УДК 681.3

ЗАГАЛЬНІ ПІДХОДИ ДО ПОБУДОВИ СИСТЕМ ЧИСЛЕННЯ В СУЧАСНИХ ЕЛЕКТРОННИХ СИСТЕМАХ

О.А Борисенко
Сумський державний університет

У даній роботі розглянуті основні принципи і ідеї позиційного кодування чисел і на їх основі позиційної лічби і побудови відповідних систем числення. Як основна ідея виступають розклади вихідної множини елементів на підмножини, а потім цих підмножин на інші підмножини до того часу, поки вони не стануть одноелементними. Наступна ідея, яка активно використовується при побудові позиційних систем числення, полягає в кодуванні кожної з отриманих підмножин того чи іншого розкладу на кожному кроці однією із цифр алфавіту, який дійсний тільки для даного розкладу. Самі алфавіти, як і кодування підмножин на їх основі, будуються за визначеними в роботі правилами. Важливою ідеєю в позиційних системах є також спосіб підсумовування елементів вихідних підмножин на кожному кроці розкладів з допомогою алфавітів і далі об'єднання всіх цих результатів в одну суму, яка і визначає кількісне значення числа, яке кодується. Реалізуються наведені ідеї з допомогою розкладів підмножин, взятих по їх кроках, і їх алфавітів, відношень належності між підмножинами з відповідними зв'язками, а також всіх значень кількостей елементів в підмножинах на всіх їх розкладах і всіх кроках розкладів. Все це в кінцевому підсумку створює позиційну систему числення, з допомогою якої можна зобразити будь-яке велике число. Таких систем числення побудовано вже багато, але, крім вже отриманих, багато можна знайти систем, які ще не створені, але мають добрі перспективи на ефективну практичну реалізацію.

ЗАГАЛЬНА ХАРАКТЕРИСТИКА СИСТЕМ ЧИСЛЕННЯ І ПОСТАВЛЕННЯ ЗАДАЧІ РОБОТИ

З числами і системами числення люди зустрічаються ще в дитинстві, коли починають навчатися лічбі, і продовжують з ними зустрічатися все життя. Під числом можна розуміти кількість елементів, які належать до еквівалентних множин, і тоді це буде його абстракція, а можна з допомогою системи числення розуміти зображення цих кількостей, і тоді це буде його кодована форма. Як же на разі це подається в матеріалі, який викладається, залежить від контексту, де знаходиться це число. Тому і сприйняття числа може бути різним - або як кількості елементів в множині, або як її зображення. В даній роботі поняття числа має теж два розуміння, і відповідно до них в залежності від контексту воно відповідно використовується в ній.

Сьогодні після появи таких простих і універсальних систем числення, як десяткова і подібних до неї в уяві більшості людей числа сприймаються як нероздільне ціле з системами числення. В дійсності ж між числом і системою числення спостерігається відмінність, яка проявляє себе в тому, що системи числення можуть бути різними, а от число як кількість, яка відображається в них, буде одним і тим самим, хоча при цьому може набирати в різних системах числення різних форм.

Системи числення по своїй суті матеріалізують число, кодуючи його, і тим самим надають йому ту чи іншу форму. Від цієї форми в багатьох випадках залежить ефективність їх використання при лічбі і при розв'язанні інших практичних задач з кодування і перетворення інформації в електронних системах.

Системою числення, або нумерацією, називається деяка сукупність правил, обмежень, знаків, за допомогою яких можна кодувати (формувати, зображувати) будь-яке натуральне число.

При кодуванні чисел системою числення в першу чергу розв'язуються, як правило дві задачі. Перша з них - це ефективне зображення за скінченне число кроків початкової частини невід'ємних чисел 0,1,2, ..., і друга, зворотна до неї, - перехід від цих зображень до самих чисел. В першій задачі при зображенні чисел розв'язуються також і деякі допоміжні задачі, наприклад, задача підняття стійкості зображених чисел до завад.

При розгляді другого питання розв'язується задача побудови кодових зображень чисел з мінімальною кількістю цифр. Такі зображення цифр мають ще назву номерів. Тому таке кодування чисел відоме як *нумераційне* кодування. Воно досить широко використовується як в теоретичних розділах математики, так і на практиці, наприклад, при стисненні інформації. Однак і в тому і в іншому випадку системи числення в загальному плані розв'язують задачу *кодування* чисел чи кількостей і тому теорію їх побудови слід віднести до теорії кодування.

Але специфікою систем числення є те, що, окрім вищеназваних задач кодування, вони повинні також виконувати над сформованими з їх допомогою числами арифметичні і логічні операції, що виходить за межі тільки задач кодування в класичному їх розумінні, і тоді вони більше підходять під задачі теорії чисел, хоча напряму і не є ними. Це призводить до необхідності розглядати загальну теорію систем числення як самостійну наукову дисципліну, тісно зв'язану як з теорією кодування інформації, так і з теорією чисел, в чому і проявляється її особливість як пограничної науки між цими двома теоріями. Хоча більше вона пов'язана все-таки з теорією кодування. Не виключено, що в перспективі наука про системи числення може стати самостійною наукою. Але для цього потрібно більш досконало розвивати її загальну теорію.

З точки зору кодування інформації всі системи числення умовно поділяються на два великих класи - *позиційні* і *непозиційні*. Умовно тому, що і ті і інші системи мають в своєму складі елементи відповідно непозиційних і позиційних систем числення, хоча історично першими виникли непозиційні системи числення. В їх фундаменті лежить кількісний підхід до визначення числа, в якому для кодування тих чи інших кількостей вигадувалися особливі знаки-числа. Кожному такому знаку відповідав кількісний еквівалент. Наприклад, в так званій римській нумерації знаку X відповідала кількість елементів множини, яка дорівнювала 10. В подальшому такими знаками – числами користувалися також і для одержання інших чисел. Так, якщо перед знаком X ставилась вертикальна риска, тобто формувався новий знак - $1X$, то це означало, що від десяти треба відняти одиницю, і результуюча кількість буде дорівнювати 9. Знаки, подібні до X, називаються вузловими. Вони широко використовувались в первісних непозиційних системах числення.

Кількість чисел, яку можна було одержати з допомогою непозиційного кодування, через його складність і відповідно велику кількість чисел, які треба було запам'ятовувати, була обмежена декількома сотнями і, крім того, над цими числами досить важко було виконувати арифметичні і логічні операції. Тому з'явилася потреба у більш ефективних системах числення, які б мали прості правила кодування чисел і тим самим не обмежували б їх кількість та легко виконували над ними арифметичні і логічні операції.

Такими системами чисел стали *позиційні* системи, в основі яких лежить позиційний принцип кодування чисел, тобто коли кількісні значення знаків, з яких вони складаються, залежать від їх позицій. Вони найбільш часто використовуються сьогодні на практиці. У своєму розвитку ці системи пройшли важкий шлях від найпростіших непозиційних систем через лічбу на пальцях до теперішніх позиційних систем числення. Їх поява дозволила розширити множину невід'ємних цілих чисел, і відповідно вони були однією із основ побудови сучасної теорії чисел [1].

На сьогодні інколи вважається, що основні питання, пов'язані з позиційним кодуванням чисел, вирішені, тому що на практиці наявні позиційні системи числення добре себе зарекомендували, і з цього виходить, що немає потреби розвивати теорію позиційного кодування чисел далі. Але такий підхід має свої недоліки, оскільки ігнорує подальший розвиток теорії позиційних систем числення, яка має в собі ще невикористаний на практиці позитивний потенціал. Цю думку підтверджує і те, що час від часу з'являються нові системи числення як позиційні, так і непозиційні з новими властивостями і можливостями, які можна використати в тих чи інших сферах науки і практики. Серед таких систем числення є системи з досить складною структурою, наприклад, системи залишкових класів, факторіальні, поліадичні, фібоначиєві і багато інших (2,3,4).

Так, вони не настільки економні з точки зору кількості елементів в їх структурах і простоти виконання ними арифметичних і логічних операцій, як, наприклад, десяткова, але ж вони мають свої позитивні специфічні властивості, які слід використовувати на практиці. Це і можливість генерації різних комбінаторних об'єктів, таких, наприклад, як перестановки або сполучення з самими різними обмеженнями на них, так і їх перешкодостійкість, і інші, більш специфічні властивості, які проявляються при конкретних застосуваннях. Наприклад, подібні системи числення застосовуються при стиску і захисту інформації. Крім того, в ряді специфічних випадків з допомогою таких систем числення можна підняти швидкодію обчислювальних пристроїв і систем до величин, які недосяжні при використанні звичайних систем числення.

Ці системи не є конкурентами традиційних систем числення, як інколи це вважається, хоча б тому, що традиційні системи за своєю універсальністю і простотою виконання арифметичних і логічних операцій перевищують будь-які інші системи числення з більш складною структурою. Однак системи числення з такою структурою в ряді випадків є ефективним доповненням простих систем, тому що можуть сумісно працювати при розв'язанні ними одних і тих самих задач, збільшуючи надійність чи швидкодію пристроїв і систем, які їх використовують.

Названі позитивні властивості складних позиційних систем числення призводять до необхідності подальшого дослідження в загальному плані методів їх побудови з метою розроблення таких систем для більш ефективного розв'язання спеціальних задач і використання їх на практиці. Це велика задача, яка не може бути розв'язана в одній праці, але в ній все ж можна розглянути ідеї чи засади, які лежать в основі її розв'язання.

Тому задачею даної роботи є задача вивчення ідей позиційного кодування інформації з метою їх використання в подальшому для розв'язання спеціальних задач позиційної лічби і побудови відповідних їм більш складних позиційних систем числення.

ПРАВИЛО СУМИ В ПОЗИЦІЙНИХ СИСТЕМАХ ЧИСЛЕННЯ

В основі будь-якої простої чи складної системи числення лежить та чи інша ідея лічби. В основі позиційної лічби, яка розглядається в даній праці, лежать послідовні *розклади* по *кроках* деякої скінченної вихідної множини Q з числом елементів, більшим одиниці, на не порожні *підмножини*. Ці кроки в системах числення розглядають як *розряди* відповідних їм чисел, а кількості чи числа елементів в підмножинах – як *ваги* їх цифр. Розклади йдуть крок за кроком до того останнього кроку, на якому буде отримана підмножина з *одним* елементом – *одноелементна* підмножина. Такі розклади в подальшому будемо називати *повними*. Розклади ж, які закінчуються підмножинами з більшою, ніж один, кількістю елементів, будемо називати *неповними*, або *частковими*.

Для побудови позиційних систем числення важливо, по-перше, щоб рано чи пізно при проведенні розкладів по кроках були отримані в кінцевому підсумку повні розклади і, по-друге, щоб таких повних розкладів і відповідних їм одноелементних підмножин було стільки, скільки елементів є в вихідній множині. В той самий час очевидно, що неповних розкладів буде завжди менше, ніж повних.

Ці дві властивості, а також інші, не менш важливі для побудови таких систем числення властивості позиційного кодування чисел можуть бути отримані на основі правила *суми*. Воно має такий вигляд:

Якщо деяку скінченну множину елементів розкласти на декілька підмножин, які не перетинаються між собою, то сума їх елементів буде дорівнювати числу елементів, які були в цій множині до її розкладу.

Це правило через його очевидність не викликає сумніву і в неявному вигляді широко використовується на практиці як аксіома. Але для даної роботи був сенс сформулювати його у вигляді правила чи принципу, щоб потім усвідомлено використовувати його для розв'язання задач, які в ній ставляться, - *вивчення ідей побудови позиційних чисел і позиційної лічби на їх основі.*

Правило суми є загальним. Воно поширюється на розклад будь-яких множин, в тому числі й на розклад порожньої множини на порожні множини, але для даної роботи важливим є випадок, коли при розкладі скінченної множини на підмножини порожні підмножини будуть відсутні. Далі всюди, коли буде йти мова про розклади множин чи підмножин на нові підмножини, якщо не буде спеціальних застережень, будемо вважати, що і множини, і підмножини не є порожніми.

З цього правила можна отримати деякі важливі для побудови позиційних систем числення **наслідки**:

1 Розклад вихідної множини на одну підмножину чи саму себе призводить до кількості елементів в цій підмножині, яка дорівнює кількості елементів у вихідній множині.

Дійсно, після такого розкладу, в якому є тільки одна підмножина, згідно з правилом суми сумарна кількість елементів в отриманій підмножині повинна дорівнювати кількості елементів у вихідній множині, що і потрібно було довести.

2 Розклад вихідної множини на число підмножин, яке дорівнює кількості елементів в цій множині, приводить до того, що кожна з отриманих підмножин буде містити в собі рівно один елемент.

Дійсно, якщо виходити з правила суми, то сумарна кількість елементів у всіх підмножинах повинна дорівнювати кількості елементів у вихідній підмножині. Якщо при цьому хоча б в одній з підмножин, отриманих після розкладу, кількість елементів збільшиться, тобто буде більше одного елемента, то щоб зазначена рівність не змінилася, потрібно буде одну з підмножин зробити порожньою, що рівнозначно зменшенню кількості підмножин, на які відбувається розклад. Але це заборонено вихідною умовою. Залишається одне – вважати, що в кожній з підмножин знаходиться один елемент, що і доводить даний наслідок.

3 Розклад вихідної множини елементів на кількість не порожніх підмножин, яка буде менша кількості вихідних елементів цієї множини, призведе до того, що хоча б в одній з цих підмножин буде отримана *кількість елементів, яка буде більша одного елемента, але менша кількості вихідних елементів.*

На основі того, що кількість підмножин, на які розкладається вихідна підмножина, менша кількості елементів, які в ній містяться, і ці підмножини не є порожні, можна стверджувати, що в тому випадку, коли всі підмножини, за винятком однієї або декількох, будуть містити мінімальну кількість елементів, тобто вони будуть одноелементними, загальна сума їх елементів буде меншою від кількості елементів вихідної множини. Але за правилом суми потрібно, щоб ця сума дорівнювала вихідній кількості елементів. Тоді для виконання цього правила потрібно, щоб хоч одна з підмножин мала більше одного елемента, що і доводить першу частину твердження 3.

Однак кількість елементів в цій одній підмножині не повинна згідно з тим самим правилом суми бути більшою від різниці між вихідною кількістю елементів і їх загальною сумою у решти підмножин. А оскільки ця загальна сума більша нуля, то ця різниця буде меншою кількості елементів у вихідній множині, і тому кількість елементів у згаданій підмножині також завжди буде меншою цієї кількості.

Якщо ж підмножин, які містять в собі більше одного елемента, буде декілька, то тоді ця різниця повинна бути розподілена між цими підмножинами, і кількість їх елементів тим більше не зможе бути більшою чи дорівнювати кількості елементів у вихідній множині. Тобто і друге твердження наслідку 3 виконується.

4 Серед усіх можливих розкладів вихідної множини на не порожні підмножини є розклад, в якому всі підмножини даного розкладу, за винятком однієї, будуть утримувати по одному елементу, а ця одна підмножина буде мати кількість елементів, яка буде найбільшою серед підмножин усіх можливих розкладів даної множини.

Це виходить з того, що для розкладу, який розглядається, за вимогою правила суми одна з його підмножин обов'язково повинна містити в собі різницю між кількістю елементів вихідної множини і сумарною кількістю елементів, які знаходяться в одноелементних підмножинах, більше одного елемента. Вона ж буде і найбільшою в порівнянні з усіма іншими можливими розкладами вихідної множини, тому що при збільшенні в даному розкладі хоча б в одній з одноелементних підмножин кількості елементів, що можливо при іншому розкладу вихідної множини, кількість елементів в підмножині, де знаходилась раніше різниця елементів згідно з правилом суми, повинна зменшитися на таку саму величину. Це означає, що найбільша кількість елементів серед підмножин усіх можливих розкладів вихідної множини, знаходиться в тому розкладі, в якому всі підмножини, за винятком однієї, є одноелементними, що і підтверджує твердження наслідку 4.

З наведеного вище правила суми і його наслідків випливають також такі твердження:

1 Якщо скінченна множина Q розкладається по кроках на підмножини, які не є одноелементними, то рано чи пізно в послідовності цих кроків настане такий крок, де всі підмножини будуть одноелементними.

Після першого розкладу вихідної множини на підмножини, якщо вона неодноелементна, в них повинно у відповідності до наслідку 3 правила суми міститися елементів *менше,* ніж їх було в цій множині до її розкладу. Отримані після розкладу підмножини з кількістю елементів більше одиниці на наступному кроці також можна знову розкласти на підмножини, в яких також зменшиться кількість елементів, і так буде відбуватися далі до того часу, поки всі підмножини, без винятку, не стануть одноелементними. А такими вони обов'язково стануть, тому що на кожному кроці розкладів відбувається зменшення елементів в підмножинах, які розкладаються, і рано чи пізно настане крок, де в підмножинах будуть знаходитися тільки одноелементні підмножини, що і доводить твердження 1.

2 Якщо на всіх кроках розкладів проводити всі без винятку розклади підмножин, в тому числі і розклади одноелементних підмножин, то отримане в цьому випадку на кожному кроці розкладів сумарне число елементів у всіх підмножинах буде дорівнювати числу елементів вихідної підмножини.

Це випливає з того, що оскільки на першому кроці були одержані підмножини, сумарна кількість елементів в яких дорівнює кількості елементів у вихідній множині, то і на другому ця кількість згідно з правилом суми не повинна змінитися. Такою вона залишиться і на третьому, четвертому і так далі кроках, поки не настане крок, де всі підмножини будуть одноелементними. У результаті доходимо висновку, який запропоновано в даному твердженні 2.

З цього твердження випливає такий наслідок:

Якщо на останньому кроці розкладів будуть одержані тільки одноелементні підмножини, то їх сумарна кількість буде дорівнювати кількості елементів у вихідній множині.

Більш складним буде питання про кількість одноелементних множин у випадку, коли деякі розклади вихідної множини закінчуються при появі одноелементних підмножин і в той самий час продовжуються розклади підмножин, які мають у своєму складі більше одного елемента, до того часу, поки не будуть отримані їх повні розклади. В цьому випадку після отримання в процесі розкладів одноелементної підмножини, тобто при одному із повних розкладів вихідної множини, її подальший розклад на підмножини закінчується.

Відповіддю на це питання буде таке твердження.

3 При отриманні при всіх розкладах вихідної множини по кроках одноелементних підмножин їх повна сумарна кількість буде дорівнювати кількості елементів у вихідній підмножині.

Будемо вважати, що попередньо вже проведені всі повні розклади вихідної множини, і тому в їх кінці отримані одноелементні підмножини, які, якщо їх об'єднувати, обов'язково ввійдуть в підмножини, з яких вони вийшли. Далі ці підмножини зі своїми елементами при новому об'єднанні знову ввійдуть в попередні щодо них підмножини. В ці самі підмножини ввійдуть і одноелементні підмножини, які були отримані раніш на цьому кроці розкладів. Кількість елементів в цих попередніх підмножинах згідно з правилом суми буде дорівнювати кількості елементів підмножин, які в них ввійшли. А оскільки всі ці підмножини складалися з одноелементних підмножин, то ця кількість буде дорівнювати кількості одноелементних підмножин, які в них входять. Далі за аналогією до цього будуть об'єднуватися ці підмножини і інші одноелементні підмножини в попередні до них підмножини до того часу, поки не буде отримана остання попередня підмножина, яка містить в собі елементи всіх підмножин першого кроку розкладів, тобто вихідна множина. Оскільки всі підмножини, які входять у вихідну множину, складалися тільки з одноелементних підмножин, то їх загальне число, отримане на всіх кроках розкладів, повинно дорівнювати кількості елементів вихідної множини, що і доводить наведене вище твердження 3.

Із твердження 3 з очевидністю випливає те, що повних розкладів, тобто розкладів, які закінчуються одноелементними множинами, буде стільки, скільки елементів має вихідна множина. Стільки ж буде і зображень чисел, які створюються з допомогою цих розкладів. При таких розкладах кожне з чисел, яке побудовано з його допомогою, буде сформованим. Це означає, що його можна виділити з послідовності інших чисел, які отримуються на цих розкладах.

Одноелементна підмножина, яка завжди є в кінці повного розкладу, може надалі продовжити розкладатися на саму себе, формуючи на подальших кроках уже інше за виглядом число. Однак це число з точки зору кількості, яку воно зображає, не несе в собі нової інформації. Тому такі розклади одноелементної підмножини на подальших кроках будуть далі називатися *надлишковими*. Одне з їх значень полягає в тому, що вони дозволяють отримати *перешкодостійкі* числа. Таких надлишкових розкладів для зображення однієї і тієї самої кількості у вигляді перешкодостійких чисел можна отримати багато. Однак для ефективного використання цих чисел треба брати таку кількість одноелементних підмножин в кожному повному розкладі вихідної множини, яка б дозволяла отримувати числа з однаковим числом розрядів.

Пояснимо все сказане вище з допомогою прикладу. Припустимо, що множина з 5 елементів {1,2,3,4,5} на першому кроці розкладається на 3 підмножини, такі, як {1,2.4}, {5} і {3]. Потім підмножина {1,2,.4} на другому кроці розкладається на дві підмножини {2, 4} і {1} і на третьому кроці підмножина {2,4} знову розкладеться на дві підмножини – {2} і {4}. У результаті маємо 5 повних розкладів вихідної множини, тобто розкладів, які закінчуються одноелементною множиною, але очевидно, що ці розклади отримані за різну кількість кроків – 1,2 і 3. Відповідно на цих 5 розкладах можна побудувати 5 чисел різної довжини, які з допомогою одноелементних множин в їх кінці легко відрізняються одне від одного. Як ці числа можна побудувати, буде показано в подальшому.

В розглянутому випадку повних розкладів вихідної множини на підмножини були відсутні розклади одноелементних підмножин на самих себе, але це легко можна зробити. Для цього достатньо після першого кроку продовжити розклад одноелементної підмножини {5}. Тобто на другому кроці можна було отримати, крім двох підмножин {2,4} і {1}, ще і після надлишкового розкладу підмножини {5} на саму себе ту ж саму підмножину {5}. Такий самий надлишковий розклад підмножини {5} можна буде зробити і на третьому кроці. Аналогічно на другому і третьому кроці можна буде розкласти на одну підмножину одноелементну підмножину {3}, а для множини {1} це можна виконати на третьому кроці. Ці розклади теж будуть надлишковими.

Характерною ознакою цих розкладів в даному прикладі буде те, що після них кількість кроків для п'яти повних розкладах вихідної множини є однаковим і дорівнює 3. Відповідно і побудовані на цих розкладах числа будуть мати однакову кількість розрядів, тобто по 3, хоча можна було б зробити і більше, наприклад, по 4 чи 6. Для цього треба було б лише ввести відповідно ще для всіх отриманих вище повних розкладів 1 чи 3 кроки одноелементних розкладів.

Якщо відійти від наданого вище прикладу, то при розкладах, які розглядаються в даній роботі, спочатку на першому кроці розкладається на підмножини безпосередньо сама вихідна множина, а за нею в подальшому по кроках отримані в результаті цього розкладу підмножини. При цьому в загальному випадку на кожному кроці розкладається більше ніж одна підмножина, і тільки на першому кроці розкладається лише одна вихідна множина Q. Проте на інших кроках такий розклад підмножин на одну підмножину в практичних задачах відбувається значно рідше і може сприйматися як виняток.

Однак у деяких випадках поява розкладів підмножин на одну підмножину до появи одноелементної підмножини дозволяє будувати досить цікаві надлишкові системи числення. Навіть коли вихідна множина утримує всього один елемент, можна побудувати унітарну систему числення, яка має практичне значення і використовується при побудові ряду цифрових пристроїв і систем (4). Тому загальна теорія систем числення повинна враховувати і випадки появи одноелементних підмножин на попередніх кроках розкладів.

КОДУВАННЯ ПІДМНОЖИН ПРИ ПОЗИЦІЙНІЙ ЛІЧБІ

Підмножини з одним елементом можуть бути отримані уже на першому кроці розкладів, коли розкладається безпосередньо множина Q. У цьому випадку, крім одноелементних підмножин, можуть бути отримані і підмножини з більшим числом елементів. Це відповідно до правила суми можливо, якщо кількість підмножин, на які розкладається вихідна множина Q, буде менша, ніж кількість елементів, які в ній містяться.

Візьмемо якусь одну з одноелементних множин і вилучимо її із списку множин першого кроку розкладів. Тоді згідно із правилом суми загальна сума елементів в підмножинах першого кроку, які залишилися, буде на одиницю меншою від числа елементів вихідної множини. Відповідно і кількість елементів цієї множини в ній зменшиться на 1.

Значення цієї загальної суми елементів підмножин полягає в тому, що вона може бути використана, по-перше, для кодування вилученої одноелементної підмножини і тим самим елемента вихідної множини і, по-друге, для встановлення порядкового номера (числа) цього елемента щодо інших елементів цієї множини.

Цей номер після вилучення одноелементної підмножини з розкладання вихідної множини набуде значення, яке буде дорівнювати зменшеному на 1 числу елементів вихідної множини. Очевидно, що воно буде найбільше серед можливих значень номерів, які можуть впорядковувати елементи вихідної множини. Після кодування вилученої підмножини цей найбільший порядковий номер закріплюється за нею і відповідним елементом вихідної множини. Він також, якщо до нього додати 1, стане кількісним номером чи кількісним числом, яке буде відображати кількість всіх елементів у вихідній множині. Це означає, що отримана сума елементів у підмножинах першого розкладу не тільки є найбільшим номером елемента, який кодується, а і дозволяє визначити *кількість* елементів у вихідній множині.

Однак на першому кроці розкладів може бути отримана не одна одноелементна підмножина, а дві і більше і навіть підмножини, які всі є одноелементними. Розглянемо цей хоч і крайній, але корисний для аналізу позиційного кодування випадок.

Якщо всі підмножини, на які була розкладена вихідна множина на першому кроці розкладів, будуть одноелементними, то тоді вибирається з них якась одна підмножина і у відповідності до розглянутих вище правил кодується числом, яке на одиницю буде меншим від суми елементів у цій множині. Після цього вона вилучається з розгляду.

З підмножин, що залишилися після першого кодування однієї з одноелементних підмножин, вибирається нова одноелементна підмножина і кодується числом, яке тепер уже буде на одиницю меншим від попереднього. Таким чином буде встановлений порядковий номер ще одного елемента вихідної множини і одночасно, якщо додати до цього число 1, визначена кількість елементів, що залишились у вихідній множині.

Подібне кодування із встановленням порядкових і кількісних номерів і вилучення одноелементних множин буде продовжуватися до того часу, поки воно не дійде до кодування останньої одноелементної множини. Для даного випадку уже не залишиться одноелементних множин, сумарна кількість елементів в яких визначала б номер останньої підмножини, окрім порожньої множини, яка наявна скрізь, в тому числі і в ряду одноелементних підмножин.

Відомо, що в ній відсутні елементи, тобто їх число дорівнює 0. Тому остання одноелементна множина, яка ще не має свого кодового номера, може кодуватися в даному випадку тільки *нулем*. У цьому кодуванні проявляється особлива роль 0, без якого не можливі позиційні системи числення. Розглянемо більш докладно цю роль.

Наявність 0 зазначає, що попереду нього є 0 множин і тому нема потреби в їх кодуванні. Нуль є першим порядковим числом чи номером, з якого починається перелік підмножин. Тобто 0 створює базу переліку елементів вихідної множини при позиційному кодуванні. Це означає, що порядкові натуральні числа при такому кодуванні повинні йти з 0, а не з 1. При цьому важливо також і те, що нуль не тільки вказує порядок, а одночасно і кодує в даному випадку одноелементну множину.

Після кодування початкової чи базової одноелементної підмножини нулем наступна за нею по порядку підмножина буде кодуватися числом, яке буде дорівнювати кількості елементів в підмножині, кодованій 0. Оскільки число елементів у цій підмножині дорівнює 1, то і порядковий номер цієї підмножини дорівнює 1. Число 2 визначається уже сумарною кількістю елементів в двох попередніх щодо до нього одноелементних підмножинах, кодованих 0 і 1, тобто 2. Те саме стосується будь-якого великого числа. Його номер визначається кількістю одноелементних множин, які йдуть в їх упорядкованому ряді попереду нього. Наприклад, якщо попереду числа, порядок якого визначається, йде 120 одноелементних підмножин, то це означає, що порядковий номери цього числа буде 120, а порядкові номера всіх підмножин, які передували йому, будуть змінюватися від 0 до 119.

Для більш досконалого розгляду питання позиційного кодування чисел візьмемо такий приклад. Припустимо, що у вихідній множині маємо 10 елементів. Тоді після її розкладу на 10 одноелементних підмножин і вилученні з подальшого розгляду однієї з них елемент, який знаходиться у вилученій підмножині, буде кодуватися числом, яке відповідає сумі елементів решти одноелементних підмножин - 9, наступний після вилучення чергової одноелементної підмножини - числом 8 і так буде відбуватися далі до того часу, поки залишиться одна одноелементна підмножина. Її елемент може кодуватися тільки кількістю елементів порожньої множини, тобто 0. У результаті 10 елементів вихідної множини кодуються 10 числами 0,1,2, ..., 9.

Для подальшого викладу важливо підкреслити, що всі ці числа були одержані у вигляді сум елементів в одноелементних підмножинах, які залишались при поступовому вилученні кожної з них з розгляду у випадковому порядку. Тобто ці числа, крім того, що вони кодують вилучені підмножини, ще й задають їх порядкові номери і тим самим вказують на порядок елементів вихідної множини. Цей порядок проявляється в тому, що число 0 свідчить про те, що попереду нього йде нуль одноелементних підмножин; число 1, що 1 така підмножина; число 2, що 2 і 9, що 9.

Поряд з цим порядкове число 9 свідчить про те, що кількість елементів у вихідній множині дорівнює 10, а число 8, – що після вилучення ще одного елемента з вихідної множини їх кількість стала дорівнювати 9. Це саме правило стосується також і порядкових чисел 8, 7 і в кінці числа 0, яке кодує, з одного боку, одноелементну множину з числом елементів 1, а з іншого – визначає, що кількість елементів в пустій множині, яка стоїть попереду нього, дорівнює 0. Крім того, 0 свідчить про те, що кількість елементів у вихідній множині після вилучення з неї 9 елементів стала дорівнювати 1 елементу.

Таким чином, у наведеному вище прикладі маємо 10 порядкових невід'ємних чисел 0,1,2, ...,9, які відображають порядок розміщення 10 елементів у вихідній множині. Поряд з цим маємо 10 кількісних невід'ємних чисел: 0,1,2, ..., 9, які в даному випадку збігаються з порядковими. Вони дозволяють зобразити суми кількостей елементів у підмножинах з меншими від них номерами.

Кодування, яке розглядається, належить до розкладу вихідної множини на одноелементні підмножини. Однак при кодуванні підмножин, серед яких є не тільки одноелементні, з'являється відмінність, яка полягає в тому, що 0 і інші числа, які йдуть за ним, у даному випадку необов'язково кодують підмножини тільки з одним елементом, а в загальному випадку можуть кодувати і підмножини з більшою кількістю елементів. Ця процедура кодування може відбуватися як завгодно довго до того часу, поки не закінчаться всі підмножини в розкладі вихідної множини. Тоді останнє по порядку число буде кодувати всі елементи вихідної підмножини за винятком підмножини, яка кодується останньою.

Наприклад, при розкладанні вихідної множини на три підмножини цифра 0 кодує підмножину з 500 елементів, 1 - з 1000, 2 – з 1200, 3 - 100. Відповідно до цього кодування порядковий номер підмножини 0 буде дорівнювати 0, підмножини 1 - 500, підмножини 2 – 1500, підмножини 3 – 2700. Припустимо, що елемент, який кодується, знаходиться в 3-й підмножині. Тоді його номер, очевидно, буде більше2700 і менше 2800. Для того щоб його знайти остаточно, потрібно буде виконати ще другий крок розкладів, тепер вже для підмножини 3, а можливо і подальші кроки до того часу, поки не будуть у їх кінці отримані одноелементні підмножини. Якби підмножина 3 була одноелементною, то тоді номер цього елемента дорівнював би 2700, і її другий крок розкладів був би не потрібен. Кількість елементів при цьому у вихідній множині дорівнювала б 2701, тобто на 1, більше ніж максимальний порядковий номер. Якщо ж елемент, номер якого шукається, знаходиться в підмножині 1, то його порядковий номер на першому кроці розкладів буде дорівнювати 0 і далі він буде уточнюватися на наступному другому кроці розкладів.

ПОЗИЦІЙНЕ КОДУВАННЯ ЧИСЕЛ

Порядкові і кількісні числа збігаються між собою тільки у випадку розкладання вихідної множини на одноелементні підмножини. Такий випадок, як було доведено вище з допомогою правила суми, буде завжди, коли розклади вихідної множини будуть продовжуватися як завгодно довго. Це обов'язково станеться, тому що через деяку кількість кроків розкладів всі отримані після них підмножини будуть одноелементними, тобто всі розклади стануть повними, хоча при цьому кількість кроків для різних розкладів не обов'язково буде однаковою. Однаковими вони можуть стати тільки тоді, коли кожний розклад на одному і тому самому його кроці буде відбуватися на одне і те саме число підмножин. Такими самими будуть і послідовності номерів, які кодують ці підмножини. Це означає, що позиційна система числення є вибудувана, і кожна одноелементна множина в її кінці являє окреме позиційне число, яке не може бути початком будь-якого іншого. Таке кодування ще називається *префіксним*.

Розглянута вище ідея кодування в розклад одного кроку одних одноелементних підмножин за допомогою кількостей елементів інших потребує у випадку великої кількості підмножин, які кодуються, такої самої великої кількості чисел, що, в свою чергу, призводить до необхідності мати велику пам'ять. Це є основним недоліком цієї ідеї для випадку, коли розклад вихідної множини на одноелементні підмножини проходить лише на першому кроці.

Але зазначений недолік легко виправити, якщо збільшити число кроків розкладу вихідної множини і тим самим обмежити кількість підмножин, які кодуються, і кодуючих їх чисел невеликим числом. Такі числа для такого випадку називаються *цифрами*. Ці цифри досить легко можна запам'ятати і потім уже використати для побудови кодових зображень чисел. Вони утворюють особливу множину – *алфавіт* цифр системи числення. Для нього є обов'язковим, як це було показано при кодуванні одноелементних підмножин, наявність нуля і всіх інших невід'ємних цифр із заданого відрізка без будь-яких вилучень, тобто наявність цифр 0,1,2, ...

У цьому загальному випадку з більшою кількістю кроків розкладу, ніж один, на першому і інших його кроках кожна цифра алфавіту може кодувати підмножину, яка містить у собі більше одного елемента. Тоді для визначення загальної кількості, яку кодує ця цифра, підсумовуються кількості елементів підмножин, які кодуються всіма меншими від неї цифрами алфавіту, починаючи з 0, і якраз ця сума і буде цією кількістю.

Оскільки розклад числа на підмножини може проходити не за один крок, як це розглядалося вище, а за декілька, то в результаті буде одержане позиційне число, яке відповідно буде мати стільки ж розрядів, скільки було зроблено кроків розкладів для його одержання. Для кожного з цих розрядів характерною ознакою буде його *номер,* який відповідає номеру кроку розкладів підмножин. Це означає, що названим способом можна зобразити будь-яке велике число, але за це потрібно буде заплатити новими розрядами числа, загальна кількість яких в числі, очевидно, буде збільшуватися із збільшенням кількості чисел, які зображаються з допомогою цих розрядів. Однак це зростання розрядів буде не настільки стрімким, як зростання кількості цифр в алфавіті системи числення, де кодування проходить за один крок, тобто при непозиційному кодуванні.

У відповідності до сказаного вище для того щоб зменшити кількість цифр в алфавіті на першому кроці і отримати позиційне число з більшим числом розрядів, необхідно розкласти вихідну множину на першому кроці на підмножини, серед яких повинна бути хоча б одна, яка б не була одноелементною. Тільки тоді виникає необхідність розкладати її на другому кроці на нові підмножини, і,отже, з'являються другий і подальші кроки розкладів.

Кожну одноелементну підмножину, яка залишилася при цьому після першого розкладу, можна або розкладати на другому кроці ще на одну таку саму одноелементну підмножину, або зовсім їх не розкладати. В останньому випадку одноелементні підмножини після відповідного кодування їх цифрами створять числа довжиною в один розряд, і тому в кінцевому підсумку сформовані числа після всіх розкладів і відповідних кодувань будуть мати різну довжину, тобто будуть належати до *нерівномірних* кодів.

При розкладі одноелементних множин на другому кроці на самих себе вони зможуть кодуватися тільки 0, і відповідні їм числа будуть мати уже не один, а два розряди. У такому випадку, якщо кодування всіх чисел на другому кроці розкладів буде закінчено, що можливо тільки тоді, коли всі підмножини отримані після другого розкладу, будуть одноелементними, отримані числа стануть рівномірними, тобто будуть належати до *рівномірних* кодів.

Щоб краще пояснити наведені вище міркування, розглянемо приклад. Припустимо, що за два кроки вихідна множина з 14 елементів розкладається так, що на першому кроці буде отримано 9 повних розкладів, тобто таких, які закінчуються одноелементною множиною, а на другому решту - 5. Це означає, що згідно з правилом суми на першому кроці всі елементи вихідної підмножини повинні бути розкладені на 10 підмножин, серед яких 9 буде одноелементних, а одна буде містити в собі решту з 5 елементів.

Цю одну підмножину з 5 елементів будемо кодувати 0, а залишок в довільному порядку цифрами 1, 2, ..., 9. Тоді на першому кроці розкладів одержимо послідовність цифр 0,1, ... 9. На другому кроці розкладів кодуються тільки одноелементні підмножини, які отримуються після розкладу підмножини з 5 елементів, кодованої 0. Це означає, що за два кроки будуть отримані повні розклади і відповідні їм числа - 00,01,02,03,04. Інші 9 чисел, які були отримані на першому кроці, залишаються без змін і тому більше не кодуються. У результаті будуть отримані всі зображення 14 елементів вихідної множини у вигляді чисел – 00,01,02,03,04,1,2,3, ..., 9. Як видно із зображень, отриманих у процесі розкладів чисел, вони є нерівномірними і, крім того, префіксними числами. Останнє означає, що ні одне з цих чисел не може бути початком іншого.

У випадку рівномірного кодування елементів вихідної множини потрібно 9 одноелементних підмножин, отриманих на першому кроці кодування, ще раз розкласти на самих себе на другому кроці розкладів і відповідно кодувати їх 0. Тоді кодові зображення цих елементів будуть мати вигляд 00,01, ..., 10, 20, ..., 90. Тобто мінімальне число 00 в системі числення, яка розглядається, відображає число 0, а максимальне 90 – число 14. Це буде надлишкова система числення з рівномірним кодуванням чисел, в якій рівномірні числа, на відміну від попередньої з нерівномірними числами, мають властивість перешкодостійкості. Наприклад, поява числа 21 буде сприйнята як помилка, тому що такого числа, як 21, серед їх множини діапазону немає.

До цього часу розглядалися розклади підмножин на два кроки. Якщо після другого кроку розкладів нові підмножини містять більше одного елемента, то повинна відбутися процедура їх третього, четвертого і потім подальших розкладів, які будуть проходити до того часу, поки всі отримані у результаті розкладів підмножини повністю не стануть одноелементними. Як тільки це відбудеться, відповідні цим підмножинам числа слід вважати сформованими.

При цьому ці числа можуть належати як до рівномірних кодів, так і до нерівномірних. Все залежить від попередніх розкладів одноелементних множин. Якщо вони продовжувались далі розкладатися на самих себе, то ці числа зможуть стати рівномірними. У протилежному разі, коли їх розклади зупиняються, будуть сформовані нерівномірні коди. Кількість отриманих під час розкладів чисел, як рівномірних, так і нерівномірних, в будь-якому випадку буде дорівнювати, як це випливає з правила суми, кількості елементів, яка була у вихідній множині.

Для сформованих названим способом чисел з точки зору їх кількісного значення не дуже важливо, до яких кодів вони належать – рівномірних чи нерівномірних. Для них важливим є лише те, щоб їх кількісні значення були одними й тими самими. Все інше належить лише до форм зображення цих чисел. Однак ці форми в багатьох випадках відіграють неабияку роль при розв’язанні різних практичних задач. Тому на практиці задачі вибору форми зображення чисел, тобто способу їх кодування в позиційних системах числення, приділяється значна увага. А для цього необхідно дослідити всі можливі способи такого кодування чисел, навіть у рамках одних і тих самих розкладів вихідної множини.

НУМЕРАЦІЯ ЧИСЕЛ В ПОЗИЦІЙНИХ СИСТЕМАХ ЧИСЛЕННЯ

Отримані вище розклади одних підмножин на інші до одноелементних і відношення належності між ними дозволяють побудувати довільні позиційні системи числення. Елементами таких систем будуть утворені в процесі розкладів підмножини, а зв'язки між ними створюють структури цих систем. Якщо при цьому ще задати число елементів в кожній підмножині і відповідним чином ввести для кожного розкладу їх кодування з допомогою алфавітів цифр, то можна вважати, що система числення побудована.

Тепер треба лише побудувати на її основі числову функцію, яка б дозволяла знайти за зображеннями чисел самі ці числа (номера), тобто розробити відповідний алгоритм. Задача побудови такої числової функції і обмежень на її параметри в багатьох практичних випадках є досить складною, але в загальному випадку вона має досить простий розв'язок, який і буде розглянутий нижче. Він має такий вигляд.

Для того, щоб знавідбуватися за зображенням числа в позиційній системі числення порядковий номер елемента вихідної множини, тобто порядковий номер зображеного числа, треба на кожному кроці в одному з його розкладів знайти підмножину, цифра якої дорівнює цифрі відповідного даному кроку розряду числа, і потім в цьому розкладі знайти сумарну кількість елементів у всіх підмножинах, цифри яких будуть меншими від цифри числа, а після цього знайти загальну суму всіх таким чином отриманих на кожному кроці розкладів сум. Ця загальна сума і буде порядковим номером зображеного числа.

Доведемо спроможність цього розв'язку. Дійсно, якщо число правильне, то на кожному кроці розкладів в одному з них серед всіх підмножин, пронумерованих цифрами з алфавіту того чи іншого розкладу, знайдеться одна підмножина з елементом, порядковий номер якого визначається. При цьому цифра, якою кодується ця підмножина, збіжиться з цифрою числа у відповідному до цієї підмножини розряді. Якщо ця підмножина одноелементна, то якими б не були всі інші підмножини, які кодуються меншими цифрами - одноелементними чи ні, її номер і відповідний номер її елемента буде дорівнювати сумарній кількості елементів у цих підмножинах. Тобто порядковий номер зображеного числа буде визначений.

Якщо ця підмножина не одноелементна, то сумарна кількість у зазначених вище підмножинах визначає тільки частину шуканого номера. Інша частина, якої не дістає, щоб його визначити, може бути знайдена при розкладі цієї неодноелементної підмножини на наступному кроку розкладів. Для цього на новому кроці відбувається розклад цієї підмножини на нові підмножини і нумерація їх цифрами **0,1, ...**

Далі знову проводиться порівняння цифри числа, але тепер уже наступного меншого розряду із зазначеними цифрами алфавіту, які кодують нові підмножини, і в його результаті знаходиться підмножина з необхідним елементом вихідної множини. Якщо вона одноелементна, то знаходиться сума елементів усіх підмножин, цифри яких за своїм значенням менші цифри відповідного розряду числа, і на цьому номер числа і відповідного до нього елемента вихідної множини вважається знайденим.

У протилежному разі цикл визначення порядкового номера числа повторюється, але вже для цифри меншого розряду числа. Він буде відбуватися до того часу, поки не буде при розкладах на наступних кроках отримана одноелементна множина, а що це рано чи пізно станеться, випливає, як було доведено вище, із правила суми.

Наслідок 1 *Серед чисел позиційної системи числення обов'язково буде нуль.*

Випливає з того, що серед усіх можливих розкладів обов'язково будуть розклади, де в них попереду на всіх їх кроках знаходяться порожні підмножини. Кількість елементів у кожній з них, очевидно, дорівнює нулю, і відповідна їм сума також буде дорівнювати нулю.

Наслідок 2 *Найбільше число позиційної системи числення на 1 менше кількості всіх можливих чисел, які вона представляє своєму діапазоні.*

Це випливає з того, що першим числом, яке входить в діапазон чисел позиційної системи числення, обов'язково входить 0. Як наслідок з цього випливає, що діапазон чисел, який складається з 0, дорівнює 1. Тоді наступне число у відповідності до наведенних вище алгоритмів знаходження номера позиційного числа буде дорівнювати 1, а кількість чисел в діапазоні, який в даному випадку подається, дорівнює 2 – 0 і 1. Аналогічно буде складатися діапазон і для кількості чисел 3, 4 і так далі. Тобто діапазон чисел з трьома числами подається цифрами 0,1,2, а з чотирма – 0,1,2,3.

Повертаючись до наведеного вище правила одержання номера числа з допомогою числової функції за його виглядом, доходимо висновку, що в позиційних системах числення використовуються три ідеї:

1 Введення на кожному кроці розкладів алфавітів цифр, які кодують їх підмножини в порядку 0,1, 2, ... При цьому 0 є обов'язковим елементом, з якого починається кодування підмножин, а інші цифри повинні відбуватися тільки у вказаному наростаючому порядку.

2 Знаходження значення цифри числа, яка відповідає тому чи іншому кроку розкладів на підмножини, як суми елементів у всіх підмножинах, які кодуються меншими цифрами, і потім сумування цих сум між собою з тим, щоб знайти кінцевий підсумок, який і відображає кількісне значення цієї цифри на даному кроці розкладів.

3 Знаходження значення числа як суми результатів значень цифр, отриманих під час підсумовувань на всіх кроках розкладу. В кінцевому підсумку ця сума дає значення числа, зображення якого подається у вигляді послідовності його цифр.

Ці ідеї прості і ефективні. На сьогодні неможливо протиставити їм більш досконалий аналог. Тому позиційне кодування чисел витіснило всі інші непозиційні методи кодування як менш досконалі. Людство наближалося до цих ідей досить довго і застосувало їх не так вже і давно – менше 1000 років тому у вигляді позиційних систем числення, таких, наприклад, як десяткова система числення.

ВИСНОВКИ

Якщо тепер підбити підсумок наведеного вище матеріалу, то можна стверджувати, що систем числення може бути побудовано безліч. Тільки найпростіших рівномірних систем числення, в яких кожний розклад має одну і ту саму кількість підмножин з однаковою кількістю елементів, можна побудувати стільки, скільки є натуральних чисел. Щоправда, використовується їх на практиці не дуже багато. Серед них десяткова, двійкова, вісімкова. Але ж ще можна побудувати і безліч більш складних нерівномірних систем числення, в яких розклад відбувається на підмножини з різною кількістю елементів, і серед них можна знайти системи числення, досить цікаві як з теоретичної, так і з практичної точки зору. Тому започаткований у даній роботі розгляд таких і їм подібних позиційних систем числення, на наш погляд, має перспективу на його продовження.

Крім того, він уже сьогодні дозволив отримати досить ефективні комбінаторні системи числення на основі існуючих комбінаторних об'єктів, таких, як перестановки, сполучення та числа фібоначі. Ці системи мають практичне значення в задачах кодування інформації, а також для побудови відповідних спеціалізованих цифрових пристроїв. Хоча вони, звичайно, не зможуть замінити традиційні системи числення як базові при побудові універсальних цифрових систем, але успішно можуть розв’язувати спеціальні задачі в тих самих системах, збільшуючи їх швидкодію і надійність.

Важливим також є і те, що ці і подібні їм позиційні системи числення, які ще можуть бути створені на основі більш складних комбінаторних об’єктів, ніж названі вище, мають властивість генерувати їх і їхні модифікації, тим самим розв’язуючи важливу задачу комбінаторики на побудову комбінаторних об’єктів, яка на сьогодні ще не є повністю розв’язаною. Це, у свою чергу, дає можливість будувати відповідні електронні цифрові пристрої і системи в ряді випадків з більшою ефективністю їх роботи, ніж системи, які побудовані на інших принципах.

Наведені в даній роботі результати дають теоретичний матеріал для побудови таких складних систем числення і відповідних їм електронних пристроїв і систем. Для цього лише потрібно за наявності загальної системи розглянутого в роботі кодування підмножин в кожному розкладі ввести відповідні функціональні зв’язки між цими розкладами, і тоді числа, які буде генерувати система числення, будуть відповідати структурі комбінаторних об’єктів, що дозволить врешті їх побудувати і за необхідності перелічити. А щоб ввести ці функціональні зв’язки, треба виконати наведені вище загальні умови побудови позиційних систем числення, які є одні і ті самі, як для простих систем, так і для дуже складних.

SUMMARY

The concepts and ideas on number positional coding are considered in the paper. Construction of number systems and principles of positional count are under review as well. The division of an initial set of elements into subsets, then these subsets into another subsets to one element sets is a main idea. At these separations the operation of sum is applied. The next idea, which is used to construct positional number systems, is coding each of obtained due to the division subsets at an every step by a cipher of a alphabet, which is only for current division. The alphabets, as well coding of the subsets on their basis, are built in according to the developed in the paper rules. The significant idea in positional systems is a way of summing elements of the initial subsets at an every step of division with the help of the alphabets, and then unification of all these results into one sum. This sum determines quantitative value of a number, which is coded. Aggregate of subsets divisions, taken on their steps, and their alphabets ratios of affiliation between subsets with corresponding relations, all values of quantities of elements in subsets at all their separations and steps of separations as well, creates a positional number system finally, with the help of which can be obtain ed any great number.

СПИСОК ЛІТЕРАТУРИ

1. Феферман С. Числові системи Основи алгебри і аналізу / Пер. з англ. – М.: Наука, 1971. – 440 с.
2. Кнут Д. Мистецтво програмування. Том 2. Півчислові алгоритми, 3-тє видання /Пер. з англ. – М.: Вид-во „Вільямс”, 2000. – 832 с.
3. Рейнгольд Е., Нивергельт Ю., Део Н. Комбінаторні алгоритми. Теорія і практика / Пер. з англ. – М.: Вид-во „Мир”, 1980. – 465 с.
4. Поспєлов Д. А. Арифметичні основи обчислювальних машин дискретної дії: Навчальний посібник. – М.: Вища школа, 1970. – 308 с.

О.А. Борисенко, д-р техн. наук, проф.
Сумський державний університет

Надійшла до редакції 29 березня 2007 р.