например, в [15–17].

В качестве подходящего аналога формулы (3) можно рассматривать частный случай формулы, полученной в [18]:

$$F_n^{(r)} = \sum_{j=1}^{r} C_{j,m}^{(r)} F_{n-jm}^{(r)}, \tag{6}$$

где $C_{j,m}^{(r)} \in \mathbb{Z}$ – не зависящие от n коэффициенты, играющие роль чисел Люка в теории чисел Фибоначчи высших порядков. Коэффициенты определяются формулами

$$C_{j,m}^{(r)} = (-1)^{j+1} \sum_{\substack{S\subset\{1,\ldots,r\}\\ |S|=j}} \prod_{i\in S} \xi_{r,i}^m. \tag{7}$$

Отметим, что равенство (6) легко проверяется. Во-первых, достаточно доказать (6) с подстановкой вместо $F_k^{(r)}$ чисел $\xi_{r,i}^k$. В этом случае формула (6) будет следовать из (4) в силу ее линейности относительно $F_k^{(r)}$. Остается заметить, что эквивалент формулы (6) для степеней $\xi_{r,i}^k$ равносилен тождеству

$$x^{n-rm}\left(x^m - \xi_{r,1}^m\right)\ldots\left(x^m - \xi_{r,r}^m\right) = 0$$

при $x = \xi_{r,i}$ (коэффициенты $C_{j,m}^{(r)}$ из (7) возникают при раскрытии скобок; их целочисленность вытекает из целочисленности коэффициентов многочлена $x^r - x^{r-1} - \ldots - x - 1$, корнями которого являются $\xi_{r,i}$).

Для перехода к двоичному представлению предлагается блочный метод, на уровне основной идеи вычисления – аналог метода Тоома умножения чисел [19].

Теорема 3. *Перевод N-разрядного числа из r-фибоначчиева представления в двоичное выполняется схемой сложности $2^{O(\sqrt{\log N})}N$.*

Доказательство. Обозначим через $V_{[k,\ell]}$, $k \geqslant \ell$, множество чисел, r-фибоначчиева запись которых содержит ненулевые коэффициенты только в позициях $k, \ldots, \ell$.

Опишем схему преобразования для некоторого числа $A \in V_{[n+N,n+1]}$. При $n = 1$ получаем искомую схему.

Пусть $N = km$ и $n = 1 + \ell m$, $\ell \leqslant (r-1)k$. Обозначим $V_j = V_{[jm+1,(j-1)m+2]}$. Разобьем число на блоки длины m:

$$A = A_k + \ldots + A_1, \quad A_i \in V_{i+\ell}. \tag{8}$$

Пусть $B^{\to j}$ обозначает число, получаемое из числа B сдвигом r-фибоначчиевой записи на jm позиций вправо. В частности, если $B \in V_i$, то $B^{\to j} \in V_{i-j}$.

Теперь, исходя из (8), будем применять правило (6) поблочно ко всем блокам из V_i, $i > r$, (при этом ограничении не будут возникать числа Фибоначчи с отрицательными индексами) до тех пор, пока в формуле не останутся только блоки из $V_1 \cup \ldots \cup V_r$.

В итоге каждое из чисел A_i может быть записано в виде

$$A_i = A_i^{\to i+\ell-r} B_{i,r} + \ldots + A_i^{\to i+\ell-1} B_{i,1},$$

где $B_{i,s}$ – некоторые функции от $C_{j,m}^{(r)}$. Окончательно имеем

$$A = \sum_{i=1}^{k} \sum_{s=1}^{r} A_i^{\to i+\ell-s} B_{i,s}. \tag{9}$$

По построению $B_{i,s}$ являются суммами не более чем $r^{k+\ell}$ одночленов от $C_{j,m}^{(r)}$ степени не выше $k+\ell$. Согласно (7) имеем $\left|C_{j,m}^{(r)}\right| \leqslant (2\xi_{r,1}^m)^r$. Поэтому

$$|B_{i,s}| < (r(2\xi_{r,1}^m)^r)^{k+\ell} == 2^{O(N)}.$$

С помощью (9) задача размерности N сводится к rk задачам размерности m (преобразованиям чисел $A_i^{\to i+\ell-s}$), а также rk умножениям и сложениям $O(N)$-разрядных двоичных чисел. Для сложности $\Phi(N)$ схемы имеем рекуррентное соотношение

$$\Phi(N) \leqslant kr\Phi(m) + O(kM(N)),$$

которое разрешается стандартно как $\Phi(N) = 2^{O(\sqrt{\log N})} N$. Параметр k следует выбирать в виде $c^{\sqrt{\log N}}$. Если N не делится на k, то рассматриваем вместо него число $k\lceil N/k \rceil$. ▲

Представляется правдоподобным, что сложение и вычитание чисел в r-фибоначчиевом представлении должно выполняться с линейной сложностью. Однако для обоснования оценки сложности перехода к r-фибоначчиеву представлению нам будет достаточно существенно более слабого, но просто доказываемого утверждения.

Лемма. *Прибавление к числу в r-фибоначчиевом представлении числа веса* 1, *т.е.* $F_m^{(r)}$, *выполняется с линейной сложностью.*

Доказательство. Предварительно заметим, что приведение произвольной строки из 0 и 1 к каноническому r-фибоначчиеву виду выполняется за один или два прохода строки. Во-первых, двигаясь по строке справа налево, достаточно произвести замены $01^{kr+s} \to (10^{r-1})^k 01^s$, где $s < r$. Но проще пройти по строке сначала слева направо, затем справа налево, выполняя замены $01^r \to 10^r$. Эти способы (на материале фибоначчиевых представлений) обсуждаются в [10].

Теперь, для удобства индуктивного рассуждения, будем рассматривать добавление числа, в записи которого содержится только один ненулевой блок из $p < r$ единиц подряд.

Сначала выполним сложение поразрядно. Возможны две ситуации.

1) Полученная строка состоит только из 0 и 1. Тогда поступаем, как описано выше.

2) В строке имеются двойки. Заметим, что при этом строка может содержать лишь один длинный блок из не менее чем r ненулевых символов, и лишь в этом блоке присутствуют двойки. Применяя к левой части блока с двойками правило $0x_1 \ldots x_r \to 1\bar{x}_1 \ldots \bar{x}_r$, где $\bar{x} = x - 1$, столько раз, сколько нужно (на самом деле, не более двух) добьемся того, что блок с двойками будет иметь длину меньше r.

Для самой правой двойки в блоке, если она расположена не ближе r символов к правому краю строки, можно выполнить преобразование $2x_1 \ldots x_r \to 1x'_1 \ldots x'_r$,

где $x' = x + 1$. В результате в строке появится ненулевой блок длины больше r. К нему применим описанную выше процедуру сокращения длины. При этом все двойки левее рассмотренной сократятся, но справа может появиться новый блок с двойками.

К этому блоку применяем аналогичную процедуру и действуем так либо до тех пор, пока двойки не исчезнут, либо пока позиция самой правой двойки не окажется среди правых r символов строки.

В последнем случае выполняется замена $2x_1 \dots x_s \to 1x'_1 \dots x'_{s-1}x''_s$, где $x'' = x+2$, корректная в силу $F_k^{(r)} = 2^{k-2}$ при $2 \leqslant k \leqslant r+1$. Если при этом появляется блок длины не менее r, то применяем к нему процедуру сокращения длины. Так или иначе, в результате получим строку, не содержащую ненулевых блоков длины r, в которой двойки расположены только в крайнем правом отрезке из $r-2$ символов, а в самой правой позиции может быть тройка. От этих двоек и тройки можно избавиться, просматривая крайний фрагмент справа налево и применяя правило $x, 2+y \to x+1, y$. В итоге приходим к строке из 0 и 1, которая далее обрабатывается, как указано в начале доказательства.

Сложность алгоритма линейна, поскольку он выполняется за несколько проходов по строке, а один проход реализуется префиксной схемой линейной сложности, а при желании – и логарифмической глубины (см. [8, 9]). ▲

Т е о р е м а 4. *Перевод N-разрядного числа из двоичного представления в r-фибоначчиево выполняется схемой сложности $2^{O(\sqrt{\log N})}N$.*

Д о к а з а т е л ь с т в о. Снова воспользуемся приемом деления пополам. Пусть дано $2n$-разрядное двоичное число A, где $n \geqslant n_0$. Обозначим $\xi = \xi_{r,1}$ и выберем параметр m из условия $2^n \leqslant \xi^m < 2^{n+1}$. Разделим A на ξ^m с остатком:

$$A = A'_1\xi^m + A'_0, \quad A'_1 \in \mathbb{Z}, \quad 0 \leqslant A'_0 < \xi^m.$$

Если $A'_1 < m$, то положим $A_1 = 0$ и $A_0 = A$. В противном случае положим $A_1 = A'_1 - m$ и $A_0 = A'_0 + m\xi^m$. В обоих случаях обеспечено равенство $A = A_1\xi^m + A_0$.

Построим r-фибоначчиево представление для A_1: $A_1 = \sum a_i F_i^{(r)}$. Из (5) вытекает $F_k^{(r)}\xi^m = F_{m+k}^{(r)} + \varepsilon_k\xi^m - \varepsilon_{m+k}$. Тогда

$$A = A_1^{\leftarrow m} + B, \quad B = A_0 + \xi^m \sum a_i\varepsilon_i - \sum a_i\varepsilon_{m+i},$$

где $A_1^{\leftarrow m}$ получается сдвигом r-фибоначчиевой записи числа A_1 влево на m позиций. Остаток B неотрицателен, поскольку при $A_1 = 0$ имеет место $B = A$, а в противном случае, в силу $A_1 < 2^n \leqslant \xi^m$, r-фибоначчиева запись числа A_1 имеет длину не более $m-1$, поэтому

$$|B - A_0| < (m-1)\xi^m + m < m\xi^m \leqslant A_0.$$

Отсюда же следует верхняя оценка $B \leqslant (2m+1)\xi^m$.

Вычислим двоичное представление числа $A_1^{\leftarrow m}$, найдем разность $B = A - A_1^{\leftarrow m}$, затем построим r-фибоначчиево представление для B. Наконец, искомое представление числа A получается сложением r-фибоначчиевых представлений чисел $A_1^{\leftarrow m}$ и B. Представления чисел $A_1^{\leftarrow m}$ и B пересекаются только по $O(\log n)$ разрядам, поэтому их сложение сводится к прибавлению к объединенному числу $O(\log n)$ чисел веса 1 (здесь применим лемму). В итоге, для общей сложности алгоритма получаем рекуррентное соотношение

$$T(2n) \leqslant T(n) + T(n + c\log n) + \Phi(m) + O(M(n)) + O(n\log n).$$

Подставляя оценку $\Phi(m)$ из теоремы 3, получаем что это соотношение разрешается как $T(n) = 2^{O(\sqrt{\log n})}n$. ▲

В заключение на примере $\langle q\rangle$-фибоначчиевых представлений кратко обсудим возможности расширения изложенных результатов.

Формулы (6), (7) выполняются для широкого класса рекуррентностей (см. [18]). Поэтому, в частности, результат теоремы 3 переносится также и на $\langle q\rangle$-фибоначчиевы представления.

Многочлен $x^q - x^{q-1} - 1$ при $q > 5$ имеет комплексные корни, по модулю большие единицы, поэтому свойство (4) для $\langle q\rangle$-фибоначчиевых последовательностей, вообще говоря, не имеет места. При $q \leqslant 5$ оно выполняется в форме $F_n^{\langle q\rangle} = c_1\xi^n + O(1)$, достаточной для стратегии доказательства теоремы 4. Для реализации этой стратегии нужен также аналог леммы, который легко получить в случае $q = 3$. Как следствие, оценки теорем 3 и 4 справедливы и для сложности переходов между двоичным и $\langle 3\rangle$-фибоначчиевым представлениями.

Автор благодарит С.Б. Гашкова, познакомившего его с рассматриваемой задачей.

СПИСОК ЛИТЕРАТУРЫ

1. *Lekkerkerker C.G.* Representation of Natural Numbers as a Sum of Fibonacci Numbers // Simon Stevin. 1952. V. 29. P. 190–195.
2. *Kautz W.* Fibonacci Codes for Synchronization Control // IEEE Trans. Inform. Theory. 1965. V. 11. № 2. P. 284–292.
3. *Daykin D.E.* Representation of Natural Numbers as Sums of Generalized Fibonacci Numbers // J. London Math. Soc. 1960. V. 35. № 2. P. 143–160.
4. *Bergman G.* A Number System with an Irrational Base // Math. Magazine. 1957. V. 31. № 2. P. 98–110.
5. *Apostolico A., Fraenkel A.S.* Robust Transmission of Unbounded Strings Using Fibonacci Representations // IEEE Trans. Inform. Theory. 1987. V. 33. № 2. P. 238–245.
6. *Klein S.T., Ben-Nissan M.K.* On the Usefulness of Fibonacci Compression Codes // Comput. J. 2010. V. 53. № 6. P. 701–716.
7. *Immink K.A.S.* Codes for Mass Data Storage Systems. Eindhoven, The Netherlands: Shannon Foundation Publ., 2004.
8. *Ahlbach C., Usatine J., Frougny C., Pippenger N.* Efficient Algorithms for Zeckendorf Arithmetic // Fibonacci Quart. 2013. V. 51. № 3. P. 249–255.
9. *Wegener I.* The Complexity of Boolean Functions. Chicester: Wiley; Stuttgart: Teubner, 1987.
10. *Berstel J.* Fibonacci Words—A Survey // The Book of L. Berlin; New York: Springer-Verlag, 1986. P. 13–27.
11. *Frougny C., Pelantová E., Svobodová M.* Parallel Addition in Non-standard Numeration Systems // Theor. Comput. Sci. 2011. V. 412. № 41. P. 5714–5727.
12. *Fürer M.* Faster Integer Multiplication // SIAM J. Comput. 2009. V. 39. № 3. P. 979–1005.
13. *Гашков С.Б.* Занимательная компьютерная арифметика. Быстрые алгоритмы операций с числами и многочленами. М.: Либроком, 2012.
14. *Reif J., Tate S.* Optimal Size Integer Division Circuits // SIAM J. Comput. 1990. V. 19. № 5. P. 912–924.
15. *Miles E.P., Jr.* Generalized Fibonacci Numbers and Associated Matrices // Amer. Math. Monthly. 1960. V. 67. № 8. P. 745–752.
16. *Spickerman W.R., Joyner R.N.* Binet's Formula for the Recursive Sequence of Order k // Fibonacci Quart. 1984. V. 22. № 4. P. 327–331.
17. *Dresden G.P.B., Du Z.* A Simplified Binet Formula for k-Generalized Fibonacci Numbers // J. Integer Seq. 2014. V. 17. № 4. Article 14.4.7 (9 pp.)

18. *Howard F.T.* Generalizations of a Fibonacci Identity // Appl. of Fibonacci Numbers. V. 8 (Proc. 8th Int. Research Conf. on Fibonacci Numbers and Their Applications. Rochester, NY, USA. June 22–26, 1998). Dordrecht: Kluwer, 1999. P. 201–211.
19. *Тоом А.Л.* О сложности схемы из функциональных элементов, реализующей умножение целых чисел // ДАН СССР. 1963. Т. 150. № 3. С. 496–498.

Сергеев Игорь Сергеевич
ФГУП "НИИ "Квант"
isserg@gmail.com

Поступила в редакцию
30.05.2018
После доработки
30.05.2018
Принята к публикации
18.09.2018