

ЭФФЕКТИВНОЕ ПО ВРЕМЕНИ И ПАМЯТИ ВЫЧИСЛЕНИЕ ФУНКЦИИ РИМАНА $\zeta(s)$ В МОДЕЛИ КО-ФРИДМАНА

¹Яхонтов С.В.

¹«Санкт-Петербургский государственный университет», г. Санкт-Петербург, Россия (198504, Россия, Санкт-Петербург, Старый Петергоф, Университетский пр., дом 28), e-mail: SergeyV.Yakhontov@gmail.com, S.Yakhontov@spbu.ru

В данной работе строится FP//LINSPLACE алгоритмический аналог вещественной функции Римана $\zeta(s)$ для алгоритмических вещественных чисел s , $s > 1$, с полиномиальной временной и линейной емкостной сложностью на машине Тьюринга в модели Ко-Фридмана алгоритмических чисел и функций. Для построения алгоритмического аналога функции Римана $\zeta(s)$ предлагается алгоритм расчета двоично-рациональных приближений данной функции для вещественных s , $s > 1$, с полиномиальной временной и линейной емкостной сложностью на машине Тьюринга. Данный алгоритм строится на основе разложения в числовой ряд, который сходится для вещественных s , $s > 1$, при этом показывается и используется в данном алгоритме линейная сходимость числового ряда. Предложенный алгоритм вычисления вещественной функции Римана $\zeta(s)$ включает использование алгоритмов вычисления вещественной функции $(1+x)^h$ и гипергеометрических рядов с полиномиальным временем и линейной памятью. Предложенный алгоритм (модифицированный некоторым образом для работы с комплексными числами) может использоваться для вычисления комплексной функции Римана $\zeta(s)$ для $s = \sigma + it$, $\sigma \neq 1$ (то есть также для $\sigma < 1$), с использованием полиномиального времени и линейной памяти по n , где 2^{-n} – точность вычислений. Модифицированный алгоритм также будет использовать полиномиальное время и линейную память по $\lceil \log_2(t) \rceil$ и экспоненциальное время и экспоненциальную память по $\lceil \log_2(\sigma) \rceil$.

Ключевые слова: алгоритмические (вычислимые) числа и функции, представление функциями Коши, полиномиально вычислимы по времени функции, линейно вычислимы по памяти функции, дзета-функция Римана.

TIME AND SPACE EFFICIENT EVALUATION OF RIEMANN FUNCTION $\zeta(s)$ IN KO-FRIEDMAN MODEL

¹Yakhontov S.V.

¹Saint Petersburg State University, Saint Petersburg, Russian Federation (198504, Saint Petersburg, Petrodvorets, University pr., 28), e-mail: SergeyV.Yakhontov@gmail.com, S.Yakhontov@spbu.ru

In the present paper we construct FP//LINSPLACE algorithmic analog of real Riemann function $\zeta(s)$ for FP//LINSPLACE algorithmic real numbers s , $s > 1$, with polynomial time and linear space on Turing machines in Ko-Friedman model of algorithmic numbers and functions. To construct algorithmic analog of real Riemann function $\zeta(s)$, we consider an algorithm for the evaluation of dyadic rational approximations of the function for FP//LINSPLACE algorithmic real numbers on Turing machine using polynomial time and linear space. The Algorithm is based on a series expansion which converges for real s , $s > 1$; it is shown and used in the algorithm that the series of the function converges linearly. The proposed algorithm of the evaluation of real Riemann function $\zeta(s)$ uses algorithms for the evaluation of real function $(1+x)^h$ and the evaluation of hypergeometric series using polynomial time and linear space. The algorithm from the present paper modified in some way to work with the complex numbers can be used to evaluate complex Riemann function $\zeta(s)$ for $s = \sigma + it$, $\sigma \neq 1$ (so, also for $\sigma < 1$), in polynomial time and linear space in n wherein 2^{-n} is a precision of computations. The modified algorithm will use polynomial time and linear space in $\lceil \log_2(t) \rceil$ and exponential time and exponential space in $\lceil \log_2(\sigma) \rceil$.

Key words: algorithmic (computable) numbers and functions, Cauchy function representation, polynomial-time computable functions, linear-space computable functions, Riemann zeta function.

В данной работе рассматриваются алгоритмические (вычислимые) вещественные числа и функции, которые представляются функциями Коши, вычислимыми на машине Тьюринга [1] (такая модель алгоритмических чисел и функций называется моделью Ко-Фридмана).

Основные результаты по теории алгоритмических чисел и функций и их вычислительной сложности находятся, например, в [1]; основные результаты по теории вычислительной сложности на машине Тьюринга находятся, например, в [2].

Известно, что вещественная функция Римана $\zeta(s)$ [3] является полиномиально вычислимой по времени для натуральных s [4, 5]; алгоритмы из [4, 5] требуют не менее чем $O(n \log_2(n))$ ячеек памяти для вычисления приближенных значений функции $\zeta(s)$ с точностью 2^{-n} . Также имеется алгоритм вычисления гипергеометрических рядов с произвольной точностью с квази-линейным временем и линейной памятью [6], который применим к вычислению значения $\zeta(3)$. Временная и емкостная вычислительная сложность алгоритмов из [4, 5, 6] рассматривается в контексте битовой сложности; на машине Тьюринга алгоритмы из [4, 5] являются полиномиальными по времени и квази-линейными по памяти, а алгоритм из [6] является полиномиальным по времени и линейным по памяти.

В данной работе показывается, что вещественная функция Римана $\zeta(s)$ является полиномиально вычислимой по времени и линейно вычислимой по памяти (с помощью одного и того же алгоритма) на машине Тьюринга для вещественных s , $s > 1$, в модели Ко-Фридмана. Для доказательства данного результата строится алгоритм вычисления вещественной функции Римана $\zeta(s)$ с полиномиальным временем и линейной памятью на машине Тьюринга. Данный алгоритм основан на разложении функции Римана $\zeta(s)$ в числовой ряд [7, 8] для комплексных s , $s \neq 1 + 2\pi i \frac{n}{\log_2(2)}$; в данном алгоритме используются алгоритмы из [9, 10, 11] вычисления вещественной функции $(1+x)^h$ и гипергеометрических рядов с полиномиальным временем и линейной памятью. Для построения алгоритма вычисления функции Римана $\zeta(s)$ используются алгоритм вычисления числовых рядов из [12] и прямой и обратный анализ ошибок округления в численных алгоритмах [9, 10, 11, 13].

Частично результат, изложенный в данной работе, представлялся на научной конференции «Научная дискуссия: вопросы математики, физики, химии, биологии» [14].

1.1 CF алгоритмические числа и функции

Функции Коши в модели Ко-Фридмана [1] – это функции, двоично-рационально сходящиеся к вещественным числам. Функция $\phi: \mathbf{N} \rightarrow \mathbf{D}$ (здесь \mathbf{D} – множество двоично-рациональных чисел) двоично-рационально сходится к вещественному числу x , если $|\phi(n) - x| \leq 2^{-n}$ для всех $n \in \mathbf{N}$; с помощью CF_x обозначается множество всех функций, двоично-рационально сходящихся к числу x .

Definition 1. [1] Вещественное число x называется CF алгоритмическим вещественным числом, если CF_x содержит функцию ϕ , вычислимую на машине Тьюринга.

Definition 2. [1] Вещественная функция f , заданная на отрезке $[a, b]$, называется CF алгоритмической вещественной функцией на отрезке $[a, b]$, если существует машина Тьюринга M с оракульной функцией такая, что для всех $x \in [a, b]$ и для всех $\phi \in CF_x$ функция ψ , вычисляемая машиной M с оракульной функцией ϕ , принадлежит $CF_{f(x)}$.

1.2 Вычислительная сложность алгоритмических функций

Definition 3. [1] Функция $f: [a, b] \rightarrow \mathbf{R}$ называется $t(n)$ вычислимой по времени вещественной функцией на отрезке $[a, b]$, если для всех алгоритмических вещественных чисел $x \in [a, b]$ функция $\psi \in CF_{f(x)}$ (ψ берется из определения CF алгоритмических вещественных функций) является $t(n)$ вычислимой по времени.

Definition 4. [1] Функция $f: [a, b] \rightarrow \mathbf{R}$ называется $s(n)$ вычислимой по памяти вещественной функцией на отрезке $[a, b]$, если для всех алгоритмических вещественных чисел $x \in [a, b]$ функция $\psi \in CF_{f(x)}$ (ψ берется из определения CF алгоритмических вещественных функций) является $s(n)$ вычислимой по памяти.

Входными данными функций ϕ и ψ является запись 0^n (повторение данного символа n раз) в случае, когда функции вычисляются с точностью 2^{-n} .

С помощью **FP//Linspace** обозначается класс строковых функций, вычисляемых за полиномиальное время и линейную память (с помощью одного и того же алгоритма) на машине Тьюринга. Полиномиально вычисляемые по времени и линейно вычисляемые по памяти алгоритмические вещественные числа и функции называются **FP//Linspace** алгоритмическими вещественными функциями. Множество **FP//Linspace** алгоритмических вещественных функций на отрезке $[a, b]$ обозначается с помощью **FP//Linspace** _{$C[a, b]$} .

1.3 Вычисление приближений вещественных функций

В данной работе используются следующие результаты из [9, 10, 11].

Для умножения a и b с точностью 2^{-n} , где a и b – вещественные числа такие, что $a \leq 2^p$ и $b \leq 2^p$ для некоторого натурального числа p , достаточно вычислять a и b с точностью 2^{-m} для $m = L_{a \cdot b}(n, p)$, где $L_{a \cdot b}$ – линейная функция натуральных аргументов.

Для вычисления величины, обратной к a с точностью 2^{-n} , где a – вещественное число такое, что $a \geq 2^{-p}$ для некоторого натурального числа p , достаточно вычислять a с точностью 2^{-m} для $m = L_{1/a}(n, p)$, где $L_{1/a}$ – линейная функция натуральных аргументов.

Для вычисления функции $(1+x)^h$ с точностью 2^{-n} , где вещественные $x \in [2^{-p} - 1, 2^p - 2]$ и вещественное число $|h| < 1$, достаточно вычислять x и h с точностью 2^{-m} для $m = L_{pow}^{(1)}(n, p)$, где $L_{pow}^{(1)}$ – линейная функция натуральных аргументов.

Можно показать, что для вычисления функции $(1+x)^h$ с точностью 2^{-n} , где вещественные $x \in [2^{-p_1} - 1, 2^{p_1} - 2]$ и вещественное $|h| < p_2$, достаточно вычислять h с точностью 2^{-m} для $m = L_{pow}^{(2)}(n, p_1, p_2)$, где $L_{pow}^{(2)}$ – линейная функция натуральных аргументов.

2 FP/LINSPACE вычисление функции $\zeta(s)$

Рассмотрим числовой ряд [7, 8] функции Римана $\zeta(s)$, сходящийся для всех комплексных чисел s (исключая $s = 1 + 2\pi i \frac{n}{\log_2(2)}$ для целых n):

$$\zeta(s) = \frac{1}{1-2^{1-s}} \sum_{k=0}^{\infty} \frac{1}{2^{k+1}} \sum_{q=0}^k (-1)^q \binom{k}{q} (q+1)^{-s}. \quad (1)$$

Пусть

$$f(q, s) = (q+1)^{-s},$$

$$g(k, q) = \binom{k}{q},$$

$$h(k, s) = \sum_{q=0}^k (-1)^q g(k, q) f(q, s),$$

$$u(s) = \sum_{k=0}^{\infty} \frac{1}{2^{k+1}} h(k, s), \quad \text{и}$$

$$v(s) = \frac{1}{1-2^{1-s}};$$

запишем ряд (1) следующим образом:

$$\zeta(s) = v(s)u(s). \quad (2)$$

Пусть p – натуральное число, $p \geq 1$; пусть $\lambda = \log_2(1+2^{-p})$ и s – натуральное число такое, что $1+\lambda \leq s \leq 2^p$. Будем вычислять $\zeta(s)$ по формуле (2) с точностью 2^{-n} для таких s , где n – натуральное число.

2.1 Вычисление $v(s)u(s)$

Так как

$$\begin{aligned} s &\geq 1+\lambda, \\ 2^s &\geq 2^{1+\lambda}, \\ 2^{s-1} &\geq 2^\lambda, \\ 2^{1-s} &\leq 2^{-\lambda}, \text{ и} \\ 1-2^{1-s} &\geq 1-2^{-\lambda}, \end{aligned} \quad (3)$$

выполняются соотношения

$$\begin{aligned} v(s) &= \frac{1}{1-2^{1-s}} \leq \frac{1}{1-2^{-\lambda}} = \\ &= \frac{2^\lambda}{2^\lambda - 1} = \frac{1+2^{-p}}{2^{-p}} = \\ &= 1+2^p < 2^{2p}. \end{aligned}$$

Следовательно

$$u(s) = (1-2^{1-s})\zeta(s) \leq C_1,$$

где C_1 – некоторая константа (следует из соотношений, изложенных в параграфе 2.2.3).

Это означает, что для вычисления $\zeta(s)$ по формуле (2) с точностью 2^{-n} достаточно вычислять величины $v(s)$ и $u(s)$ с точностью 2^{-n_1} , где $n_1 \geq n + C_2(p)$ ($C_2(p)$ – константа, зависящая от p).

Далее, для вычисления величины $v(s)$ с точностью 2^{-n_1} воспользуемся алгоритмом из [9, 10, 11] вычисления функции $(1+x)^h$ с точностью 2^{-n_1} ; при этом достаточно вычислять величину s до точности 2^{-m_1} для $m_1 = L_1(n_1, p)$, где L_1 – линейная функция натуральных аргументов (m_1 – некоторое натуральное число).

2.2 Вычисление $u(s)$

Будем вычислять функцию $u(s)$ следующим образом:

1) вычисляем $f(q, s)$ с точностью 2^{-n_4} ;

2) вычисляем $g(k, q)$ с точностью 2^{-n_4} ;

3) вычисляем

$$h(k, s)^* = \sum_{q=0}^k (-1)^q g(k, q)^* f(q, s)^*,$$

где $g(k, q)^*$ и $f(q, s)^*$ – приближенные значения величин $g(k, q)$ and $f(q, s)$ соответственно с точностью 2^{-n_4} ; пусть 2^{-n_3} – точность вычисления величины $h(k, s)^*$;

4) вычисляем

$$u_i(s)^* = \sum_{k=0}^i \frac{1}{2^{k+1}} h(k, s)^*;$$

пусть 2^{-n_2} – точность вычисления величины $u_i(s)^*$.

Для вычисления величины $f(q, s)$ с точностью 2^{-n_4} воспользуемся алгоритмом из [9, 10, 11] вычисления функции $(1+x)^h$ с точностью 2^{-n_4} ; для этого достаточно вычислять величину s с точностью 2^{-m_3} для $m_3 = L_3(n_4, p, \lceil \log_2(q) \rceil)$, где L_3 – это линейная функция натуральных аргументов (m_3 – натуральное число).

Следовательно, для вычисления величины $u_i(s)^*$ с точностью 2^{-n_2} с помощью суммирования ряда величину $h(k, s)^*$ достаточно вычислять с точностью 2^{-n_3} , где $n_3 \geq n_2 + 1$.

2.2.1 Вычисление $g(k, q)$

Запишем величину $g(k, q)^{-1}$ следующим образом:

$$g(k, q)^{-1} = \omega(k, q) = \prod_{\tau=1}^q b_{\tau} = \prod_{\tau=1}^q \frac{q - \tau + 1}{k - \tau + 1}.$$

Будем вычислять $\omega(k, q)$ в цикле по $\tau \in [1..(q-1)]$, на каждом шаге вычисляя

$$\omega(k, \tau)^* b_{\tau+1}^*$$

(при этом $\omega(k, 1)^* = b_1$), где $\omega(k, \tau)^*$ – приближенное значение величины $\omega(k, \tau)$ с точностью ε_{τ} , $b_{\tau+1}^*$ – приближенное значение величины $b_{\tau+1}$ с точностью ε_{τ} ; $\varepsilon_{\tau} = 2^{-m} < 2^{-1}$ для всех

натуральны m . Будем вычислять $\omega(k, \tau)^* b_{\tau+1}^*$ с точностью ε_τ , отбрасывая все биты после точки, начиная с q -го бита.

Используя метод математической индукции по $\tau \in [1..(q-1)]$, покажем, что $\varepsilon_\tau < 2^{-3q+2\tau}$ для каждого $\tau \in [1..(q-1)]$, если брать $\varepsilon_1 \leq 2^{-3q}$.

База индукции: $\tau = 1$; в этом случае, величина $\omega(k, 1)^*$, равная b_1^* , вычисляется с точностью $\varepsilon_1 \leq 2^{-3q+2}$. Индукционный переход: пусть $|\omega(k, \tau)^* - \omega(k, \tau)| \leq \varepsilon_\tau$ для $\tau \in [1..(q-2)]$. В этом случае

$$\begin{aligned}
 & |\omega(k, \tau+1)^* - \omega(k, \tau+1)| \leq \\
 & \leq |\omega(k, \tau)^* b_{\tau+1}^* - \omega(k, \tau) b_{\tau+1}| + \varepsilon_\tau = \\
 & = |\omega(k, \tau)^* b_{\tau+1}^* - \omega(k, \tau)^* b_{\tau+1} + \\
 & \quad + \omega(k, \tau)^* b_{\tau+1} - \omega(k, \tau) b_{\tau+1}| + \varepsilon_\tau \leq \\
 & \leq |\omega(k, \tau)^* (b_{\tau+1}^* - b_{\tau+1})| + \\
 & \quad + |b_{\tau+1} (\omega(k, \tau)^* - \omega(k, \tau))| + \varepsilon_\tau < \\
 & < \varepsilon_\tau + \varepsilon_\tau + \varepsilon_\tau < \\
 & < 4\varepsilon_\tau
 \end{aligned}$$

(здесь используется оценка $\omega(k, \tau) \leq 1$). Следовательно, выполняется оценка

$$\varepsilon_{\tau+1} < 2^{-3q+2\tau+2} = 2^{-3q+2(\tau+1)};$$

в частности, $\varepsilon_q < 2^{-q}$. Это означает, что достаточно брать $k = n_\omega + 1$ и $\varepsilon_1 \leq 2^{-3q}$ для вычисления $\omega(k, q)$ с точностью 2^{-n_ω} . И, далее, для вычисления $g(k, q) = \omega(k, q)^{-1}$ с точностью 2^{-n_4} достаточно брать $n_\omega = L_g(n_4)$ где L_g – линейная функция натурального аргумента, так как $\omega(k, q) \geq 2^{-k}$.

2.2.2 Вычисление $h(k, s)^*$

Будем вычислять $h(k, s)^* = \sum_{q=0}^k (-1)^q g(k, q)^* f(q, s)^*$ в цикле по $q \in [0..k]$. Так как

$$\begin{aligned}
 h(k, s)^* &= \sum_{q=0}^k (-1)^q g(k, q)^* f(q, s)^* = \\
 &= \sum_{q=0}^k (-1)^q (g(k, q) + \varepsilon_g)(f(q, s) + \varepsilon_f),
 \end{aligned}$$

где $\varepsilon_g \leq 2^{-n_4}$ и $\varepsilon_f \leq 2^{-n_4}$, выполняется соотношение

$$\begin{aligned} h(k, s)^* &= h(k, s) + \varepsilon_h = \\ &= h(k, s) + \sum_{q=0}^k (-1)^q (g(k, q)\varepsilon_f + f(q, s)\varepsilon_g + \varepsilon_g \varepsilon_f). \end{aligned}$$

Далее, из

$$g(k, q) \leq 2^k \quad \text{и} \quad f(q, s) \leq 1$$

следует

$$|\varepsilon_h| \leq (k+1)(2^k 2^{-n_4} + 2^{-n_4} + 2^{-2n_4}) < 2^{C_3 k + C_4} 2^{-n_4}.$$

Следовательно, если взять $n_4 = C_3 k + C_4 - n_3$ и вычислять $g(k, q)$ и $f(q, s)$ с точностью 2^{-n_4} , то точность вычисления $h(k, s)^*$ будет 2^{-n_3} .

2.2.3 Вычисление $u(s)$

Найдем точность вычисления of $u(s)$ используя следующие соотношения:

$$|\Delta(u; s)| = |u_i(s)^* - u(s)| \leq |u_i(s)^* - u_i(s)| + |u_i(s) - u(s)|$$

где

$$u_i(s)^* = \sum_{k=0}^i \frac{1}{2^{k+1}} h(k, s)^* \quad \text{и} \quad u_i(s) = \sum_{k=0}^i \frac{1}{2^{k+1}} h(k, s).$$

Так как

$$|u_i(s)^* - u_i(s)| \leq 2^{-n_2},$$

выполняется соотношение

$$|\Delta(u; s)| \leq 2^{-n_2} + |u_i(s) - u(s)|.$$

Оценим сверху величину

$$R_i(s) = |u_i(s) - u(s)| = (s) \sum_{k=i+1}^{\infty} \frac{1}{2^{k+1}} h(k, s).$$

Пусть k – нечетное натуральное число и $k' = (k \bmod 2) + 1$; пусть

$$D_{k,q}(s) = g(k, q)f(q, s) - g(k, k-q)f(k-q, s).$$

Так как $g(k, q) = g(k, k-q)$, выполняются равенства

$$\begin{aligned} |D_{k,q}(s)| &= |g(k, q)f(q, s) - g(k, k-q)f(k-q, s)| = \\ &= \left| g(k, q) \left((q+1)^{-s} - (k-q+1)^{-s} \right) \right| = \end{aligned}$$

$$= |g(k, q) d_{k,q}(s)|.$$

Следовательно,

$$(q+1)^{-s} \leq 1, \quad (k-q+1)^{-s} \leq 1 \quad \text{и} \quad g(k, q) < 2^k,$$

из чего получаем

$$|d_{k,q}(s, s)| < 1 \quad \text{и} \quad |h(k, s)| \leq \sum_{q=0}^{k'} |D_{k,q}(s)| \leq \frac{k+1}{2} 2^{k'}.$$

(те же самые соотношения выполняются в случае, когда k – четное). Следовательно

$$R_t(s) = v(s) \sum_{k=t+1}^{\infty} \frac{1}{2^{k+1}} \cdot \frac{k+1}{2} 2^{k'} \leq C_5 2^{2p} 2^{-2^{-1}t}.$$

Как результат, если взять $t = 4p + 2n_2 + C_6$, то

$$|\Delta(u; s)| \leq 2^{-n_2} + C_5 2^{2p} 2^{-2^{-1}(4p+2n_2+C_6)} \leq 2^{-n_2} + 2^{-n_2} = 2^{-n_2+1}.$$

Это означает, что достаточно взять $n_2 = n_2 + 1$ и $t = 4p + 2n_2 + C_6$ для вычисления $u(s)$ с точностью 2^{-n_1} .

3 Основной результат

Как результат, получаем, что для вычисления $\zeta(s)$ с точностью 2^{-n} для $s \in [1+\lambda, 2^p]$ достаточно выполнить следующие шаги:

- 1) вычислить s с точностью 2^{-m} для $m = L_s(n, p)$ где L_s – линейная функция натуральных аргументов, и
- 2) использовать $L_{prod}(n, p)$ умножений двоично-рациональных чисел длины $L_s(n, p)$, где L_{prod} – линейная функция натуральных аргументов.

Следовательно выполняется следующая теорема.

Теорема 1. *Вещественная функция Римана $\zeta(s)$ принадлежит классу $\mathbf{FP//Linspace}_{C[a,b]}$ для любого интервала $[a, b]$, где $a = 1+\lambda$, $b = 2^p$, $\lambda = \log_2(1+2^{-p})$ и p – натуральное число, $p \geq 1$.*

Также выполняется следующая теорема (учитывая параграф 1.3).

Теорема 2. *Вещественная функция Римана $\zeta(s)$ является экспоненциально вычислимой по времени и экспоненциально вычислимой по памяти по p для любого отрезка $[a, b]$, где $a = 1+\lambda$, $b = 2^p$, $\lambda = \log_2(1+2^{-p})$ и p – натуральное число, $p \geq 1$.*

Список литературы

1. Ko K. Complexity Theory of Real Functions. Boston: Birkhauser, 1991. 309 p.
2. Du D., Ko K. Theory of Computational Complexity. New York: John Wiley & Sons, 2000. 491 p.
3. Edwards H. M. Riemann's Zeta Function. Dover Publications, 2001. 330 p.
4. Карацуба Е.А. Быстрое вычисление дзета-функции Римана $\zeta(s)$ при целых значениях аргумента s . Пробл. передачи информ., 31: 4 (1995), С. 69–80.
5. Карацуба Е.А. Быстрое вычисление значений дзета-функции Гурвица и L-рядов Дирихле, Пробл. передачи информ., 34: 4 (1998), С. 62–75.
6. Cheng H., Gergel B., Kim E., Zima E. Space-efficient evaluation of hypergeometric series // ACM SIGSAM Bull. Communications in Computer Algebra, 2005. Vol. 39, No. 2. P. 41–52.
7. Hasse H. Ein Summierungsverfahren fur die Riemannsche ζ -Reihe. Math. Z., 1930. 32: P. 458–464.
8. Sondow J. Analytic continuation of Riemann's zeta function and values at negative integers via Euler's transformation of series // Proc. Amer. Math. Soc., 1994. 120 (120): P. 421–424.
9. Яхонтов С.В. FLINSPACE конструктивная функция $(1+x)^h$. // Вестник Самарского государственного технического университета. Серия «Физико-математические науки». Вып.2 (17), 2008. С. 239–249.
10. Яхонтов С.В. FLINSPACE конструктивные вещественные числа и функции. LAMBERT Academic Publishing, 2010. 176 с.
11. Яхонтов С.В., Косовский Н.К., Косовская Т.М. Эффективные по времени и памяти алгоритмические приближения чисел и функций, 2012. Учебное пособие. СПб.: Изд-во С.-Петербур. ун-та, 2012. 256 с.
12. Muller N. Th. Polynomial Time Computation of Taylor Series // Proc. 22 JAIOO–PANEL '93, Part 2, Buenos Aires, 1993. P. 259–281.
13. Higham N. J. Accuracy and Stability of Numerical Algorithms. Society of Industrial and Applied Mathematics, Philadelphia, 1996. 688 p.
14. FP//LINSPACE computability of real Riemann zeta function $\zeta(s)$ in Ko-Friedman model //И 34. Научная дискуссия: вопросы математики, физики, химии, биологии. № 8 (19): сборник статей по материалам XX международной заочной научно-практической конференции. М.: Изд. «Международный центр науки и образования», 2014 (август). С. 68–71.

Рецензенты:

Косовский Н.К., д.ф.-м.н., профессор, зав. кафедрой информатики, математико-механический факультет, Санкт-Петербургский государственный университет (Министерство образования и науки Российской Федерации), г. Санкт-Петербург.

Ермаков С.М., д.ф.-м.н., профессор, зав. кафедрой статистического моделирования, математико-механический факультет, Санкт-Петербургский государственный университет (Министерство образования и науки Российской Федерации), г. Санкт-Петербург.