

АДАПТИВНИЙ СИНТЕЗ ФОРМУЛ АБСТРАКТНИХ АЛГОРИТМІВ

© Василюк А.С., Басюк Т.М., 2012

Описано процеси адаптації формул алгоритмів. Наведено алгоритм комп'ютерної адаптації формул алгоритмів. Синтезовано, мінімізовано, побудовано математичну модель і досліджено алгоритм адаптації базового знаку операції.

Ключові слова: адаптація, адаптивний синтез, алгоритм, математична модель.

This article is about determination adaptation processes of algorithms formulas. The adaptation algorithm of formulas is resulted. Synthesized, minimized, and built mathematical model and probed adaptation algorithm of operation base sign.

Keywords: adaptation, adaptive synthesis, algorithms, mathematical model.

Вступ. Постановка проблеми

Відома [1,2] алгебра алгоритмів, яка має оригінальні операції, наприклад, секвентування, елімінування, паралелення та циклічні операції, котрі позначаються спеціальними знаками, яких немає серед відомих математичних знаків. Для набору та редагування формул абстрактних алгоритмів розроблено спеціалізовану комп'ютерну підсистему МОДАЛ [3], але вона не виконує автоматично адаптацію формул абстрактних алгоритмів.

У таких графічних пакетах, як Microsoft Visio, Corel DRAW, Adobe Illustrator не реалізовано систему адаптування формул алгоритмів. Без реалізації таких функцій процеси редагування та набору формул алгоритмів значно ускладнюються.

Аналіз останніх досліджень та публікацій

У роботах [5–7] описуються підходи до розв'язання описаних задач, проте за всієї різноманітності у жодному з наведених досліджень автори не проаналізували та не описали основних задач, які виникають за спроби адаптувати формули алгоритмів.

За всієї актуальності завдання сьогодні накопичено відносно невеликий досвід його вирішення, який насамперед визначається відносно новим напрямком досліджень.

Формулювання цілей статті

Як відомо, за спроби набору та редагування формул алгоритмів засобами відомих інформаційних систем виникають ускладнення. Основна причина в тому, що ці системи не реалізують процес адаптації, без якого фактично не можливо коректно відтворити формулу алгоритму. Ціллю дослідження є синтез, дослідження та побудова математичного забезпечення процесу адаптації формул алгоритмів.

Аналіз отриманих наукових результатів**Ілюстрація адаптації формул абстрактних алгоритмів**

Процес адаптації формул абстрактних алгоритмів є складовою частиною процесу набору формул. Цей процес повинен виконуватися після здійснення користувачем будь-яких змін у редакторі, тобто адаптація є автоматичним процесом, який користувач безпосередньо інструментальними засобами не викликає.

Вербальне подання алгоритму розв'язання квадратного рівняння $ax^2 + bx + c = 0$ є таким: знаходять дискримінант квадратного рівняння D за формулою: $D = b^2 - 4ac$, якщо $D > 0$, то $x_{1,2} = (-b \pm \sqrt{D})/2a$, якщо $D = 0$, то $x_1 = x_2 = -b/2a$, якщо $D < 0$, то дійсних коренів немає. Подання цього алгоритму блок-схемою наведено на рис. 1.1, де $P = a$ – присвоєння p першого коефіцієнта a , $Q = b$ – присвоєння q другого коефіцієнта b , $R = c$ – присвоєння r третього коефіцієнта c , обчислення

дискримінанта квадратного рівняння $D = b^2 - 4ac$. За виконання умови $D > 0$ виконується обчислення коренів квадратного рівняння $x_{1,2} = (-b \pm \sqrt{D})/2a$, за виконання умови $D = 0$ виконується обчислення коренів квадратного рівняння $x_1 = x_2 = -b/2a$, за виконання умови $D < 0$ квадратне рівняння дійсних коренів не має.

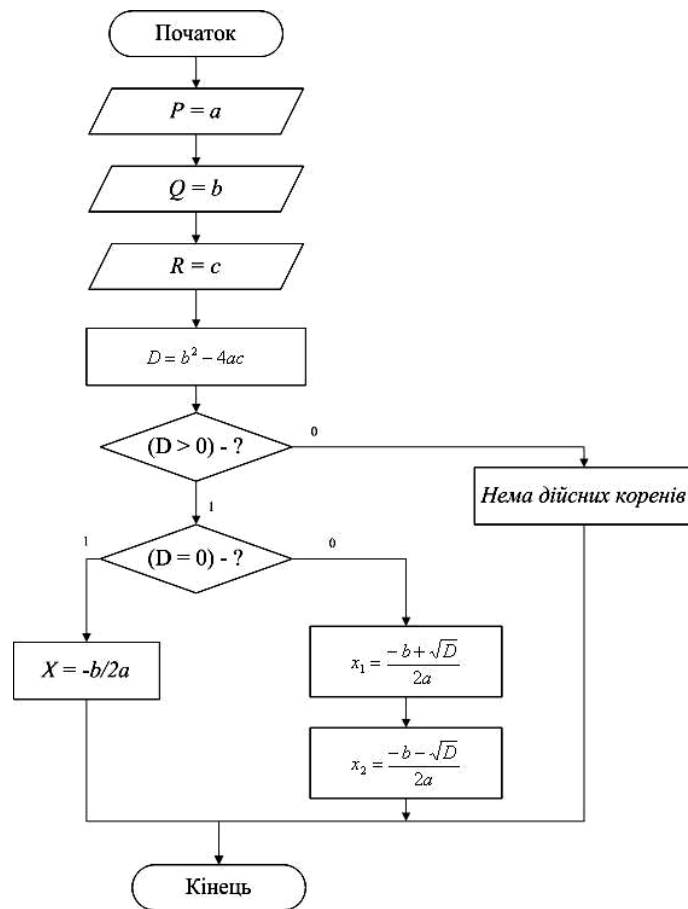


Рис. 1. Блок-схема алгоритму знаходження коренів квадратного рівняння

Алгоритм знаходження коренів квадратного рівняння поданий засобами алгебри алгоритмів буде мати вигляд:

$$\left(\begin{array}{l} p = a \\ q = b \\ r = c \\ D = b^2 - 4ac \\ \left(\begin{array}{l} x_1 = (-b + \sqrt{D})/2ac ; \text{Немає дійсних коренів} ; (D > 0) - ? \\ x_2 = (-b - \sqrt{D})/2ac \end{array} \right) ; x = -b/2a ; (D \neq 0) - ? \end{array} \right. \quad (1)$$

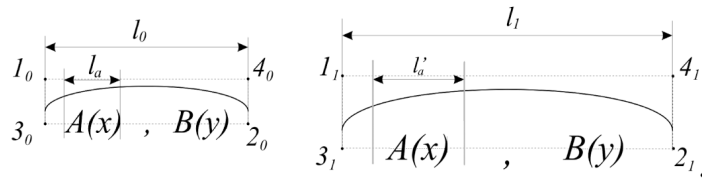
У формулі (1) $\overbrace{\hspace{2cm}}$ – знак операції секвентування, $\overline{\hspace{2cm}}$ – знак операції елімінування, $p = a, q = b, r = c, D = b^2 - 4ac, X = -b/2a, x_1 = (-b + \sqrt{D})/2a, x_2 = (-b - \sqrt{D})/2a, (D \neq 0) - ?, (D > 0) - ?, *Немає дійсних коренів$ – унітерми. Знаки операцій можуть бути різними за розмірами залежно від розмірів вкладених формул. Для коректного відображення формул абстрактних алгоритмів необхідно обчислювати ці розміри і, враховуючи їх, виконувати процес адаптації.

Суть процесу адаптації формул алгоритмів

Адаптацією називатимемо процес зміни розмірів знаків операцій залежно від розмірів, розташування (згори/знизу), орієнтації (вертикальна/горизонтальна), кегля, верхніх та нижніх індексів унітермів, а також знаків операцій алгебри алгоритмів.

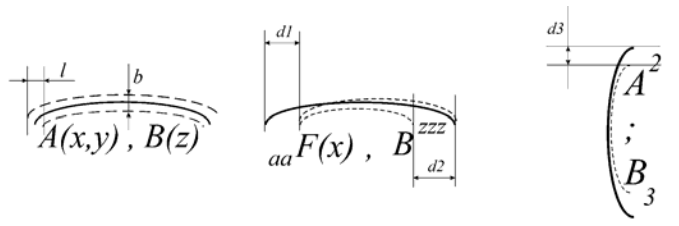
Задачі адаптації:

а) залежно від розмірів шрифту



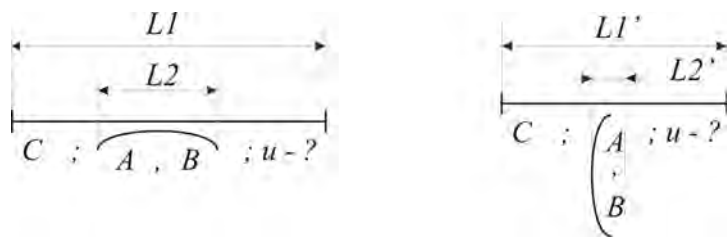
де l_0 – довжина формули до адаптації; l_a – довжина вкладеної формули; l_1 – довжина формули після адаптації; l_a' – довжина вкладеної формули після зміни кегля; $l_0, 2_0, 3_0, 4_0$ – точки побудови знака операції до зміни розмірів знака, $l_1, 2_1, 3_1, 4_1$ – точки побудови знака операції після зміни розмірів знака;

б) з врахуванням індексів



де l – діапазон зміни розміру знака операції по горизонталі; b – діапазон зміни розташування знака операції по вертикалі; d_1 – діапазон зміни координат розташування формули по горизонталі; d_2 – діапазон зміни розмірів формули по горизонталі; d_3 – діапазон зміни координат розташування формули по вертикалі;

в) з врахуванням вертикальної або горизонтальної орієнтації знаків операцій

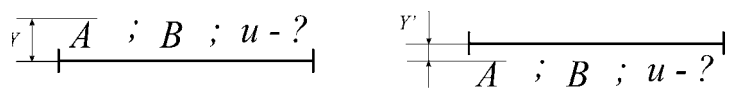


де L_1 – довжина формули; L_2 – довжина вкладеної формули; L_1' – довжина формули після адаптації; L_2' – довжина формули після зміни орієнтації;

г) залежно від розташування унітермів;

1) згори

2) знизу



де Y – відстань за висотою від знака операції до унітерму у разі розташування згори, Y' – відстань за висотою від знака операції до унітерму у разі розташування знизу.

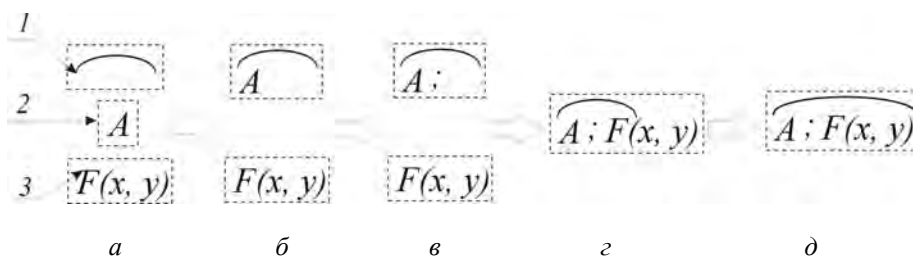


Рис. 2. Схема етапів процесу адаптації

На рис. 2,а наведено три складові формули алгоритму: 1 – базовий знак операції секвентування; 2 – перший вкладений унітерм A ; 3 – другий вкладений унітерм $F(x, y)$. Процес адаптації складається з таких етапів:

1. Набір трьох складових формули алгоритму (базового знака операції та вкладених формул);
2. Переміщення першої вкладеної формули (2) під базовий знак операції (1);
3. Введення розділювача (“ , ” або “ ; ”);
4. Переміщення другої вкладеної формули (3) під базовий знак операції (1);
5. Зміна розмірів базового знаку операції (1) залежно від геометричних параметрів вкладених формул (2) і (3). Схематично ці процеси показано на рис. 3,а,б.

Зміна розмірів знаку операції залежно від геометричних параметрів унітермів, своєю чергою, складається з таких процесів (рис. 3):

1. Ідентифікація базової формули;
2. Зчитування поточних початкових і кінцевих точок знаку операції;
3. Адаптація базового знаку операції залежно від типу ідентифікованого знаку операції (секвентування, елімінування, паралелення та циклічних операцій).



Рис. 3. Структурна схема етапів процесу зміни розмірів знаків операції залежно від геометричних параметрів унітермів

Процес адаптації формул абстрактних алгоритмів є складовою частиною процесу набору формул. Цей процес повинен виконуватися після здійснення користувачем будь-яких змін у редакторі, тобто адаптація є автоматичним процесом, який користувач безпосередньо інструментальними засобами не викликає. Процес адаптації базового знаку операції секвентування враховує нелінійне співвідношення його довжини і ширини та залежить від розмірів шрифту унітермів, наявності чи відсутності індексів, типу орієнтації знаків операцій та від розташування вкладених формул.

На рис. 4 зображено адаптацію базового знаку операції секвентування із врахуванням геометричних параметрів вкладених унітермів, де $x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4$ – абсциси і ординати першої, другої, третьої та четвертої точок побудови знаку операції секвентування до адаптації, $x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4$ – абсциси і ординати першої, другої, третьої та четвертої точок побудови знаку операції секвентування після адаптації, l – довжина знаку операції секвентування до адаптації, l_a – довжина знаку операції секвентування після адаптації.

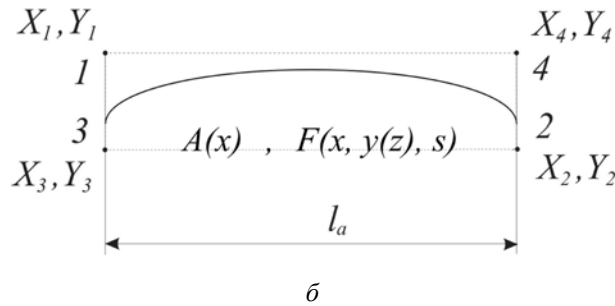
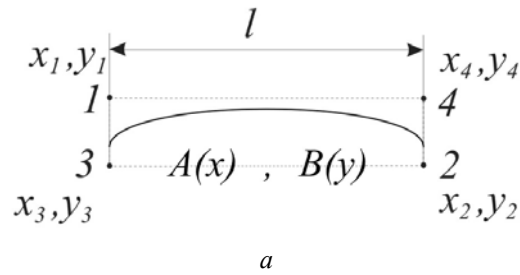


Рис. 4. Адаптація базового знака операції секвенчування до унітермі:
а – до адаптації; б – після адаптації

Загальний абстрактний алгоритм процесу адаптації формул алгоритмів

Синтез секвенцій. Секвенція, яка описує процес адаптації за горизонтальної орієнтації формули алгоритму і за наявності першої вкладеної формули має такий вигляд:

$$S_1 = \overbrace{P_1(l_a, Z_1); F_1; F_2}$$

де Z_1 – позначення алгоритму зміни адаптованих геометричних розмірів за наявності першої вкладеної формули і горизонтальної орієнтації базового знака операції; F_1 – алгоритм адаптації будь-якої базової формули до вкладеної, F_2 – алгоритм переміщення вкладеної формули до базової.

Процес адаптації за горизонтальної орієнтації формули алгоритму і за наявності другої вкладеної формули описується такою секвенцією:

$$S_2 = \overbrace{P_1(l_a, Z_2); F_1; F_2}$$

де Z_2 – позначення алгоритму зміни адаптованих геометричних розмірів за наявності другої вкладеної формули і горизонтальної орієнтації базового знака операції.

Секвенція, яка описує процес адаптації за вертикальної орієнтації формули алгоритму і за наявності першої вкладеної формули має такий вигляд:

$$S_3 = \overbrace{P_1(H_a, Z_3); F_1; F_2}$$

де Z_3 – позначення алгоритму зміни адаптованих геометричних розмірів за наявності першої вкладеної формули і вертикальної орієнтації базового знака операції.

Процес адаптації за вертикальної орієнтації формули алгоритму і за наявності другої вкладеної формули описується такою секвенцією:

$$S_4 = \overbrace{P_1(H_a, Z_4); F_1; F_2}$$

де Z_4 – позначення алгоритму зміни адаптованих геометричних розмірів за наявності другої вкладеної формули і вертикальної орієнтації базового знака операції.

Синтез елімінувань. Секвенції S_1 і S_2 елімінуються за умовою перевірки на наявність першої вкладеної формули (u_2). Отримаємо такий вираз:

$$L_1 = \overline{S_1 ; S_2 ; u_2^- ?}$$

За умовою перевірки на наявність першої вкладеної формули (u_2) елімінуємо секвенції S_3 і S_4 . Отримаємо таку формулу алгоритму:

$$L_2 = \overline{S_3 ; S_4 ; u_2^- ?}$$

Елімінування L_1 і L_2 елімінуємо за умовою перевірки на тип орієнтації, в результаті чого отримаємо таку формулу:

$$L_3 = \overline{E_1 ; E_2 ; u_1^- ?}$$

Після підстановки секвенцій у відповідні елімінування отримаємо такий абстрактний алгоритм загального алгоритму адаптації формул:

$$\overline{\left(\begin{array}{c} P_1(l_a, Z_1) \\ ; \\ F_1 \\ ; \\ F_2 \end{array} ; \begin{array}{c} P_1(l_a, Z_2) \\ ; \\ F_1 \\ ; \\ F_2 \end{array} ; u_2^- ? \right) ; \left(\begin{array}{c} P_1(h_a, Z_3) \\ ; \\ F_1 \\ ; \\ F_2 \end{array} ; \begin{array}{c} P_1(h_a, Z_4) \\ ; \\ F_1 \\ ; \\ F_2 \end{array} ; u_2^- ? \right) ; u_1^- ?}$$

На підставі властивості дистрибутивності знака операції елімінування алгебри алгоритмів вносимо унітерми F_1 і F_2 за знак операції елімінування. Отримаємо таку формулу:

$$A = \left(\overline{\begin{array}{c} P_1(l_a, Z_1) \\ ; \\ P_1(l_a, Z_2) \\ ; \\ u_2^- ? \end{array} ; \begin{array}{c} P_1(h_a, Z_3) \\ ; \\ P_1(h_a, Z_4) \\ ; \\ u_2^- ? \end{array} ; u_1^- ? \right) ; \left(\begin{array}{c} F_1 \\ ; \\ F_2 \end{array} \right)$$

Алгоритм адаптації
 будь-якої базової
 формули до вложеної

Інтерфейс користувача редактора АбстрактАл

Інтерфейс користувача редактора формул абстрактних алгоритмів містить: головне меню (“Алгоритм”, “Редагувати”, “Зображення”, “Вікно” та “?”), кожне з яких розгалужується на свої підменю. Меню “Алгоритм” має такі підменю: “Новий”, “Відкрити”, “Зберегти”, “Зберегти як...”, “Закрити”, “Експорт”, “Параметри”, “Параметри друку”, “На друк” та “Вихід”. Меню “Редагувати” має такі підменю: “Хід назад”, “Хід вперед”, “Вирізати”, “Копіювати”, “Вставити”, “Вставити із заміною”, “Знищити”, “Мінімізувати модуль алгоритму”, “Мінімізувати модель”, “Спрощення із використанням допоміжної змінної”, “Генерувати математичну модель”, “Секвенційний (лінійний) алгоритм” та “Покрокове згортання”. Меню “Зображення” має такі підменю: “Шрифт”, “Колір”, “Товщина лінії” та “Адаптація”. Меню “Вікно” має такі підменю: “Панель інструментів”, “Параметри алгоритму”, “Впорядкувати вікна”, “Обчислювач формул”, “Структура алгоритму” та “КоБА (Комп’ютерна бібліотека абстрактних алгоритмів)”.

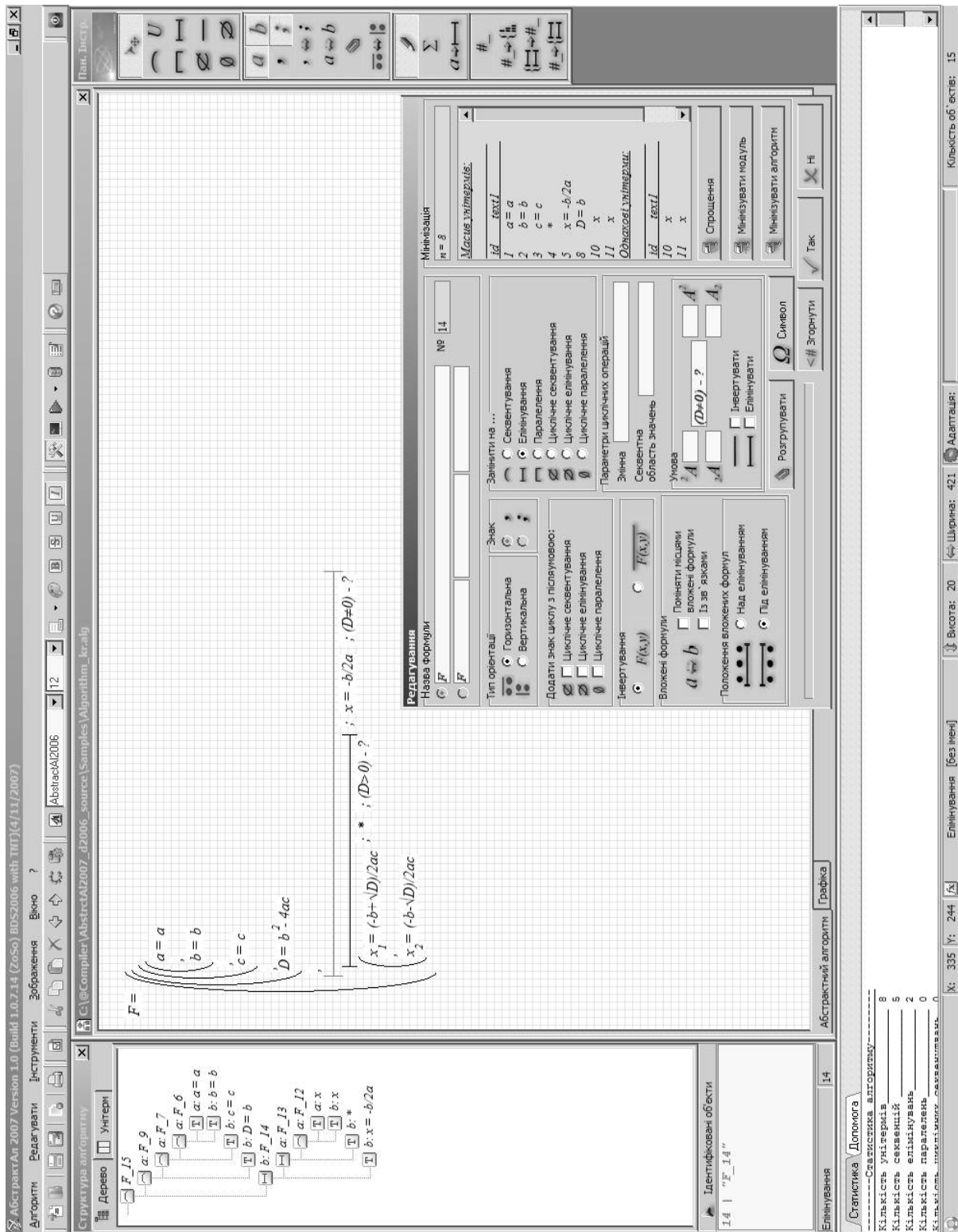


Рис. 5. Інтерфейс користувача редактора формул абстрактних алгоритмів «АбстрактАл»

Меню “?” має такі підменю: “Про...”, “Швидкі клавіші” та “Допомога”.

Панель інструментів являє собою набір інструментальних засобів для набору та редагування формул абстрактних алгоритмів. Вона містить 25 елементів типу “кнопка”, кожна з яких відповідає за певну операцію чи дію. Важливу роль відіграють діалогові вікна набору та редагування знаків операцій та унітермів. Загалом редактор абстрактних алгоритмів налічує 14 вікон, кожне з яких несе своє функціональне навантаження. Нижче показано головне вікно редактора разом з допоміжними вікнами. Робочою поверхнею редактора буде об’єкт класу TCanvas, що автоматично створюється для всіх видимих компонентів, зокрема і для PaintBox. На “канві” можна рисувати такими засобами, як: Pen, Brush чи Font, вона також володіє численними методами, які будуть необхідні для подальшого розроблення.

Дослідження процесів набору та адаптації формул абстрактних алгоритмів

Після запуску редактора абстрактних алгоритмів “АбстрактАл” на екрані монітора з’явиться головне вікно редактора. Нехай необхідно набрати засобами редактора таку формулу:

$$\left(\begin{array}{l} A \\ , \\ \hline F_1(a) ; x^2 ; u_1 - ? \end{array} \right)$$

Дії, які необхідно здійснити для набору заданої формули:

1. Для набору унітермів вибираємо інструмент “Вставити унітерм”;
2. Засобами діалогового вікна “Вставити унітерм” набрати необхідні унітерми;
3. Вибрати інструмент “Елімінування” і в будь-якому місці робочої зони цокнути лівою клавішею миші. На екрані відобразиться знак операції елімінування;
4. Ідентифікувати знак операції елімінування та за допомогою інструментів “Перший вкладений унітерм” та “Другий вкладений унітерм” адаптувати формулу абстрактного алгоритму.
5. Вибрати інструмент “Секвентування” і в будь-якому місці робочої зони клацнути лівою клавішею миші. На екрані відобразиться знак операції секвентування;
6. Ідентифікувати знак операції секвентування та за допомогою інструментів “Перший вкладений унітерм” та “Другий вкладений унітерм” адаптувати формулу абстрактного алгоритму.

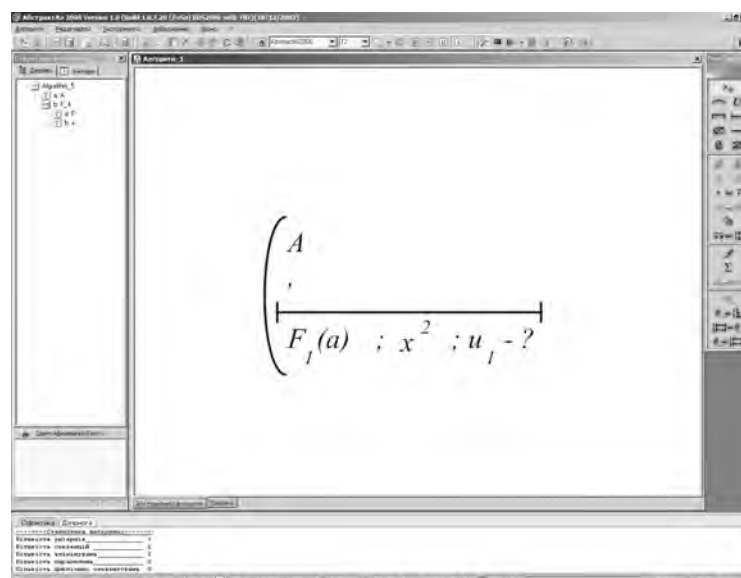


Рис. 6. Синтез формул абстрактних алгоритмів

Висновки

1. Синтезована, мінімізована і досліджена математична модель алгоритму адаптації формул абстрактних алгоритмів описує ідентифікації підалгоритмів адаптації базових операцій до вкладених формул і вкладених формул до базових, реалізація якої забезпечує більшу наочність подання алгоритмів у вигляді формул теорії абстрактних алгоритмів.

2. Дослідження математичної моделі ще до її практичної реалізації і апробації забезпечило виявлення помилок, допущених у процесі її синтезу та доводить, що вона описує необхідні процеси.

3. Абстрактним алгоритмом адаптації формул абстрактних алгоритмів описано процеси адаптивного синтезу формул алгоритмів.

1. Овсяк В., Бритковський В., Овсяк О., Овсяк Ю. Синтез і дослідження алгоритмів комп'ютерних систем. – Львів, 2004. – 276 с. 2. Овсяк В.: методи побудови, оптимізації, дослідження вірогідності. – Львів: Світ, 2001. – 160 с. 3. Бритковський В. М. Моделювання редактора формул секвенційних алгоритмів. Автореф. дис. ... канд. техн. наук. – Львів: видавничо-поліграфічний відділ ЛвЦНТЕІ. – 18 с. 4. Математическая энциклопедия: Гл. ред. И. М. Виноградов. Т.3 – М.: “Советская энциклопедия”, 1982. – 1184 с. 5. Овсяк В., Василюк А. Принцип побудови підсистеми редагування формул абстрактних алгоритмів // Комп'ютерні технології друкарства – 2004. – №12. – С. 137–146. 6. Василюк А. Абстрактний алгоритм редактора формул абстрактних алгоритмів “АбстрактАл” / А. Василюк // Комп'ютерні технології друкарства : Збірник наукових праць. – Львів: УАД, 2006. – №16. – С. 99–108. 7. Овсяк В., Василюк А. // Комп'ютерні технології друкарства. – 2004. – №12. – С. 137–145.