УДК 004.043

Гамаюн В.П., д-р техн. наук,
Журавель С.В.,
Бандура О.Ю.,
Горенко С.О.,
Єфременюк Д.А.

ТЕОРЕТИКО-АРИФМЕТИЧНІ ОСНОВИ РОЗРЯДНО-ЛОГАРИФМІЧНОГО ПРЕДСТАВЛЕННЯ ЧИСЕЛ

Інститут комп'ютерних технологій Національного авіаційного університету

Розглянуті теоретичні засади побудови розрядно-логарифмічної арифметики, яка забезпечує комп'ютерну обробку даних великого діапазону без округлення та нормалізації.

Вступ

Арифметика чисел великого діапазону є актуальною для розв'язання задач, які чутливі до похибок, високоточних розрахунків у високотехнологічних галузях, системах захисту інформації тощо. Розрядно-логарифмічне представлення чисел використається для побудови комп'ютерної арифметики даних великого діапазону. Відмінною ознакою такого представлення є те, що така арифметики забезпечує високоточну обробку та не потребує нової технологічної бази для реалізації.

Конструктивне введення в системи числення.

Система числення – це спосіб представлення, позначення (запису) чисел, що визначається набором правил, формул [1-4].

Визначення 1. Нехай $A=\{a_1\ a_2\ a_3\ a_4\ a_5\ ...\ a_n\ \}$– кінчена множина, що має назву алфавіт. Слово q в такому алфавіті визначимо як кінцеву послідовність символів із A

$q= a_{i1}\ a_{i2}\ a_{i3}\ a_{i4}\ a_{i5}\ ...\ a_{in}$.

Кількість символів(букв), що складають таке слово, дамо назву довжини слова та позначимо як $l\ (q)= (a_{i1}\ a_{i2}\ a_{i3}\ a_{i4}\ a_{i5}\ ...\ a_{in})=k$.

Клас всіх кінцевих слів у алфавіті A позначимо як A^*. Слово з нульовою довжиною позначимо як Λ, яке також входить до A^*.

На множині слів A^* можливо задати рядок операцій, що є побудовою деякої алгебри. Наприклад, можливо використати операцію конкатенації - зчеплення або приписання слів або ланцюгів слів. Якщо α та β два слова , то операція конкатенації дає такий результат - нове слово $\gamma=\alpha\beta$:

$\alpha = (a_1 a_2 a_3 ... a_к)$

$\beta = (в_1\ в_2\ в_3\ в_4\ в_5\ в_6\ в_7....в_м)$

$\gamma=\alpha\beta = a_1 a_2 a_3 ... a_к\ в_1\ в_2\ в_3\ в_4\ в_5\ в_6\ в_7....в_м$.

Відзначимо, що приписання пустого слова не змінює цього слова.

Операція конкатенації є асоціативною, застосування такої операції до двох множин ланцюгів, наприклад двох мов L_1 та L_2 , є множина ланцюгів, що формується попарним приписанням елементів L_1 и L_2 з побудовою нової мови :

$$L_1 L_2 = \{a_1 a_2\ a_1 \mid \in L_1\ , a_2 \in L_2\ \}$$

Алфавіт з кінцевою кількістю слів може бути використаний для побудови системи числення. Загальна схема побудови наступна:

1. Вибирається деяка числова система $A=\langle \mathrm{A}, \Omega \rangle$;

2. Визначається кінцевий алфавіт A^*

3. Задамо деяке відображення β - зображення (представлення), що приписує словам з A^* відповідне числове значення з A:

$\beta : A^* \ \rightarrow \ A,$

слово $\rightarrow \alpha \rightarrow \beta(\alpha) \leftarrow$ числове значення слова.

Визначимо, що образ усіх можливих значень слів з A^* , котрі можливо отримати при застосуванні відображення β, складають діапазон представлення $D=\beta\ (\ A^*)$ для чисел, які мають хоча б одне зображення (представлення) в A^*. При цьому , якщо

$X = \beta(\alpha)$, то слово α називають зображенням числа X. Треба відмітити , що функція β не обов'язково всюди визначена - деякім словам неможливо приписати числові значення та різні слова з A^* не обов'язково зображують різні числа (функція β не є ін'єктивною).

Таким чином, побудована система числення має область визначення – кодовий діапазон $D^* =$ dom β та числовий діапазон $D^{**} = Im\ \beta \in A$.

Нехай $N^* = \langle N, \Omega \rangle$ система натуральних чисел та є алфавіт з однієї букви $A = \{1\}$, а також слова A мають вигляд

Λ, 1111...111.

Задамо функцію представлення в A наступним чином образом:

$\beta : A \rightarrow N$, $\beta(\alpha) = l(\alpha) = \alpha$ — довжина слова;

$\beta(\Lambda) = 0$.

Таким чином, функція β просто підраховує кількість букв у слові та повертає його довжину як натуральне число.

Операційний етап побудови систем числення передбачає завдання арифметичних операцій над словами (кодами) таким чином, щоб виконувалась відповідність операціям нал числовими значеннями. Так відповідність реалізується при гомоморфізмі алгебраїчних систем.

Визначення 2. Нехай $A^*_1 = \langle A_1, \Omega \rangle$ та $A^*_2 = \langle A_2, \Omega \rangle$ — алгебраїчні системи з однаковою сигнатурою. Назвемо гомоморфізмом системи у систему h таке відображення їх носіїв $h : A_1 \rightarrow A_2$, при якому операції та відношення не змінюються.

Іншими словами для любої операції $\omega \in \Omega_o$ та для любого набору

$a = ({}_1\ a\ a_2\ a_3\ a_4\ a_5\ ..\ a_n) \in dom\,\omega$ маємо

$h\ (\omega\ (a_1\ a_2\ a_3\ a_4\ a_5\ ..\ a_n\ ) = \omega\ (h(a_1),h(a_2),h(a_3),h(a_4),\ ..\ h(a_n\ ))$.

Для кожного співвідношення з $p \in \Omega_r$ повинна виконуватись умова: для $a = (a_1\ a_2\ a_3\ a_4\ a_5\ ..\ a_n) \in dom\ p$ має місце

$h(\ p(a_1\ a_2\ a_3\ a_4\ a_5\ ..\ a_n\ )) = p((h(a_1),h(a_2),h(a_3),h(a_4),\ ..\ h(a_n\ ))$.

Наприклад для арифметичної сигнатури повинні виконуватись співвідношення

$$h(x+y) = h(x)+h(y);$$

$$h(x\text{-}y) = h(x)\text{-}h(y);$$

$$h(x*y) = h(x)*h(y);$$

$$h(x/y) = h(x)/h(y).$$

Гомоморфізм визначає відповідності у операціях та співвідношеннях різних алгебраїчних систем. У системі числення з носієм A^* арифметичні операції та співвідношення визначаються функцією β з урахуванням гомоморфізму. Визначив множину A^* ооператціями та співвідношеннями з арифметичної сигнатури, побудуємо числову систему $A^* = \langle \mathrm{A}^*, \Omega \rangle$, яка представляє вихідні систему $A = \langle \mathrm{A}, \Omega \rangle$ за допомогою гомоморфізму представлення β:

$\beta :\ A \rightarrow A^*$.

Обидві системи числення, вихідна A та, що отримали A^*, є числові системи. Однак необхідно відмітити різницю між ними. Вихідну числову систему ми розглядали як абстрактну, без аналізу її елементів. Існування такої системи та її властивості можуть бути описані аксіоматично та розглядатися як абстрактна структура даних. Структура системи числення A^* визначена однозначно як система зі словами з кінцевого алфавіту. Об'єкти цієї системи конструктивні та операції сигнатури (арифметичного типу) можуть бути реалізовані через алгоритми над такими об'єктами (словами). Робота кожного алгоритму складається з ряду елементарних дій, що реалізують елементарну операцію - мікро операцію, котра може оперувати не з цілими словами, а з буквами(цифрами). Все це дозволяє визначити систему числення як конструктивну числову систему, що представляє абстрактну систему числення, що задана.

Для системи числення з прикладу застосування конкатенації визначення арифметичних операцій над кодами чисел співпадає з функцією представлення (конкатенації). Наприклад додавання двох слів:

$\beta\ (a_1+a_2) = \beta(a_1) + \beta(a_2)$;

$a_1 = 111$; $a_2 = 11111$; $a_1 + a_2 = 11111111$.

Операція множення виконується як операція багатократного додавання

(м= $\beta\ (a_2)$ раз)

$a_1 * a_2 = a_1\, a_1\, a_1\, a_1\, a_1\, a_1\, a_1 \ldots\ldots a_м$

Позиційні системи числення.

Для опису позиційних систем числення застосовується система натуральних чисел:

$N = \{N, +, —, *, / ; = , \leq \}$.

На підставі такої системи побудова позиційних систем може виконуватись наступним чином [5-7].

Визначаємо вихідний алфавіт A, елементи котрого назвемо цифрами. Кожна цифра має деяке зображення та визначене цифрове значення, що є результатом застосування функції C:

$C: A \rightarrow N$,

цифра $\rightarrow$ $C(a)$ числове значення.

Маємо також розрядну функцію δ : $N \rightarrow N$, значення якої має назву ваги деякого розряду $\delta\ (i) = S_i$. Відповідно функція представлення β визначається як

$$\beta(\, a_n\, a_{n-1}\, a_{n-2} \,.\,.\,.\,.\,.\, a_1\ a_0) = \sum_{i=0}^{n-1} C(a_i)\ \delta_i$$

Позиційні системи числення з основою S, що задається, будуються шляхом завдання алфавіту $A = Z_S = \{0,1,2,3,4,\ldots.S\text{-}1\}$ та розрядної функції δ (i)= S_i , i= 0,1,2,... .

Функція, що представляє S–ічну систему числення (систему числення з основою S) позначимо β_S

$\beta_S\ :\ Z^*_S \rightarrow N.$

Область зображення(представлення) S – ічної системи числення співпадає з класом всіх натуральних чисел $\beta_S(\,Z^*_S\,) = N$. Представлення слів за допомогою β_S не є однозначним, бо $\beta_S(00000a) = \beta_S(a)$. Однак, якщо через Z^{**}_S позначити клас слів які не починаються з нуля, то націй множині функція представлення взаємно-однозначна, тобто кожне слово має тільки одне зображення та кожному набору символів відповідає одне числове значення. Операціям у S–ічній системі задаються відповідними алгоритмами.

За допомогою розширення алфавіту Z_S – приєднання до нього двох знакових символів "+" та "–", можливо побудувати систему числення для кільця цілих чисел. Кодовий діапазон, область визначення для такої системи визначається наступною множиною

$\{+,-\} \times Z_S = \{uw \mid u \in \{+,-\},\ w \in Z_S\}$,

в якій слова (елементи) починаються зі знакового символу.

За аналогічним розширенням можливо, якщо використати S–ічну кому, побудувати систему, що представлятиме S-ічні раціональні дійсних чисел. Алфавіт будується з додаванням символу крапки ".".

Ієрархічні системи числення.

Принцип побудови ієрархічних систем числення полягає в наступному [8,9]. Вибирається деяка система числення $A = \langle A, \Omega \rangle$, потім цифри цієї системи (елементи алфавіту А) зображуються у вигляді слів (кодів) другої системи числення $B = \langle B, \Omega \rangle$ таким чином, щоб *числові значення кодів* співпадали.

Нехай задана вагова функція μ , що виконує перетворення кожної цифри з А у числове значення A^*

$\mu : A \rightarrow A^*$.

У системі числення B числова область така ж як у системі A. Тоді кожна цифра $z \in A$ має зображення у системі B як $h(z) = z^* \in B^*$. При цьому, якщо β - функція перетворення кожної цифри з В у числове значення, то виконується наступне співвідношення

$\beta\ (z^*) = \beta\ (h(z)) = \mu\ (z)$,

яке означає, що числове значення цифри z у системі числення A співпадає з числовим значенням слова z^*, що представляє цю цифру в системі числення *B*. На рис.1 зображена діаграма представлення числа у різних системах числення

$$\begin{tikzcd} A \arrow[r, "\mu"] \arrow[d, "h"'] & \mathrm{A} \\ B^* \arrow[ur, "\beta"'] & \end{tikzcd}$$

Рис.1 Діаграма побудови ієрархічних систем

де μ - вагова функція, *A*- цифра, А- числове значення

Кодування чисел в ієрархічній системі числення виконується як поцифрове кодування, іншими словами при кодуванні цифр задається словарний гомоморфізм

h: $A \rightarrow B^*$;

$a=(z^*_1, z^*_2, z^*_3, z^*_4, z^*_5, \dots z^*_k) \rightarrow h^*(a)= h^*(h(z_1), h(z_2), h(z_3), h(z_4), h(z_5), \dots h(z_k))$.

Іншими словами гомоморфізм *h* можливо продовжувати завдане число разів, яке дорівнює значенню ступіні вкладеності ієрархічної системи. Ступінь вкладеності ієрархічної системи визначається як кількість внутрішніх систем числення.

Отриману систему числення позначимо як *A*[*B*], де *A* є зовнішня система числення, *B* – внутрішня система числення

Розглянемо приклад. Нехай *A* – десяткова система числення, а *B* – двійкова система числення. Алфавіти таких систем відповідно визначені як

$A = \{0,1,2,3,4,5,6,7,8,9\}$, $B = \{0,1\}$.

Цифрове кодування цифр системи *A* виконується тетрадами

$0 \xrightarrow{h} 0000$

$1 \xrightarrow{h} 0001$

$2 \xrightarrow{h} 0010$

$3 \xrightarrow{h} 0011$

....

$9 \xrightarrow{h} 1001$

і в результаті отримаємо двійково-десяткову систему числення, що застосовується у комп'ютерній техніці.

Ступінь вкладеності ієрархічно системи може бути більше одиниці. Можливо створення ієрархічних систем числення з *k*-тою ступеню вкладеності (*k* більше одиниці)

$A_0\ [A_1\ [A_2\ [A_3\ ......[A_k\ ].....]]]...]$

з різними основами $S_0\ S_1\ S_2\ S_3\ S_4\ ...Sk$ та різними функціями представлення $\beta_0 \beta_1 \beta_2 \beta_3 \beta_4 .... \beta_k$.

Для побудови та застосування різноманітних засобів комп'ютерної техніки використовують ієрархічні системи числення [7,8,9]. Наприклад код "2, 4, 2, 1" та код "+3" визначаються як коди з самодоповненням – у таких кодах доповнення до 9 реалізується шляхом заміни 1 на 0 та 0 на 1. У коді " 7, 4, 2, 1" використовують мінімальну кількість розрядів с цифрою 1 (одиницею). Близьким до властивостей попереднього коду є код „5, 4, 2, 1". Код „ω, *x*, *y*, *z*" використовується для спрощення мультиплікативних операцій – молодший розряд добутку десяткової цифри на 3, 7, 9 можливо отримати перестановкою по колу коду ω,*x*,*y*,*z*: для першого варіанту маємо код добутку *z*,ω,*x*, *y*; для другого - *x*,*y*,*z*,ω; для третього - *y*, *z*,ω, *x*. Код *A*,*B*,*C*,*D* має максимальну вірогідність виправлення помилок. Такі ж властивості є у рефлексному коді, в якому для сусідніх комбінації відрізняються тільки на одиницю.

Двійково-п'ятирковий код є позиційним кодом з постійною кількістю символів у комбінаціях, що використовується у системах контролю. Такі ж властивості є у кода "2 з 5". Наведені коди є прикладами застосування ієрархічних систем у комп'ютерних засобах.

Розрядно-логарифмічна система числення.

Сучасний стан представлення (зображення) кодування даних у комп'ютерному середовищі може характеризуватися як невідповідний між теоретичними результатами та їх практичним використанням. Якщо у теоретичному плані отримано цілий ряд дуже ефективних розв'язань, то на практиці застосується невелика кількість кодів.

Системи числення визначають структури даних, що використовуються у ЕОМ для обробки інформації. Саме різні показники, що визначають ефективність процесу обробки в цілому, складають граничні умови застосування деякої системи числення.

Наприклад складність алгоритмів операцій обробки, технологічні проблеми реалізації обумовили застосування двійкової системи числення яке є позиційною системою.

Однією з основних тенденцій у розвитку засобів обчислювальної техніки є застосування нетрадиційного кодування-зображення даних. Наприклад, незважаючи на принципові відмінності між непозиційним та позиційним кодуванням доведена можливість об'єднання цих систем в таку, якій властиві позитивні ознаки складових [10-12]. Досліджуючи можливості застосування нетрадиційних кодів для реалізації мультиплікативних операцій, а також загальні вимоги до конструктивності кодування даних (можливості виконання операцій) було запропоновано оригінальне кодування даних, що отримало назву *розрядно-логарифмічне* [10-12].

Визначення 3. Розрядно-логарифмічним кодуванням даних називають зображення двійкового операнду у вигляді набору двійкових кодів ненульових розрядів $\{Nx_i\}$ того ж операнду, кожний з котрих визначається як результат обчислення логарифму від ваги цього розряду:

$$x_i \longrightarrow \{ Nx_i = log_2 x_i * p^i \mid x_i \neq 0\},$$

де x_i — ненульовий розряд, p – основа системи числення.

Приведемо основні положення РЛ представлення даних [10-12]. Відомо, що будь-яке позитивне число A може бути представлене у вигляді скінченного або нескінченного дробу [5,7]:

$$A = a_m * p^m + a_{m-1} * p^{m-1} + a_{m-2} * p^{m-2} + \ldots$$

$$+ a_1 * p^1 + a_0 * p^0 + a_{-1} * p^{-1} \ldots$$

або $$A = \sum_{i=-\infty}^{m} a_i * p^i ,$$

де a_i – цифри числа A (набір цифр залежить від основи системи числення), p – основа системи числення, a_m– старший значущий розряд числа A. При позиційному зображенні число A записується у вигляді:

$$a_m a_{m-1} \grave{a}_{m-2} \ldots a_1 a_0 a_{-1} \ldots$$

При РЛ представленні кожен двійковий розряд a_j, який не дорівнює нулеві, представляється як код N_i, який дорівнює двійковому логарифмові від кількості, представленої цим розрядом:

$$N_i = \log_2 a_i 2^i = i \ (a_i \neq 0).$$

Іншими словами, N_i – це номер (код) ненульового розряду. Число A буде записане у вигляді масиву двійкових кодів - $N_1 N_2 N_3 \ldots N_i \ldots$. Наприклад, двійкове число з фіксованою комою

$$\grave{A} = \overset{12}{1}\overset{11}{0}\overset{10}{0}\overset{9}{0}\overset{8}{1}\overset{7}{1}\overset{6}{0}\overset{5}{1}\overset{4}{0}\overset{3}{0}\overset{2}{1}\overset{1}{0}\overset{0}{1}.\overset{-1}{1}\overset{-2}{0}\overset{-3}{1}$$

в розрядно-логарифмічному представленні буде мати такий вигляд:

$$A = 12.8.7.5.2.0.\text{-}1.\text{-}3.$$

Найбільш зручною для використання розрядно-логарифмічного зображення з практичної точки зору є структура, у якій для числа A буде зазначений знак цього числа, кількість значущих одиниць у РЛ представленні і набір РЛ кодів. Загальний вигляд такої структури наступний:

$$A \rightarrow \text{sign A } Q_A N_1 N_2 N_3 \ N_4 \ldots N_Q ,$$

де *signA* – поле знака числа A (0 - позитивне число або 1 - негативне), Q_A - поле кількості значущих одиниць, $N_1 N_2 N_3 \ldots N_{Q_A}$ - поле РЛ кодів. Для приведеного вище приклада число A буде записано як $A_{РЛ}$=0.8.12.8.7.5.2.0.-1.-3.

Перехід від двійкової форми подання A до форми, де кожний ненульовий розряд подано своїм номером, визначається однозначно і реалізується без додаткових функціональних перетворень – виконується операція підстановки по кожному ненульовому розряду.

Для зворотного переходу у двійкове подання (двійковий код) треба застосувати операцію дешифрування.

Моделювання інформаційних потоків при реалізації розрядно-логарифмічного представлення даних в ЕОМ показує, що реальні числа подаються у вигляді масиву з числом елементів від 5-7 до 15-20.

Застосування РЛ зображення дозволяє побудувати єдину структуру для плаваючої коми та фіксованою коми. Для плаваючою коми можна збільшити розрядність зображеня кожного номеру ненульового розряду на один біт та додати до кожного номеру значення порядка, якщо порядок виражається показником ступеня двійки.

Отже, структура узагальненого зображення для плаваючої коми та фіксованої коми має вигляд:

$$signA\text{ -}\qquad Q\qquad sign\,N_{1\pm p}\qquad sign\,N_{2\pm p}\qquad sign\,N_{3\pm p}\qquad \ldots..\qquad sign\,N_{Q\pm p}$$

де p – порядок.

Множина чисел, зображених у розрядно-логаріфмичному коді, РЛ має наступні властивості:

- розрядно-логарифмічне кодування (зображення) є обмежена підмножина раціональних чисел Q;

- множина РЛ кодів є симетричною відносно числа нуль;

- елементи РЛ розподілені рівномірно на дійсній прямій – важливий чинник підвищення точності зображення даних;

Треба окремо визначити *зображеня числа нуль* при розрядно-логарифмічному кодуванні. У структурі РЛ використається показник значущих розрядів у кожному операнді Q. Якщо операнд нульовий (тобто нуль), то такий показник Q дорівнює нулю Q=0 – операнд не має значущих розрядів. Таке представлення нуля у комп'ютерному середовищі є найбільш точним, бо відповідає фізичному трактуванню нульового значення, а не математичній моделі, яка прийнята у різних ЕОМ. Таким чином кожне розрядно-логарифмічне зображення числа є одномірний масив. Доцільно, щоб тип даних елементів такого масиву був „ЦІЛЕ", так як номера позицій ненульових розрядів є цілими числами.

Розрядно-логарифмічне кодування (зображення) є ієрархічною системою з алфавітом (носієм), що співпадає з алфавітом двійкової системи. Функція відображення при цьому є логарифмічною функцією, областю визначення якою є ненульові розряди операндів

$$\beta\ S\ :\ A_2\quad \rightarrow A^{\text{рл}},\quad \beta\ S = log_2\, a_i p^i\,.$$

Числовий діапазон при розрядно-логарифмічному представленні (кодуванні) значно розширюється (діапазон представлення чисел збільшується на кілька порядків у відповідності до розмірів розрядної сітки)[10-12]. Це пов'язано з тим, що в розрядній сітці комп'ютера буде представлятися не саме число A, а лише елементи N_i, які кількісно збігаються (рівні) степеням двійки значущих одиниць у двійковій формі числа A. Зазначені елементи N_i складають РЛ форму числа, а для їхнього представлення використовується машинна розрядна сітка.

Так при традиційному підході при n-розрядній сітці максимальне ціле знакове число можна записати як:

$$A_{max} = 2^{n\text{-}1}\ \text{-}1,$$

а при РЛ кодуванні:

$$A_{max\,RL} = 2^{mm}\ \text{-}1,\ \text{де}\ mm{=}2^{n\text{-}1}\ \text{-}1$$

(один розряд виділяється для знака двійкового числа).

У табл. 1.1 наведені діапазони двійкового представлення чисел, діапазони РЛ представлення, розрядності РЛ чисел при базовій розрядності комп'ютера. У результаті розширення діапазону представлення даних скорочується вплив на точність обчислень процедурою округлення.

Таблиця 1.1.

Порівняльна характеристика двійкового та РЛ представлень

Розрядність двійкового коду	Діапазон двійкових чисел	Розрядність РЛ коду	Діапазон РЛ чисел
4	$2^{+3} \le \grave{A} \le 2^{-3}$	7	$2^{+7} \le \grave{A} \le 2^{-7}$
8	$2^{+7} \le \grave{A} \le 2^{-7}$	127	$2^{+127} \le A \le 2^{-127}$
16	$2^{+15} \le \grave{A} \le 2^{-15}$	32767	$2^{+32767} \le A \le 2^{-32767}$
32	$2^{+31} \le \grave{A} \le 2^{-31}$	2147483647	$2^{+2147483647} \le A \le 2^{-2147483647}$

Приведена порівняльна характеристика (табл. 1.1) демонструє важливу особливість розрядно-логарифмічного представлення чисел: не перебудовуючи архітектури комп'ютера з короткою розрядною сіткою, розрядно-логарифмічне кодування забезпечує моделювання роботи комп'ютера зі значно більшою розрядністю та, відповідно, більшим діапазоном числових даних і більшою точністю обчислень. Комп'ютери з малою розрядністю, в такий спосіб, можна перевести по показнику точності в клас суперкомп'ютерів.

Висновки

Реалізація великого числового діапазону з використанням розрядно-логарифмічного представлення даних забезпечує комплексну обробку великих та малих числових змінних. Тобто дослідник має можливість отримувати точні числові характеристики при моделюванні різних фізичних процесів не змінюючи модель обчислень для різних діапазонів даних.

Розрядно-логарифмічне кодування є розширенням двійкової арифметики, що дає можливість використати на новому якісному рівні бібліотеку алгоритмів, які вже були розробленні.

Теоретичний апарат побудови ієрархічних систем числення може бути застосований для розробки інших, спеціалізованих систем числення для розв'язання багатьох прикладних проблем.

Список літератури

1. *Куликов Л.Я.* Алгебра и теория чисел . - М.: Высшая школа, 1979.- 560 с.

2. *Амербаев В.М.* Теоретические основы машинной арифметики. - Алма-Ата: Наука, 1976.- 323с.

3. *Вивальнюк Л.М.* Числові системи.- К.: „Вища школа", 1977.- 184 с.

4. *Акушский И.А., Юдицкий Д.И.* Машинная арифметика в остаточных классах. - М.: „ Советское радио", 1968.- 444 с.

5. *Карцев М.А.* Арифметика цифровых машин. – М.: Наука, 1969. - 576 с.

6. *Грегори Р., Кришнамурти Е.* Безошибочные вычисления. Методы и приложения: Пер. с англ. – М.: Мир, 1988. – 208 с.

7. Справочник по вычислительной технике// Под.ред. Малиновского Б.Н. - К.: „Техніка", 1974. - 512 с.

8. *Евстигнеев В.Г., Городенко Е.И., Сведе-Швец В.Н., Краснобаев В.А.* Функциональные преобразования для систем остаточных классов// Радиотехника.- 1984.- Вып.69/84. Харьков, Вища школа.- С.57-59.

9. *Евстигнеев В.Г.* Позиционно-остаточная система счисления для быстродействующих ЭВМ. Научн. техн. сб. Вопросы оборонной техники , серия 6, 1984, вып.1 (41), -С.10 - 14.

10. *Гамаюн В.П.* Макрооператорные методы вычисления многоместных произведений// Микропроцессорные системы и их применение. - Киев: Ин-т кибернетики им. В.М. Глушкова АН УССР,1990.- С.23-28.

11. *Гамаюн В.П.* Повышение точности вычислений при разрядно-логарифмическом представлении данных// Кибернетика и системный анализ . – 1997. -№1. – С.21-37.

12. *Гамаюн В.П.* Организация вычислений при разрядно-логарифмическом представлении данных // УСиМ .- 1996. - №3. – С.3 -7.