

УДК 519.765(045)

МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ ВІДОБРАЖЕННЯ МОДЕЛЕЙ ПОНЯТЬ

В. О. Малярчук, канд. техн. наук; **А. Ю. Кононюк**, канд. техн. наук, доц.;
І. В. Малінкін, канд. техн. наук, доц.

Національний авіаційний університет

e-mail: mva-nau@mail.ru

Наведено структурну схему універсального математичного забезпечення створення математичних моделей понять, представлених у растрових пристроях. Описано математичне забезпечення, орієнтоване на випуск документації про сформовану математичну модель поняття, що являє собою набір текстових і графічних елементів. Запропоновано HTML-інтерфейс, який складається з набору підпрограм двох груп: загальносистемних підпрограм та підпрограм, що формують елементи графічного зображення моделі поняття.

Ключові слова: поняття, математична модель, HTML-інтерфейс, растровий інтерфейс, алгоритм, синтаксис, семантика, множина, масив.

The structural diagram of the universal software, of the creation mathematical models of the concepts produced in raster devices, was provided. The software oriented on documentation release about the created mathematical model of concept that is a set of textual and graphic elements is described. The offered HTML- interface consists of a set of subprograms of two groups: general-system subprograms and the subprograms, which create elements of the graphics of model of the concept.

Keywords: concept, mathematical model, HTML-interface, bitmap interface, algorithm, syntax, semantics, set, array.

Вступ

Робота системи формування науково-розроблювальних понять (надалі — понять) пов'язана з обробкою і виведенням великих обсягів формалізованої інформації як текстової, так і графічної. Поняття, яке використовується в комп'ютерних системах, повинно бути подане у формі математичної моделі, як правило, або аналітичної, або графічної [1; 2]. Кінцеві результати формування поняття виводяться на різні типи візуальних носіїв інформації.

Інформація про поняття, що виводиться, у відносно компактному растровому або векторному видах від ЕОМ або від пристрою зчитування з проміжного носія надходить у пристрій управління.

У разі представлення моделі поняття в графічному вигляді, остання містить координати початку і кінця графічних об'єктів, необхідних для завдання відрізків прямих, дуг і кіл; службові ознаки, зокрема ознаки виду ліній, коду символів тощо. Координатна інформація про поняття надходить в інтерполятор, який за певним алгоритмом виробляє унітарні коди для блоків управління приводами по координатах X і Y [3].

На рис. 1 показано структурну схему універсального математичного забезпечення створення математичних моделей понять, представлених у растрових пристроях [4].

Система має два орієнтовані входи, розраховані на користувача-непрограміста і на користувача-програміста.

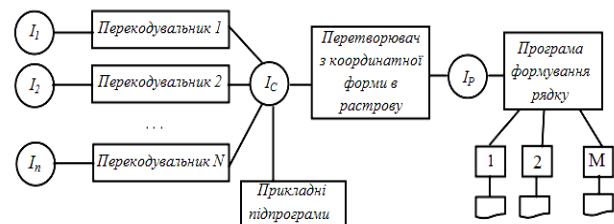


Рис. 1. Структура комплексу формування і виведення растрової інформації

Основними мовами (програмними інтерфейсами) системи є: множина координатних мов (I_1, I_2, \dots, I_N), перша проміжна мова — HTML-інтерфейс (I_c), друга проміжна мова — узагальнений растровий інтерфейс (I_r).

Множина координатних мов є входом, орієнтованим на користувача-непрограміста HTML-інтерфейсу, з одного боку, є проміжною мовою, з іншого — слугує входом, орієнтованим на користувача-програміста. З кожної вхідної мови може бути побудований перекодувальник (транслятор) на мову HTML.

Зазвичай такі транслятори нескладні, оскільки обидва типи мов близькі за семантикою. HTML-інтерфейс є стандартним програмним інтерфейсом, тому включення нової вхідної мови не становить ніяких труднощів. Для цього потрібно розробити лише один транслятор і включити його в систему.

За необхідності увійти в систему безпосередньо через HTML-інтерфейс користувач-програміст пише прикладну програму, елементами якої є HTML-оператори.

Переведення інформації з HTML-інтерфейсу в узагальнену растрову мову здійснює програма перетворення з координатної форми в растрову. Узагальнений растровий інтерфейс — це мова, орієнтована на формування виразів (рядків) виводу інформації про математичну модель поняття в реальному масштабі часу. Інформація в растровій мові зберігається у файлах.

Особливості програми формування растрової інформації про математичну модель поняття полягають у тому, що структура узагальненого растрового інтерфейсу повинна забезпечити формування виразу (у разі растрового координатографа) у вигляді бітів за час одного оберту барабана [3].

Подібна структура є універсальною, оскільки різні зовнішні растрові пристрої можна підключати до такої системи через драйвери або програми виводу. Зміни в програмі формування виразу, залежні від типу растрового пристрою, вводяться через мікропроцесори. Транслятори і перетворювачі з координатної форми в растрову не залежать від типу растрового пристрою.

HTML-інтерфейс

Математичне забезпечення, що описується, орієнтоване на випуск документації про сформовану математичну модель поняття, що являє собою набір текстових і графічних елементів (графічні зображення понять, креслення, текстові документи та ін.).

HTML-інтерфейс складається з набору підпрограм двох груп: загальносистемних підпрограм, які на початку роботи налаштовують систему на певні пристрої і значення параметрів (METADATA), а в кінці роботи формують інформацію про поняття в узагальненій растровій мові (GLYPH); підпрограм, що формують елементи графічного зображення моделі поняття: для кожного символу за допомогою підпрограми (UNICODE) цифрами прописувалися контури дуг, кіл, ліній і іншого. Усі координати і розміри в цих програмах задаються в міліметрах дійсними числами. Розглянемо ці підпрограми докладніше.

Підпрограма ініціації — METADATA. Звернення до підпрограми: GLYPH. Параметри підпрограми: UNICODE — ім'я символу в коді ASCII (не більше 20 байтів). Ім'я пристрою для растрового координатографа — RK, для растрового DZM — DZ.

Крок виводу задається тільки для пристроїв із змінним кроком (для графічного пристрою встановлюється значення кроків — 0,125; 0,0625; 0,0250 мм) [3].

Підпрограма остаточного формування інформації про поняття в узагальненій растровій

мові — HORIZ-ADV. Звернення до підпрограми: HORIZ-ADV. При написанні призначених для користувача програм підпрограма GLYPH повинна викликатися першою, а підпрограма HORIZ-ADV — останньою.

Підпрограма формування відрізка. Зупинимося передусім на системі координат і понятті «товщина лінії».

Координати точок зображення завжди задаються в декартовій системі координат, початок якої збігається з лівим верхнім кутом поля виводу, а осі спрямовані уздовж меж поля виводу таким чином, що координати усіх точок поля виводу не негативні (рис. 2).

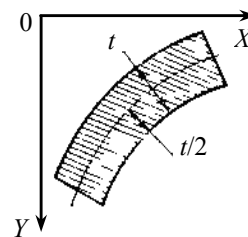


Рис. 2. Система координат і товщина лінії

При обробці інформації про геометричні елементи враховується товщина лінії, що задається як параметр до відповідної підпрограми. Будь-яка крива задається своєю середньою лінією, тоді параметр t (рис. 2) є товщиною лінії. Звернення до підпрограми: GLYPH.

Параметри: XY — масив координат відрізка; $XY(1)$, $XY(2)$ — координати x і y першої точки відрізка; $XY(3)$, $XY(4)$ — координати другої точки відрізка; THIN — товщина відрізка; I — тип лінії (1 — суцільна, 2 — штрихова, 3 — штрихпунктирна, 4 — розмірна із стрілками з обох кінців, 5 — розмірна із стрілкою у початковій точці, 6 — розмірна із стрілкою у кінцевій точці).

Приклад відрізка наведений на рис. 3.

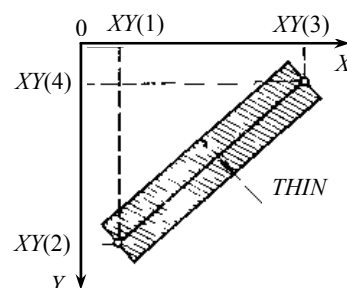


Рис. 3. Завдання лінії

Підпрограма формування кола. Звернення до підпрограми: GLYPH.

Параметри: XY — масив з інформацією про коло; $XY(1)$, $XY(2)$ — координати центру кола; $XY(3)$ — радіус (при параметрі $K = 0$) або діаметр (при $K = 1$) середньої лінії кола; THIN і I — те ж, що і в попередньому випадку (рис. 4).

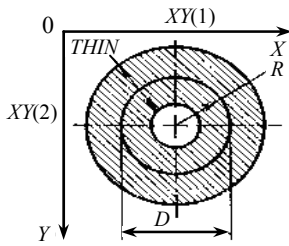


Рис. 4. Завдання кола

Підпрограма формування дуги. Звернення до підпрограми: GLYPH. Параметри: XY — масив з інформацією про дугу; $XY(1), XY(2)$ — координати центру дуги; $XY(3), XY(4)$ — координати початкової точки дуги; $XY(5), XY(6)$ — координати кінцевої точки дуги; $IDIR$ — напрямок дуги (рис. 5). Інші параметри такі ж, як і в попередніх випадках.

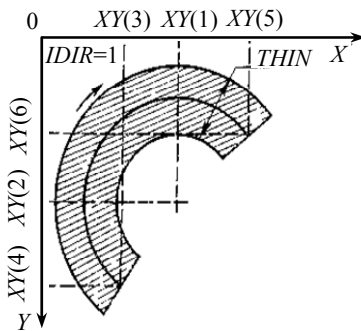


Рис. 5. Завдання дуги

Підпрограма формування стандартних графічних моделей понять. Звернення до підпрограми: GLYPH.

Параметри: XY — координати точки поля виводу, куди поміщається початкова точка прямокутника, в який вписується стандартний знак; NULL — ознака положення початкової точки (при NULL-1 — лівий нижній кут прямокутника, при NULL-2 — лівий верхній кут); LIBR — ім'я бібліотеки стандартних знаків; NAME — ім'я стандартної моделі поняття (рис. 6).

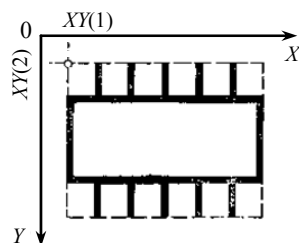


Рис. 6. Завдання стандартної моделі поняття

Для кожної предметної області понять створюються свої бібліотеки стандартних моделей понять. Множина бібліотек має дворівневу структуру, де перший рівень — ім'я бібліотеки стандартних моделей понять, а другий — ім'я стандартної моделі поняття.

На рис. 7 показано структуру бібліотек стандартних моделей понять.

Підпрограма формування тексту поняття. Звернення до підпрограми: GLYPH.

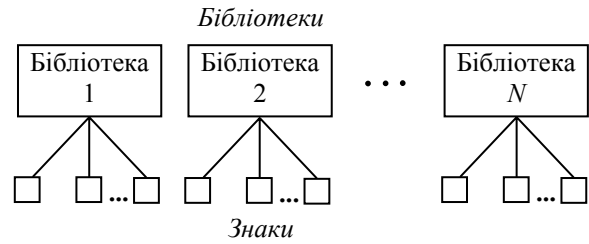


Рис. 7. Структура бібліотек стандартних моделей понять

Параметри: XY — координати точки поля виводу, куди поміщується верхній лівий кут прямокутника, в якому розміщується текст; H — висота шрифту в міліметрах; FI — кут повороту рядка в градусах.

Узагальнений растровий інтерфейс і алгоритми формування понятійного виразу в реальному масштабі часу

Для растрового представлення моделі поняття будемо розбивати модель відображення поняття на множину понятійних виразів і множину позицій у кожному виразі. Кожна позиція може відображатися одним бітом (темне, світле) або набором бітів (найчастіше байтом), наприклад, для кольорового зображення. Для пристроїв з невеликою роздільною здатністю інформацію про усі моделі понять можна зберігати в оперативній пам'яті. Для растрового координатографа це неможливо, оскільки, наприклад, при кроці 0,025 мм і розмірі поля 420×420 мм для зберігання зображення в бітах потрібно 35 280 000 байтів. Для стислого зберігання зображення запропонований узагальнений растровий інтерфейс [3].

Для того щоб сформувати понятійний вираз, необхідно провести горизонтальну пряму і визначити точки перетину цієї прямої з усіма елементами зображення, що виводиться. При використанні апарату аналітичної геометрії це завдання вирішити неважко.

Розглянемо структуру растрового інтерфейсу разом з алгоритмами формування одного понятійного виразу, орієнтованими на роботу в реальному масштабі часу. Поле зображення розбивається на дискрети, причому ціна дискрета дорівнює кроку виводу. Усі параметри елементів в узагальненій растровій мові (координати, радіуси і т.п.) виражені в дискретах. Крім того, поле понятійного зображення розбивається горизонтальними прямими на області, таким чином щоб сусідні області відрізнялися одна від одної наявністю принаймні одного нового елементу (відрізка, кола та ін.).

Точніше, нехай $Y = \{y_i\}_{i=1}^k$ — множина понятійних виразів растрового документа; $A = \{a_j\}_{j=1}^N$ — множина елементів, що утворюють даний документ. Будемо вважати, що вираз $y_i \in Y$ перетинає елемент $a_j \in A$, і позначати $a_j \cap y_i$, якщо у виразі y_i є позиції, що належать елементу a_j . Розглянемо відображення $f: Y \rightarrow A$, так що для $y_i \in Y$ [5; 6].

$$y_i \xrightarrow{f} \{a_j \in A \mid a_j \cap y_i\}.$$

Розглянемо розбиття множини $Y = \bigcup_{l=1}^M Y_l$ на набори послідовних понятійних виразів $Y_l = \{y_s\}_{s=I_l}^{Q_l}$, що володіють такою властивістю: для будь-яких $y_p, y_q \in Y_l$

$$f(y_p) \equiv f(y_q),$$

а при $y_p \in Y_l, y_q \in Y_r (e \neq R)$

$$f(y_p) \neq f(y_q).$$

Узагальнений растровий інтерфейс являє собою опис розбиття множини виразів разом з відображенням f (основний масив) і множину елементів A (масиви елементів).

Основний масив. Синтаксис

<кортеж виразів> ::= <група однакових виразів> <ознака кінця>, [<група однакових виразів> <ознака кінця> ..] <ознака кінця>

<група однакових виразів> ::= <номер початкового виразу> <номер кінцевого виразу> [<опис елемента> ..]

<опис елемента> ::= <тип елемента> <посилання на елемент>

<тип елемента> ::= <відрізок> | <коло> | <стандартний знак> | <дуга> | <текст>

<відрізок> ::= 1

<коло> ::= 2

<стандартний знак> ::= 3

<дуга> ::= 4

<текст> ::= 5

<посилання на елемент> ::= <восьмирічне число>

<ознака кінця> ::= - 1.

Семантика. В основному масиві у впорядкованому по виразах вигляді зберігається інформація про зображення моделі поняття, що виводиться. Інформація стисла за рахунок групування послідовних однакових виразів. Елементами зображення моделі поняття є відрізки, кола, стандартні моделі понять, дуги і текст. Таким чином, у групі однакових виразів може знаходитися множина з вказаних елементів.

Усі елементи впорядковані і зберігаються у відповідних масивах. Посилання на елемент — восьмирічне число, що вказує номер початкового слова в описі даного елемента. Таким чином, опис однакових виразів складається з тежкоподібних моделей понять — покажчика типу моделі поняття і його адреси у відповідному масиві.

Масив відрізків. Уся множина відрізків розбита на сім груп таким чином, що довільний відрізок належить завжди одній з цих груп.

Синтаксис

<список відрізків> ::= <опис відрізка> [<опис відрізка> ..]

<опис відрізка> ::= <ознака викреслювання суцільної лінії> <опис суцільної лінії> | <ознака викреслювання пунктирної лінії> <опис пунктирної лінії> | <ознака викреслювання штрихпунктирної лінії> <опис штрихпунктирної лінії>

<ознака викреслювання суцільної лінії> ::= 0

<ознака викреслювання пунктирної лінії> ::= 0 | 1

<ознака викреслювання штрихпунктирної лінії> ::= 0 | 1 | 2 | 3

<опис суцільної лінії> ::= 1 <опис 1> | 2 <опис 2> | 3 <опис 3–4> | 4 <опис 3–4> | 5 <опис 5–6> | 6 <опис 5–6> | 7 <опис 7>

<опис пунктирної лінії> ::= 8 <опис 8-9> | 9 <опис 8-9> | 10 <опис 10-11> | 11 <опис 10-11> | 12 <опис 12>

<опис штрихпунктирної лінії> ::= 13 <опис 13–14> | 14 <опис 13–14> | 15 <опис 15–16> | 16 <опис 15–16> | 17 <опис 17>

<опис 1> ::= <початкова координата x> <кінцева координата x>

<опис 2> ::= <початкова координата x> <ширина відрізка>

<опис 3–4> ::= <поточна координата x> <величина, обернена кутовому коефіцієнту відрізка> <проекція ширини відрізка на вісь X> <min координата x> <max координата x>

<опис 5–6> ::= <поточна координата x> <кутовий коефіцієнт відрізка> <проекція ширини відрізка на вісь X> <координата y>

<опис 7> ::= <поточна координата x> <+ -> 1 <проекція ширини відрізка на вісь X> <min координата x> <max координата x>

У разі програмної реалізації для економії пам'яті наступні елементи описів є цілими числами: усі елементи описів 1; 2; 7; <поточна координата x> і <проекція ширини відрізка на вісь X> в описах 3–4 і 5–6; <координата y> в описі 5–6, <min координата x> і <max координата x>. Інші елементи є дійсними числами.

<опис 8–9> ::= <опис 3–4> <опис пунктиру>

<опис 10–11> ::= <опис 5–6> <опис пунктиру>

<опис 12> ::= <опис 7> <опис пунктиру>
 <опис пунктиру> ::= <довжина штриха> <довжина пропуску> <лічильник виразів>
 <опис 13–14> ::= <опис 3–4> <опис штрихпунктира>
 <опис 15–16> ::= <опис 5–6> <опис штрихпунктира>
 <опис 17> ::= <опис 7> <опис штрихпунктира>
 <опис штрихпунктира> ::= <довжина штриха> <довжина пропуску> <довжина «точки» > <лічильник рядків>

Елементи <довжина штриха>, <довжина пропуску>, <довжина «точки» > і <лічильник слів> — цілі числа.

Семантика. Нехай $(x_1, y_1), (x_2, y_2)$ — координати початкової і кінцевої точок відрізка. Причому $\Delta x = x_2 - x_1$; $\Delta y = y_2 - y_1$.

Слід мати на увазі, що початкова точка завжди має мінімальну ординату, а при рівних ординатах — мінімальну абсцису. Тоді розбиття відрізків на групи відбувається відповідно до таких співвідношень:

- 1 — $\Delta y = 0$ (горизонтальний відрізок);
- 2 — $\Delta x = 0$ (вертикальний відрізок);
- 3 — $\Delta x > \Delta y$ і $\Delta x / \Delta y > 0$;
- 4 — $|\Delta x| > \Delta y$ і $\Delta x / \Delta y < 0$;
- 5 — $\Delta x < \Delta y$ і $\Delta y / \Delta x > 0$;
- 6 — $|\Delta x| < \Delta y$ і $\Delta y / \Delta x < 0$;
- 7 — $|\Delta x| = \Delta y$ і $\Delta x \neq 0$.

Розглянемо алгоритм визначення для поточного виразу y_i координат x_{in} і x_{ik} — початку і кінця затемненої зони (затемнена зона або ділянка — множина точок відрізка з ординатою y_i). При переході від координатної форми до растрової замість середньої лінії відрізка береться його ліва границя, а права границя отримується складанням проекції ширини відрізка на вісь X і координати, що задає його ліву границю.

Введемо такі позначення даних, що містяться в описі відрізка: x_n — початкова координата затемненої ділянки; x_k — кінцева координата затемненої ділянки; t — ширина відрізка; t_x — проекція ширини відрізка на вісь X ; k — кутовий коефіцієнт відрізка; $m = 1/k$ — величина, обернена кутовому коефіцієнту відрізка; $x_{пт}$ — поточна координата x для рядка, що розглядається; $y_{пт}$ — координата y , якій відповідає поточна координата $x_{пт}$ (для груп 5 і 6).

Для кожної групи використовується свій алгоритм визначення граничних координат затемненої ділянки для поточного рядка i . Дані $x_{пт}$ і $y_{пт}$ модифікуються в процесі обчислення координат затемненої ділянки (x_{in}, x_{ik}). Групи 3 і 5 виділені для випадку додатного кутового коефіцієнта, групи 4 і 6 — для від'ємного.

Розглянемо ці групи докладніше.

Група 1 (рис. 8, а) $x_{in} = x_n$; $x_{ik} = x_k$.

Група 2 (рис. 8, б) $x_{in} = x_n$; $x_{ik} = x_k + t - 1$.

Групи 3 і 4 (рис. 8, в).

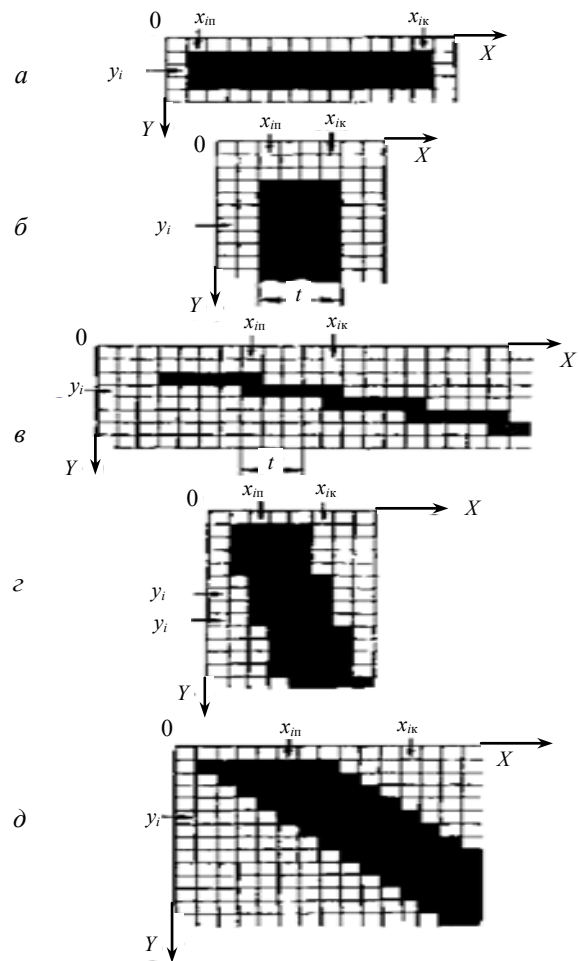


Рис. 8. Інтерполяція відрізків

Слід зазначити, що для груп 3–7 (рис. 8) $x_{пт}$ — поточна координата початку затемненої ділянки для поточного виразу, тобто $x_{in} = x_{пт}$. Значення x_{ik} розраховується за формулою

$$x_{ik} = x_k + t - 1.$$

Після розрахунку x_{ik} , значення $x_{пт}$ має бути модифіковано для такого виразу. Оскільки $x_i = (\Delta x / \Delta y)$, $y_i = m y_i$, то модифікація значення $x_{пт}$ проводиться за виразом

$$x_{i+1к} = m(y_i + 1) = x_i + m.$$

Для того щоб при переході від одного виразу до іншого не було «дірок», доводиться коригувати значення координати x_{ik} .

Для групи 3: якщо $x_{i+1п} < x_{ik}$, то поправка не вводить; інакше $x_{ik} = x_{i+1п}$.

Для групи 4: якщо $x_{i-1п} < x_{ik}$, то поправка не вводить; у протилежному випадку $x_{ik} = x_{i-1п}$.

У групах 5 і 6 (рис. 8, з), як і в попередніх випадках, $x_{in} = x_{пт}$; $x_{ik} = x_{in} + t_x - 1$.

Проте коригування «дірок» проводиться дещо складніше, ніж для груп 3 і 4. Змінна $y_{пт}$ вказує максимальне значення координати y , якій ще відповідає поточна координата $x_{ін}$.

Інакше, координата $x_{ін}$ не змінюється від деякого виразу y' до виразу $y_{пт}$. Величина $y_{пт}$ зберігається незмінною до тих пір, доки не зміниться величина $x_{ін}$. Момент, коли модифікується величина $y_{пт}$, визначається такими умовами.

Для групи 5: якщо $y_{пт} + k \geq y_i - 1$, то $x_{ін}$, $x_{ік}$, $x_{пт}$ і $y_{пт}$ не модифікуються, інакше відбуваються привласнення:

$$x_{ін} = x_{ін} + 1; \quad x_{пт} = x_{ін}; \quad x_{ік} = x_{ік} + 1; \\ y_{пт} = y_{пт} + k.$$

Для групи 6: якщо $y_{пт} - k \geq y_i - 1$, то $x_{ін}$, $x_{ік}$, $x_{пт}$ і $y_{пт}$ не модифікуються; інакше відбуваються привласнення: $x_{ін} = x_{ін} - 1$; $x_{пт} = x_{ін}$; $x_{ік} = x_{ік} - 1$; $y_{пт} = y_{пт} - k$.

Група 7 (рис. 8, д) ідентична групам 3 і 4. Відмінність полягає в тому, що завжди $x_{ін} = x_{пт}$; $x_{ік} = x_{ін} + t_x - 1$. Потім здійснюється модифікація $x_{пт}$. Якщо $k > 0$, то $x_{пт} = x_{ін} + 1$, якщо $k < 0$, то $x_{пт} = x_{ін} - 1$.

Для груп 3, 4 і 7 здійснюється обмеження по вертикалі по крайніх точках відрізка. Якщо $x_{мін}$ і $x_{макс}$ — координати x крайніх точок відрізків і коли $x_{ін} < x_{мін}$, то $x_{ін}$ привласнюється значення $x_{мін}$; якщо $x_{ік} > x_{макс}$, то $x_{ік}$ привласнюється значення $x_{макс}$.

Формування пунктирних і штрихпунктирних відрізків для груп 3–7 здійснюється таким чином. В елементах <довжина штриха>, <довжина пропуску>, <довжина «точки»> зберігаються довжини відповідних елементів у виразах, тобто кількість виразів, що замінюються черговим штрихом, пропуском між штрихами <штрихом і точкою, точкою і штрихом>. В елементі <лічильник виразів> знаходиться відносний номер виразу, що закреслюється. Після досягнення лічильником виразів значення, що дорівнює числу, записаному в поточному елементі <довжина штриха>, <довжина пропуску>, <довжина «точки»>, відбувається перехід до наступного елемента і цикл повторюється.

Послідовність елементів для пунктиру — штрих, пропуск; для штрихпунктиру — штрих, пропуск, точка, пропуск.

Для вертикального відрізка формування пропусків здійснюється пропуском посилань на опи́си відрізків в основному масиві, для горизонтального відрізка — один відрізок розбивається на множину відрізків, що є пунктиром або штрихпунктиром.

Масив кіл. Синтаксис

<вирази кіл> ::= <опис кола> [(опис кола> ..]

<опис кола> ::= <центр> <квадрат зовнішнього радіуса> <квадрат внутрішнього радіуса>

<центр> ::= <координата x > <координата y >.

Конструкції <координата x >, <координата y >, <квадрат зовнішнього радіуса> і <квадрат внутрішнього радіуса> — дійсні числа.

Семантика. Вона не потребує пояснень, оскільки зрозуміла з синтаксису.

Інтерполяція кіл полягає у визначенні для поточного виразу i координат $(x_{i1п}, x_{i1к})$ і $(x_{i2п}, x_{i2к})$ затемнених ділянок. Можливі дві ситуації (рис. 9): рядок $i = j$ перетинає тільки зовнішнє коло; рядок $t = k$ перетинає як зовнішнє, так і внутрішнє кола. У першій ситуації є одна затемнена ділянка, в другому випадку — дві.

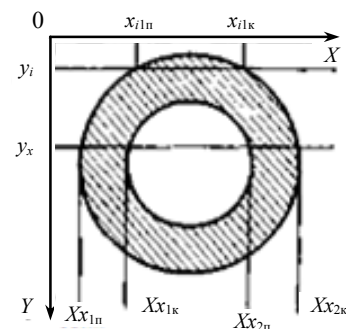


Рис. 9. Інтерполяція кіл

Розглянемо алгоритм інтерполяції. Введемо позначення: x_0, y_0 — координати центра кола; R_1, R_2 — квадрати радіусів зовнішнього і внутрішнього кіл; y_i — координата y поточного рядка.

Крок 1. Визначення координат перетину рядка y_i із зовнішнім колом. Якщо $D_1 = R_1 - (y_i - y_0)^2 \geq 0$, то

$$x_{i1п} = x_0 - \sqrt{D_1}; \\ x_{i2к} = x_0 + \sqrt{D_1},$$

інакше перехід до кроку 4.

Крок 2. Визначення координат перетину рядка y_i з внутрішнім колом. Якщо $D_2 = R_2 - (y_i - y_0)^2 \geq 0$, то

$$x_{i1к} = x_0 - \sqrt{D_2}; \\ x_{i2п} = x_0 + \sqrt{D_2},$$

інакше перехід до кроку 5.

Крок 3. Перехід до кроку 6.

Крок 4. Формування ознаки про відсутність перетину рядка y_i з зовнішнім колом і перехід до кроку 6.

Крок 5. Оскільки рядок y_i не перетинає внутрішнє коло, то $x_{i1к}$ привласнюється значення $x_{i2к}$, а $x_{i2п}$ і $x_{i2к}$ обнуляються.

Крок 6. Кінець алгоритму.

Таким чином, алгоритм зводиться до таких «довгих» операцій: два здобуття кореня і одне множення (зведення в квадрат).

Масиви дуг. Дуги представлені двома масивами: каталог і описи дуг.

Каталог дуг. Синтаксис

<каталог дуг> ::= <каталог для однієї дуги> [<каталог для однієї дуги> ..]

<каталог для однієї дуги> ::= <кількість виразів в записі даної дуги> <адреса опису дуги в масиві описів дуг> <радіус зовнішнього кола> <радіус внутрішнього кола> <адреса, що модифікується> <максимальний номер виразу, для якого істина ознака> <ознака> [<максимальний номер виразу, для якого істина ознака> <ознака>] [<максимальний номер виразу, для якого істина ознака> <ознака>]

<ознака> ::= 1 | 2 | 3.

Семантика. Кожна дуга розбивається горизонтальними виразами не більше ніж на три ділянки залежно від характеру перетину дуги горизонтального виразу. Кожній ділянці привласнюється ознака: 1 — права частина дуги; 2 — ліва частина дуги; 3 — обидві частини дуги (рис. 10).

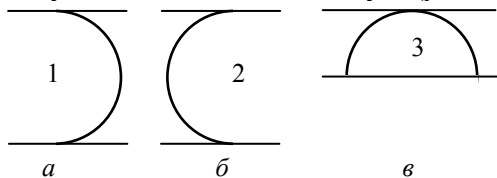


Рис.10. Розбиття дуги на ділянки:
а — права частина дуги; б — ліва частина дуги;
в — обидві частини дуги

Конструкція <кількість слів у записі даної дуги> являє собою довжину <у виразах> записи в каталозі для даної дуги, оскільки для кожної дуги записи мають змінну довжину.

Конструкція <адреса опису дуги в масиві описів дуг> являє собою адресу в масиві, де знаходиться опис кола, частина якого збігається з даною дугою.

Конструкція <адреса, що модифікується> — це посилання на одну з трьох можливих конструкцій <максимальний номер виразу, для якого діє ознака>. У процесі виводу ця адреса модифікується. Оскільки дуга розбита на 1, 2 або 3 ділянки, то конструкція <максимальний номер виразу, для якого діє ознака> вказує верхню границю для кожної ділянки, тобто максимальну координату у, що належить даній ділянці.

На рис. 11, а, б наведено приклад дуги і її каталог відповідно.

Описи дуг. Синтаксис і семантика масиву опису дуг повністю збігаються з описом масиву кіл. Як і для кіл, інтерполяція дуг полягає у визначенні для цього виразу і координат $(x_{i1п}, x_{i1к})$ і $(x_{i2п}, x_{i2к})$ затемнених ділянок.

Алгоритм інтерполяції складається з двох частин: визначення для поточного виразу і ознаки дуги; апроксимація дуги як кола і вибір $x_{i1п}, x_{i1к}$ і $x_{i2п}, x_{i2к}$.

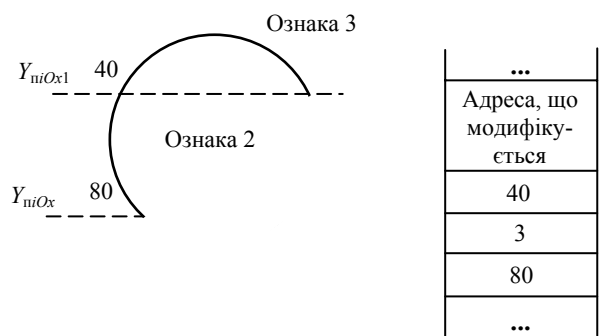


Рис. 11. Приклад кодування дуги

Для ознаки 1 значення координат $x_{i2п}, x_{i2к}$ привласнюються змінним $x_{i1п}, x_{i1к}$, потім змінні $x_{i2п}, x_{i2к}$ обнуляються.

Для ознаки 2 визначаються значення $x_{i1п}, x_{i1к}$. Змінні $x_{i2п}, x_{i2к}$ обнуляються.

Для ознаки 3 значення змінних $x_{i1п}, x_{i1к}, x_{i2п}$ і $x_{i2к}$ визначаються так само, як і для кіл.

Таким чином, час інтерполяції дуг того ж порядку, що і час інтерполяції кіл.

Масиви стандартних моделей понять

Стандартні моделі понять (СМП) представлені двома масивами: каталог СМП і бібліотека СМП.

Каталог СМП. Синтаксис

<каталог> ::= <позиція> <адреса> <слово, що модифікується> [<позиція> <адреса> <слово, що модифікується> ..]

<позиція> ::= <координата x> <координата y>

<адреса> ::= <восьмирічне число>

<слово, що модифікується> ::= <восьмирічне число>

Семантика. СМП вписується в прямокутник, що має N рядків і K стовпців. Конструкція <позиція> — це координати верхнього лівого кута прямокутника, в який вписана СМП. Конструкція <адреса> — є адреса в масиві описів СМП даного СМП. У слові, що модифікується, завжди записана адреса початку кодування поточної групи однакових рядків СМП у бібліотеці СМП.

Бібліотека СМП. Синтаксис

<опис стандартних моделей понять> ::= <опис СМП> [<опис СМП>...]

<опис СМП> ::= <розмір СМП по осі X, бітах> <розмір СМП по осі Y, бітах> <список груп однакових рядків>

<список груп однакових виразів> ::= <група однакових виразів > | <група однакових виразів > <список груп однакових виразів >

<група однакових виразів> ::= <номер початкового виразу> <номер кінцевого виразу> <число затемнених ділянок> <позиція> <число затемнених дискретів>

Усі конструкції в описі стандартних моделей понять — цілі числа.

Семантика. Масив бібліотек стандартних моделей понять побудований аналогічно основному масиву, тобто увесь прямокутник, що обмежує стандартну модель поняття, розбитий на групи однакових виразів (I, II, III, IV, V на рис. 12).

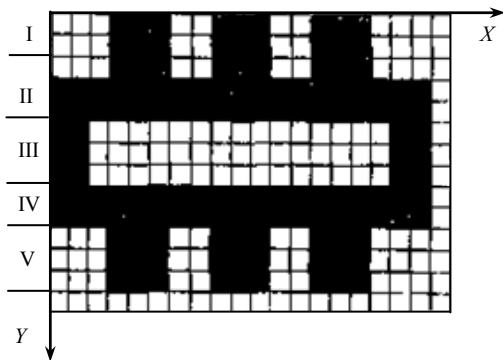


Рис. 12. Інтерполяція стандартних знаків

Інтерполяція СМП полягає у визначенні для даного виразу i координат (x_{in}, x_{ik}) для кожної затемненої ділянки, що потрапляє в цей вираз.

Алгоритм інтерполяції такий.

Крок 1. За модифікованим словом каталога СМП визначається початкова адреса поточної групи однакових виразів у бібліотеці СМП.

Крок 2. Перевірка: чи належить поточній групі однакових виразів вираз, що формується. Якщо — так, то перехід до кроку 3; у протилежному випадку — зміна виразу, що модифікується, для переходу до наступної групи однакових виразів і перехід до кроку 1.

Крок 3. Визначення числа затемнених ділянок у виразі, що формується.

Крок 4. Визначення координат поточної затемненої ділянки (x_{in}, x_{ik}) .

Крок 5. Перевірка: чи сформовані усі затемнені ділянки. Якщо — так, то перехід до наступної ділянки і кроку 4; інакше — перехід до наступного кроку.

Крок 6. Кінець алгоритму.

Масиви тексту

Текст представлений двома масивами: каталог тексту і бібліотека СМП.

Каталог тексту. **Синтаксис**

<каталог тексту> ::= <опис виразу> [| <опис виразу> ..]

<опис виразу> ::= <число символів у виразі> <координата x першого символу виразу> <координата y першого символу виразу> <слово 1, що модифікується> <адреса символу> <слово 2, що модифікується> [<адреса символу> <слово 2, що модифікується> ..]

Семантика. В основному масиві вказується адреса опису конкретного виразу в каталозі тексту. При цьому в один вираз входять тільки ті

символи, які мають однакову висоту (великі і рядкові букви входять у різні рядки).

Конструкція <слово 1, що модифікується> є координатою x поточного символу виразу; у початковий момент це координата x першого символу виразу. Конструкція <слово 2, що модифікується> така сама, як і в каталозі стандартних моделей понять. Бібліотека символів така сама, як і бібліотека стандартних моделей понять.

Таким чином, текст зводиться до стандартних моделей понять, причому кожен символ є окремою стандартною моделлю поняття. Інтерполяція тексту полягає в тому, що за каталогом тексту послідовно вибираються кодування для поточних виразів символів виразів. Кожен наступний символ виразу розміщується поряд з попереднім, що забезпечується включенням у кодування символу проміжку між буквами.

Вибірка кодувань символу з бібліотеки СМП і вписування його в буфер виразу відбувається за тим самим алгоритмом, що і для СМП.

Висновки

Представлена структурна схема універсального математичного забезпечення створення математичних моделей понять надає можливість формувати і виводити растрову інформацію про математичні моделі понять.

Описане математичне забезпечення орієнтоване на випуск документації про сформовані математичні моделі понять, що являють собою набір текстових і графічних елементів (графічні зображення понять, креслення, текстові документи та ін.). Запропонований алгоритм формування понятійного виразу надає можливість створювати математичну модель поняття в реальному масштабі часу.

ЛІТЕРАТУРА

1. Кононюк А. Ю. Введення в математичну теорію системи науково-визначуваних понять / А. Ю. Кононюк, І. В. Малінкін, В. О. Малярчук // Проблеми інформатизації та управління: зб. наук. праць. — К. : НАУ, 2013. — Вип. 4(44). — С. 53–63.
2. Кононюк А. Ю. Аксиоматичний підхід побудови фрагментів математичної теорії системи науково-визначуваних понять / А. Ю. Кононюк, І. В. Малінкін, В. О. Малярчук // XVII міжнар. наук.-практ. конф. з проблем видавничо-поліграфічної галузі, 19 груд. 2013 р.: тези доп. — К., 2013. — С. 65–68.
3. Кононюк А. Е. Общая теория систем. К. 1 / А. Е. Кононюк. — К. : Освіта України, 2012. — 548 с.
4. Кононюк А. Е. Общая теория информации. К. 1 / А. Е. Кононюк. — К. : Освіта України, 2012. — 430 с.
5. Кононюк А. Е. Дискретна математика. К.1 / А. Е. Кононюк. — К. : Освіта України, 2011. — Ч. 1: Множества, отношения, пространства (четкие и нечеткие). — 2011. — 450 с.