

**О.І. ІВАНЕШКІН**

Міжнародний науково-навчальний центр інформаційних технологій та систем  
НАН та МОН України, Київ, Україна,  
e-mail: [al.ivaneshkin@gmail.com](mailto:al.ivaneshkin@gmail.com).

**УНІВЕРСАЛЬНА ІНФОРМАЦІЙНА SOFTWARE-ТЕХНОЛОГІЯ РОБОТИ  
З НЕОРІЄНТОВАНИМИ ЗМІШАНИМИ ЛІСАМИ**

**Анотація.** Створено нову універсальну інформаційну software-технологію, реалізовану у комплексі програмних засобів та призначену для виконання широкого, що легко поповнюється, набору базових операцій на полікомпонентних структурах, які включають неорієнтовані дерева та саджанці. Вона дає змогу працювати з об'єктами, що містять до 65536 вершин, які можуть бути розташовані на 6400  $X$ -рівнях, і потребує (з урахуванням власних номерів) лише 4 байти інформації на кожен вершину. Завдяки широким функціональним та прикладним можливостям технологія не вимагає застосування матриць і списків як засобів формалізованого представлення досліджуваних об'єктів. Для її використання немає потреби в опануванні значної частки раніше отриманих результатів. Вона зводить до мінімуму участь користувача і багаторазово скорочує витрати технічних ресурсів та часу в процесі дослідження.

**Ключові слова:** граф-дерево, неорієнтоване дерево, саджанець, змішаний ліс, комплекс програмних засобів.

**ВСТУП**

Стрімке розширення наукових і практичних сфер діяльності людини зумовлює збільшення кількості задач, розв'язання яких сприяє розвитку теорії графів-дерев [1] і робить її одним з найважливіших математичних інструментів отримання відповідей на численні актуальні питання. Пов'язані з цією теорією методи ефективно використовують у теоріях планування, керування, прийняття рішень, розкладів та ігор, лінгвістиці, інформатиці, програмуванні, хімії, генетиці тощо. Однак цим методам притаманна велика складність реалізації ( $N^2$ ) і на практиці вони є вельми громіздкими засобами, які використовують різні формалізовані представлення дерев (матриці, списки). Створені на базі теорії графів-дерев методи і підходи до роботи з деревами є широко відомими і визнаними та вже багато десятиліть не змінюються. Через відсутність ефективніших аналогів їх досі активно використовують, незважаючи на суттєві витрати технічних ресурсів і часу.

Зазначені обставини зумовлюють актуальність розв'язання комплексу задач, загальних для різних сфер науково-прикладної діяльності, та розроблення нових, основаних на нестандартних підходах інформаційних software-технологій. Останні не важко опанувати, вони зручні для отримання та наочного представлення результатів у звітах, їх можна оперативно з малими витратами модифікувати, що значно збільшує їхній користувацький потенціал.

Мета запропонованої роботи — розв'язання задачі створення універсальної високоефективної інформаційної software-технології та її реалізація в комплексі програмних засобів (КПЗ), що забезпечують можливість виконання над більш загального вигляду неорієнтованими графами-деревами максимальної кількості різноманітних операцій (зокрема, пошук характеристик) в єдиному, інтегрованому користувацькому середовищі, легкому для розуміння, опанування та практичного використання.

## ОПИС ОБ'ЄКТА ДОСЛІДЖЕНЬ ТА ОТРИМАНІ РЕЗУЛЬТАТИ

Розвиваючи сформульований у [2, 3] напрямок досліджень, наводимо результати досягнення поставленої мети для загальнішого типу об'єктів — полікомпонентних неорієнтованих графів без циклів або змішаних лісів (mixed forests)  $MF(T_I; S_J)$  [4].

Результати розв'язання низки задач було інтегровано в створену універсальну software-технологію роботи з такими об'єктами, реалізованою на практиці мовою MS VISUAL C++ 6 у другій версії КПЗ. Значно розширюючи функціонально-прикладні можливості першої версії узагальненням раніше отриманих результатів, збільшенням кількості виконуваних операцій та їхньої якісної різноманітності, нова версія забезпечила можливість працювати як з окремими компонентами та елементами  $MF(T_I; S_J)$ , так і з усією їхньою сукупністю одночасно.

Візуальна ідентичність структури зазначених об'єктів з реальними деревами зумовила закономірне використання в теорії графів низки термінів («корінь», «дерево», «гілка», «лист», «ліс») із області знань, пов'язаної з лісовим господарством. Дотримуючись правил однотипності та уніфікації понятійного апарату, що використовується для зручності викладу та сприйняття суті отриманих результатів, доцільно доповнити зазначений перелік термінів низкою понять, які належать тій самій предметній області і мають інтуїтивно усвідомлюваний зміст.

Для узгодження цілочислових значень координат  $(x, y)$  ( $0 \leq x; 0 \leq y$ ) положення вершин  $\{A\}$  дерева на горизонтальній ( $X$ ) та вертикальній ( $Y$ ) віртуальних осях екрана ( $(0,0)$  — координати кореня) та порядкових номерів елементів масивів даних, які у більшості мов програмування починаються цифрою 0, уведемо низку означень для загального вигляду  $N$ -вершинного ( $1 < N$ ) дерева та його виродженого представника — ізольованої вершини ( $N \equiv 1$ ):

**висота (height) дерева  $h$**  ( $0 \leq h$ ) — максимальне значення  $y$ -координати ( $y$ -рівня) вершини, що входить до структури дерева (у графічній інтерпретації дерева на екрані  $h \equiv N - 1$ );

**ширина (width) дерева  $w$**  ( $0 \leq w$ ) — максимальне значення  $x$ -координати ( $x$ -рівня) вершини дерева у його графічному зображенні на екрані;

**саджанець (sapling)** — позбавлене всіх гілок дерево з  $N = 1$ ,  $h = 0$  (у термінах теорії графів — ізольована вершина степеня  $d = 0$ , що не має предка та нащадків);

**змішаний ліс** — двопараметричний об'єкт вигляду  $MF(T_I; S_J)$ , де  $T_I = \{t_i\}_{i=1}^I$  — множина (ліс), що включає  $I$  ( $1 \leq I$ ) довільного вигляду дерев  $t_i$  (trees), а  $S_J = \{s_j\}_{j=1}^J$  — множина  $J$  ( $1 \leq J$ ) саджанців  $s_j$  (saplings);

**компонент** — будь-який  $t_i \in MF(T_I; S_J)$  або  $s_j \in MF(T_I; S_J)$ ;

**елемент** — вершина, або гілка з коренем, що входить до складу компонента;

**монопетворення** (локальна операція) — сукупність операцій, локально і послідовно виконаних на конкретному  $t_i$  або  $s_j$  без використання елементів інших  $\{t_k\}_k$  ( $k \neq i$ ) та  $\{s_l\}_l$  ( $l \neq j$ ) із  $MF(T_I; S_J)$ ;

**крос-перетворення** (перехресна операція) — сукупність операцій, послідовно виконаних на множині компонентів  $\{t_i\}_i \subseteq T_I$  і  $\{s_j\}_j \subseteq S_J$  з використанням елементів тих самих підмножин  $\{t_i\}_i$  і  $\{s_j\}_j$ .

Якщо  $Y$ -порядок прямування  $t_i$  та  $s_j$  один за одним (згори донизу по осі  $Y$ ) регламентований умовами задачі, яка має бути розв'язана,  $MF(T_I; S_J)$  назвати-

мемо впорядкованим, інакше — невпорядкованим. Легко бачити, що відображення один на одного будь-яких пар об'єктів  $\{MF(T_I^0; S_J^0); MF(T_I^1; S_J^1)\}$ , отриманих з  $MF(T_I; S_J)$  у результаті  $(I + J)!$  «вертикальних»  $Y$ -перестановок компонентів з  $\{t_i\}_{i=1}^I$  та  $\{s_j\}_{j=1}^J$ , є бієктивним та ізоморфним (бієктивно-ізоморфним). Крім того, виконання над окремими компонентами  $t_i \in T_I$  будь-якої кількості ізоморфних перетворень за допомогою операції генерування ізоморфних структур [3] зберігає бієктивну ізоморфність усієї множини отриманих таким чином лісів  $\{MF(T_I^q; S_J^q)\}_q$  і робить замкнутою їхню сукупність відносно зазначеної пари перетворень.

Для уніфікації можливостей КПЗ у його наступну версію буде включено операції приєднання «attach» та від'єднання «detach», що відповідають додаванню на визначеному місці та вилученню із зазначених місць окремих промаркованих лісів або компонентів.

Далі як аналог поняття «змішаний ліс» будемо використовувати слово «об'єкт», а вибраний для роботи  $t_i \in T_I$  ( $s_j \in S_J$ ) будемо називати «робочим» (або «активним»).

Для збереження здатності низки операцій ефективно розв'язувати задачі під час роботи з  $MF(T_I; S_J)$  ( $1 < I$ ) створено засоби локалізації потрібного  $t_i$  та пов'язаних з ним даних. Зокрема, цього неможливо уникнути під час виконання таких операцій:

- перенумерування вершин;
- відшукування вершини за власним номером;
- ізоморфне реконфігурування  $MF(T_I; S_J)$  (побудова ізоморфного аналога);
- оцінювання ізоморфності пари довільного вигляду об'єктів  $MF(T_I; S_J)$ .

Тож кожного разу під час виконання на робочому компоненті  $t_i$  будь-яких операцій будемо брати до уваги, що він і його дані вже були попередньо вибрані за допомогою реально розроблених і вбудованих у КПЗ спеціалізованих програмних засобів.

Також зазначимо, що у другій версії комплексу всі саджанці  $s_j \in MF(T_I; S_J)$  та корені  $t_i \in MF(T_I; S_J)$  (за своєю суттю — вершини нульового  $X$ -рівня) отримали однакові права з усіма вершинами, що входять до складу відповідних  $t_i$ , і унаслідок чого їх можна використовувати у всіх реалізованих у КПЗ операціях.

Головні особливості реалізації в попередній версії КПЗ раніше створеного підходу до роботи з неорієнтованими зв'язними деревами було викладено в [2, 3]. Тож далі будемо наводити лише ті результати, які сприяли значному розширенню функціонально-прикладних можливостей підходу, його постійному оперативному та маловитратному вдосконаленню, надали статус універсальної software-технології інтегрованому користувацькому середовищу, яке програмно реалізоване у новій версії КПЗ.

На рис. 1 показано восьмикомпонентний 53-вершинний  $MF(T_3; S_5)$ . Об'єкт спочатку було створено, а потім модифіковано за допомогою низки описаних далі операцій, кнопки яких помічено темнішим кольором, і радіокнопку зі знаком +. Цей об'єкт містить три дерева (ROOT TREES № 3, № 4, № 8 з 17, 7, 24 вершинами відповідно) та п'ять саджанців (TREE SAPLINGS № 1, № 2, № 5, № 6, № 7). Сірим кольором полів прямокутників та виразом «Mark = \*» у рядку додаткових параметрів промарковано 12 вершин.

Розглянемо операції, що включено у нову (другу) версію КПЗ (див. рис. 1).

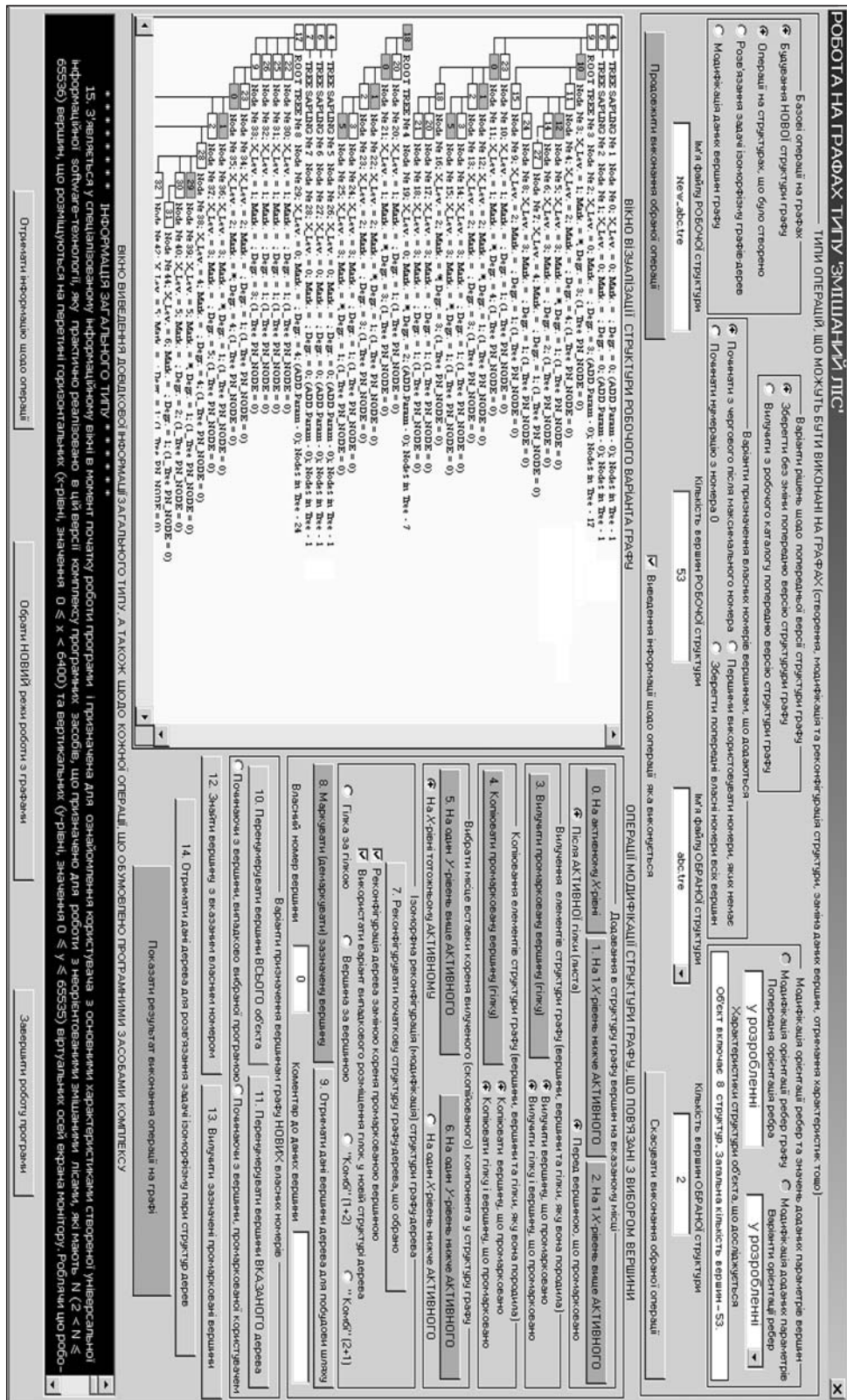


Рис. 1. Вигляд інтерфейсу користувача другої версії КІП

## РЕЖИМ БУДУВАННЯ НОВОЇ СТРУКТУРИ ГРАФУ

У новій версії КПЗ реалізовано можливість починати будовання структури  $MF(T_I; S_J)$  ( $1 \leq I + J$ ) з початкового генерування в ній довільної  $K$  ( $1 \leq K < 100$ ) кількості саджанців та подальшого (зокрема, без переривання сеансу роботи з об'єктом) перетворення деякої їхньої кількості  $r$  ( $1 \leq r$ ) у повноцінній, заданої структури дерева за допомогою виконання реалізованих у комплексі операцій.

## РЕЖИМИ МОДИФІКАЦІЇ СТРУКТУРИ ГРАФІВ

**Операції 0, 1, 2. Додавання до структури графу вершин на вказаному місці.** Ці операції призначено для збільшення загальної кількості вершин робочого  $t_i \in T_I$  ( $s_j \in S_J$ ) з використанням вбудованих у КПЗ спеціалізованих засобів їхнього генерування.

У створеній версії комплексу стало можливим застосовувати зазначені три операції до будь-яких компонентів  $MF(T_I; S_J)$ , а також у випадку, коли орієнтиром місця додавання до структури нових вершин є промаркований корінь дерева або саджанець, що мають координати  $(0, v)$ .

Якщо в процесі додавання  $q$  вершин використовують радіокнопку з написом «Перед промаркованою вершиною» (саджанцем), то їх буде розміщено на нульовому  $X$ -рівні безпосередньо над промаркованим коренем. Зі збереженням підпорядкованості на  $q$   $Y$ -рівнів донизу будуть зсунуті всі вершини, які входять до робочого компонента, а також вершини всіх компонентів  $MF(T_I; S_J)$ , що візуально розташовані нижче робочого.

З використанням радіокнопки «Після активної гілки» (саджанця) всі  $q$  вершин буде розміщено на нульовому  $X$ -рівні безпосередньо після всіх елементів активного компонента і зсунуто донизу на  $q$   $Y$ -рівнів лише вершини компонентів  $MF(T_I; S_J)$ , що візуально розташовані нижче робочого.

Згідно з «початковою безособовістю»  $q$  вершин, які генеруються і додаються комплексом до структури, в цих операціях для надання їм упорядкованих за зростанням власних номерів можуть використовуватися три розроблені варіанти нумерації:

- 1) починати з чергового після максимального номера (який має місце в структурі робочого компонента);
- 2) першими використовувати номери, яких немає у робочому компоненті та які не перевищують максимального власного номера, наявного в структурі;
- 3) починати нумерацію з номера 0.

Зазначимо, що використання варіанта 3 призводить до дублювання власних номерів, уникнути цього дають змогу операції перенумерації 10 або 11.

**Операція 3. Вилучити промарковану вершину (гілку).**

**Операція 4. Копіювати промарковану вершину (гілку).**

У наступній версії КПЗ цю тимчасово розділену в інтерфейсі пару операцій буде інтегровано із спільною назвою.

Оскільки під час маркування вершини степеня  $d$  ( $1 < d$ ) промаркованою вважається також і породжена нею гілка [2], що містить  $r$  ( $1 \leq r$ ) вершин з урахуванням її кореня, суть кожної операції стає зрозумілою безпосередньо з назви і тому не потребує додаткових пояснень.

У новій версії КПЗ для зручності візуального контролю після вибору операції та активізації радіокнопки, яка констатує прийняте рішення, поля прямокутників окремої вершини або всіх вершин гілки, які підлягають копіюванню або вилученню зі структури активного компонента, набувають жовтого кольору.

В операціях, що пов'язані з вилученням із структури об'єкта окремих компонентів, скасовано принципову заборону на використання їхніх коренів для переведення в категорію «АКТИВНИХ». Тепер, якщо  $1 \leq I + J$ , дозволено маркування будь-якого кореня та саджанця для вилучення з  $MF(T_I; S_J)$  усього робочого компонента та його наступні вставки на потрібних місцях структури об'єкта. У процесі виконання таких операцій КПЗ видає попереджувальне повідомлення щодо можливих наслідків, що дає змогу уникнути порушення зв'язності робочого компонента.

Збереження у пам'яті всіх даних  $r$  вилучених (скопійованих) вершин до моменту виконання чергової операції, відмінної від операцій 3 і 4, забезпечує можливість багаторазово їх вставляти у будь-які компоненти  $MF(T_I; S_J)$  на будь-яких зазначених користувачем місцях. Для забезпечення візуального контролю коректності виконання операції поля всіх вершин, що вставлені у структуру зазначеного компонента, також тимчасово позначають жовтим кольором.

**Операції 5, 6. Вибрати місце вставки кореня вилученого (скопійованого) компонента у структуру графу.** Завдяки здатності нової версії КПЗ працювати з  $MF(T_I; S_J)$  стало реальним розроблення нових операцій, аналогами яких є відомі методи, що досі широко застосовують, і які за функціональними можливостями подібні операціям виробництва саджанців дерев. Інформаційний аналог одного з таких методів (метод черенкування) було створено, програмно реалізовано і включено до складу модулів, що реалізують цю пару операцій.

Зазвичай місце вставки раніше вилученого або скопійованого елемента вказує користувач маркуванням відповідної вершини  $MF(T_I; S_J)$ . До того ж, якщо орієнтиром місця є корінь будь-якого  $t_i \in T_I$  або будь-який  $s_j \in S_J$  з координатами  $(0, v)$ , то вершини вибраного  $t_i$ , починаючи з промаркованої і нижче, будуть візуально переміщені на  $r$  ( $1 \leq r$ )  $Y$ -рівнів донизу. У «порожнечі», що тимчасово утворюється, буде розташований  $r$ -вершинний елемент, який має бути вставлений. Він стане новим «самостійним» компонентом об'єкта і збільшить на одиницю значення  $I$  або  $J$ . Корінь вставленого в  $MF(T_I; S_J)$  елемента з координатами  $(0, v)$  візуально буде розташований на місці раніше промаркованої вершини. Інші вершини, що входять у елемент, розташуються на координатах  $\{(m, n)_{m=1}^{r-1}\}_{n=v+1}^{v+r-1}$ , зберігши раніше регламентовану взаємну співвідлеглість, значення додаткових параметрів і показників.

Під час виконання таких операцій стає можливим ще один варіант призначення вершинам власних номерів, який був реалізований у КПЗ. Для констатації функціональної спільності з раніше наведеною в операціях 0, 1, 2 сукупністю варіантів 1, 2, 3 йому доцільно присвоїти черговий номер та сформулювати таким чином варіант 4 — зберегти попередні власні номери всіх вершин (такі, що були до виконання операції копіювання або вилучення).

У більшості випадків використання варіанта 4 може призвести до дублювання власних номерів і уникнути цього допоможе раніше вказаний спосіб перенумерації вершин.

Візуально коректність виконання операцій підтверджується забарвленням жовтим кольором полів прямокутників вершин, які були вставлені у структуру.

Застосування цієї операції до раніше вилученого з  $MF(T_I; S_J)$  компонента дає можливість «простої» зміни місця його розташування всередині об'єкта (окремий випадок ізоморфної реконфігурації всього об'єкта) або перенесення у інші компоненти зі зміною значень  $I$  та  $J$ .

**Операція 7. Реконфігурувати початкову структуру вибраного графу-дерева.** Як і в попередній версії КПЗ, розв'язання задачі побудови ізомор-

фного аналога об'єкта обмежилося застосуванням операції до конкретного  $t \in T_I$ . Оскільки  $\{s_j\} \in S_J$  є саджанцями, вони перетворення не потребують.

Це зумовлено відсутністю об'єктивних, коректно переконливих доказів того, що взагалі має сенс розв'язання цієї задачі одразу для всього  $MF(T_I; S_J)$ . Доцільність цього зауваження можна пояснити тим, що за допомогою операції 7 [3], яка вже налагоджена та ефективно працює, за потреби і у довільному порядку можна побудувати ізоморфні аналоги  $\{\bar{t}_k\}_{k=1}^K$  усіх компонентів сукупності  $\{t_k\}_{k=1}^K \subseteq T_I$ , а потім, застосувавши операцію інтегрування даних у загальному файлі, отримати один із варіантів змішаного лісу, що є ізоморфним аналогом усього  $MF(T_I; S_J)$ .

Розширення функціональних можливостей цієї операції було досягнуто завдяки виявленню та практичної програмної реалізації двох варіантів ізоморфної реконфігурації довільного компонента  $t_i \in T_I$ . Суть першого варіанта полягає у заміні діючого кореня компонента  $t_i$  на будь-яку вершину, що входить до його структури. Другий варіант можна охарактеризувати як «локальний». Він призначений для реконфігурування лише однієї з гілок  $t_i$ , що вибирається маркуванням її кореня.

В обох варіантах, крім чотирьох раніше описаних у [2, 3] типів модифікації, стало можливим використовувати ще один (п'ятий) варіант, що пов'язаний з рандомізованим розміщенням (перетасовуванням) гілок в елементі, який модифікується.

Контроль коректності виконання операції здійснюється за допомогою візуального аналізу пари структур, які виводяться на екран одна під іншою після завершення операції. Нижньою структурою є початкова. Її вершину, що позначено жовтим кольором, вибрано як новий корінь для побудови верхньої структури, яка є ізоморфно модифікованим аналогом нижньої.

#### РЕЖИМИ РОЗВ'ЯЗАННЯ ДОПОМІЖНИХ ЗАДАЧ НА ГРАФАХ

**Операція 8. Маркувати (демаркувати) зазначену вершину.** Ця операція забезпечує маркування червоним кольором (демаркування — кольором фону) поля прямокутника будь-якої вершини, що входить до  $MF(T_I; S_J)$ , зокрема, для вилучення групи вершин із структур компонентів  $\{t_i\} \in T_I$  за допомогою описаної далі операції 13.

**Операція 9. Отримати дані вершини дерева для побудови шляху.** На відміну від описаної в [2] операції, що формувала шлях від кореня активного компонента до вибраної вершини, яка йому належить, ця операція призначена для аналізу та запам'ятовування даних пари довільних вершин, які є кінцевими пунктами шляху, що буде формуватися. Оскільки розв'язання цієї задачі можливе тільки на зв'язній, властивій будь-якій  $t_i \in T_I$  структурі, контроль полягає в строгому виконанні цієї вимоги та видачі повідомлення у разі ненавмисної спроби користувача її порушити. Кнопка з написом «Отримати дані вершини для побудови шляху» має бути використана двічі в процесі конкретизації кожної з вершин, які пропонуються, а вибір дерева для побудови на ньому шляху має відбуватися синхронно з маркуванням першої з них.

Для демонстрації можливостей КПЗ, що реально виявляються лише у разі великої кількості вершин, на рис. 2 наведено результат побудови шляху між вершинами  $A = \text{Node } \# 38$  та  $B = \text{Node } \# 56$  (координати  $A(12, 38)$  та  $B(7, 56)$ ) 102-вершинного  $MF(T_1; S_0)$ . Жирними лініями помічено сторони прямокутників, які позначають вершини, і ребра шляху. Сірим кольором позначено поля прямокутників промаркованих вершин. У нижньому вікні інтерфейсу цей шлях конкретизовано власними номерами вершин та інцидентними до них ребрами.





**Операція 10. Перенумерувати вершини ВСЬОГО об'єкта.** Ця операція призначена для вилучення зі структури об'єкта  $MF(T_I; S_J)$  усіх власних номерів, що повторюються у різних компонентах, або впорядкування вершин для зручності роботи користувача. Маючи «загальний» характер, вона може бути визначена як «глобальна» перенумерація всіх вершин  $MF(T_I; S_J)$ .

**Операція 11. Перенумерувати вершини ВКАЗАНОВОГО дерева.** Ця операція є локальною і призначена для перенумерації вершин об'єкта, що належать компоненту  $t_i \in T_I$ , який отримав статус активного.

Програмні реалізації операції на всьому  $MF(T_I; S_J)$  і зазначеному  $t_i \in T_I$  є принципово різними, тому виконуються різними програмними модулями. На відміну від раніше єдиного варіанта перенумерації, до цієї версії КПЗ включено два варіанти надання вершинам нових власних номерів, що використовуються операціями 10 і 11 (див. рис. 2):

- починаючи з вершини, випадково вибраної програмою;
- починаючи з вершини, промаркованої користувачем.

Надання вершинам нових номерів у вигляді натуральних чисел  $n$  ( $n=0, 1, 2, \dots, N$ ) починається з місця, яке вказане вибраним варіантом, виконується строго за порядком зверху донизу по колу і виключно в межах тієї структури, яка є регламентованою вибраною кнопкою.

**Операція 12. Знайти вершину з вказаним власним номером.** Ця операція є особливо корисною у разі великої кількості вершин, що входять в об'єкт, коли користувачеві необхідно дізнатися статус конкретної вершини в структурі або ознайомитися з низкою її параметрів і додаткових характеристик. Важливо, що призначення власного номера є необхідною, але недостатньою умовою для однозначності пошуку. Без виконання операцій перенумерації і у  $MF(T_I; S_J)$ , і навіть, у конкретному  $t_i \in T_I$  можлива будь-яка кількість вершин, які мають ідентичні власні номери. Якщо цього не зробити, операція виявить візуально першу зверху вершину, власний номер якої було занесено користувачем у поле редактора з назвою «Власний номер вершини».

**Операція 13. Вилучити зазначені промарковані вершини.** Ця операція призначена для одночасного вилучення із  $MF(T_I; S_J)$  чи конкретного  $t_i \in T_I$  усіх або частини вершин, що є промаркованими і за будь-яких причин не повинні мати місце у їхній структурі. Водночас з вершинами вилучаються всі дані, які пов'язані з ними, що унеможливує їхнє відновлення

Для всього  $MF(T_I; S_J)$  і окремих  $t_i \in T_I$  досліджено та програмно реалізовано два варіанти (всього — чотири) розв'язання цієї задачі, а саме:

а) вилучити всі вершини, що промарковані, в тому числі лише промарковані  $\{s_r\}_{r=1}^R \in S_J$  та  $\{t_q\}_{q=1}^Q \in T_I$ ;

б) вилучити всі вершини, що промарковані, за винятком промаркованих  $\{s_r\}_{r=1}^R \in S_J$  та  $\{t_q\}_{q=1}^Q \in T_I$ .

У разі використання варіанта а) частину  $t_i$  буде перетворено у сукупність окремих дерев та саджанців зі збереженням  $Y$ -порядку слідування власних номерів вершин, який існував у  $MF(T_I; S_J)$  до виконання операції.

#### РЕЖИМ РОЗВ'ЯЗАННЯ ЗАДАЧІ ІЗОМОРФІЗМУ ГРАФІВ

**Операція 14. Отримати дані дерева для розв'язання задачі ізоморфізму пари структур дерев.** Отримання розв'язку задачі ізоморфізму пари довільного вигляду  $MF(T_I; S_J)$  є більш ресурсовитратним, ніж технічно складним. Створених способів, методів та алгоритмів для цього достатньо і версія КПЗ із [2, 3] вже ефективно розв'язує цю задачу для будь-яких неорієнтованих  $\{MF(T_1^0; 0), MF(T_1^1; 0)\}$ . Розв'язок буде отримано, коли стане очевидно теоретична та користувачька актуальність задачі. Тому ця операція представлена у КПЗ у своєму первісному вигляді.

Наразі задача модифікації додаткових даних вершин на стадії розв'язання.

Виконання будь-якої операції та отримання результату відбувається лише після натискання кнопки з написом «Показати результат виконання операції на дереві». Скасування раніше вибраної операції здійснюється повторним натисканням відповідної кнопки перед натисканням кнопки, що приводить до безпосереднього виконання цієї операції.

Максимальна кількість віртуальних  $Y$ -рівнів графічного представлення  $MF(T_I; S_J)$  на екрані (вершин графу) тимчасово не перевищує величини 65536, а горизонтальних  $X$ -рівнів — 6400. Ці значення є достатніми для більшості випадків, що мають бути практично реалізовані. У разі постійного зберігання вкрай важливих для багатьох застосунків власних номерів створені способи, методи, алгоритми, засоби побудови та подальшої всебічної роботи з такими об'єктами вимагають всього 4 байти інформації на кожен вершину. Безнадлишкові дані довільного вигляду  $N$ -вершинного  $MF(T_I; S_J)$  зберігаються в найпростішого вигляду бінарному файлі обсягу  $4N$  байт. На рис. 1 і 2 додаткові параметри наведено правіше прямокутників вершин.

#### ВИСНОВКИ

З використанням розроблених підходів, методів, способів та алгоритмів створено нову універсальну інформаційну software-технологію, яка є програмно реалізованою у комплексі програмних засобів і призначена для виконання широкого, що допускає збільшення, набору базових операцій на змішаному лісі, який містить сукупності неорієнтованих дерев і саджанців. За потреби всього 4 байти інформації на кожен вершину (з урахуванням власних номерів) ця технологія дозволяє працювати з об'єктами, що містять до 65536 вершин, які розташовані на 6400  $X$ -рівнях, вибраними з них компонентами та елементами, а також з усією сукупністю останніх одночасно. Зберігаючи всі переваги першої версії КПЗ і значно розширюючи її функціонально-прикладні можливості, software-технологія ще суттєвіше знижує рівень загальних інтегральних витрат технічних ресурсів і часу в процесі розв'язання задач із різних предметних областей знань.

#### СПИСОК ЛІТЕРАТУРИ

1. Ахо А., Хопкрофт Дж., Ульман Дж. Построение и анализ вычислительных алгоритмов. Москва: Мир, 1979. 536 с.
2. Ivaneshkin A.I. A new approach to operating with undirected trees. *Cybernetics and Systems Analysis*. 2021. Vol. 57, N 1. P. 124–132.
3. Ivaneshkin A.I. The solution of the question of the Isomorphism of non-oriented trees by the method of generating isomorphic structures. *Journal of Automation and Information Sciences*. 2020. Vol. 52, N 8. P. 68–79.
4. Иванешкин А.И. Универсальные многофункциональные средства для работы с поликомпонентными структурами неориентированных деревьев и изолированных вершин. Пр. IX-ї Міжнар. наук.-практ. конф. *Проблеми інформатики та комп'ютерної техніки — ПІКТ-2020* (Чернівці, 28–31 жовтня 2021 р.). Чернівці, 2021. С. 95–98.

#### A.I. Ivaneshkin

##### UNIVERSAL INFORMATION SOFTWARE TECHNOLOGY FOR NON-ORIENTED MIXED FORESTS

**Abstract.** A new information software technology has been created, which is implemented in a software package and is destined to perform a wide, easily supplemented set of basic operations on polycomponent structures, including non-oriented trees and saplings. It provides the ability to work with objects containing up to 65536 nodes that can be placed on 6400  $X$ -levels and requires (including personal numbers) only 4 bytes of information for each node. Possessing wide functional and applied capabilities, the technology excludes the need to use matrices and lists as means of formalized representation of the objects under study. For its use, it does not require the development of a significant part of previously obtained results, minimizes user participation and greatly reduces the cost of technical resources and time in the research process.

**Keywords:** graph-tree, non-oriented tree, sapling, mixed forest, a set of software tools.

*Надійшла до редакції 10.01.2022*