УДК 510.6, 681.3.06

М.С. Нікітченко, С.С. Шкільняк

КОМПОЗИЦІЙНІ ЛОГІКИ НОМІНАТИВНИХ ДАНИХ*

Пропонуються логіки, орієнтовані на специфікації програм. Логіки будуються в семантико-синтаксичному стилі на основі композиційно-номінативного підходу. Перша з них (неокласична композиційна логіка) є логікою еквітонних предикатів над нескінченними іменними множинами, вона зберігає основні властивості класичної логіки предикатів. Друга логіка (над скінченними номінативними даними) є конкретизацією першої, на її основі визначається клас багатозначних натурально (абстрактно) обчислюваних функцій над номінативними даними.

Вступ

Математична логіка і теорія обчислюваності, які формують математичні підвалини теорії програмування, також знаходять у програмуванні одне з потужних джерел їхнього власного розвитку. В даній статті ми визначаємо і досліджуємо спеціальні логіки, що виникають як формалізми специфікацій програм. Ми також розглядаємо спеціальне поняття абстрактної обчислюваності і вивчаємо його зв'язок із запропонованими логіками. Побудову аксіоматичних систем над номінативними даними було започатковано в [1, 2]. Такі системи визначались на базі класичної логіки предикатів. В даній статті викладаються результати відповідних досліджень, проведених в рамках композиційно-номінативних логік.

Формальні методи побудови програм звичайно пропонують послідовне зведення абстрактних специфікацій програм до більш конкретних специфікацій. Для ефективного використання таких методів повинні бути визначені адекватні формалізми специфікацій програм різних рівнів абстракції. Відомі мови специфікацій, наприклад *VDM*, *RSL*, *Z* та *B*, базуються на традиційному теоретико-множинному підході до програмних формалізмів, при цьому здебільшого використовується аксіоматична теорія множин Цермело—Френкеля. Застосування такого могутнього формалізму для побудови програм дає змогу досить ефективно вирішувати деякі прикладні проблеми. Але багато питань відносно точних визначень програм (особливо аспекту недетермінованості програм) і використання логічних формалізмів для мов специфікацій залишаються відкритими. Це робить актуальним розвиток нових підходів, що ведуть до побудови більш адекватних формалізмів специфікацій програм. Пропонована стаття базується на композиційно-номінативному підході [3, 4] до формалізації поняття програми.

1. Композиційно-номінативний підхід

Основна мета підходу полягає в побудові ієрархії формальних моделей програм різних рівнів абстракції та загальності. Такий підхід може розглядатися як подальший розвиток композиційного програмування, запропонованого [5] В.Н. Редьком.

Для полегшення розуміння роботи наведемо основні визначення і поняття.

Композиційно-номінативні системи. Поняття композиційно-номінативної системи (КНС) є основним математичним поняттям композиційно-номінативного підходу. Така система може розглядатися як трійка простіших систем: композиційної, дескриптивної, денотаційної. Композиційна визначає семантичні аспекти програм, дескриптивна — дескрипції або описи програм (синтаксичні аспекти), денотаційна — значення (денотати) дескрипцій. У даній статті подамо тільки визначення композиційної системи.

* Робота виконана в рамках проекту INTAS 2000-447.

Під *композиційною системою* будемо розуміти кортеж $Cm = <D, Fn, C>$, де D — множина даних; $Fn \subseteq D \xrightarrow{m} D$ — деяка множина часткових багатозначних функцій; C — множина композицій на Fn.

КНС можуть трактуватися як спеціальні мовні системи для визначення різних класів функцій. Такі системи тісно пов'язані з (нетрадиційними) алгебрами даних і функцій.

Для багатозначної функції запис $f(a){\downarrow}=r$ означає, що f може бути визначена на a (пишемо $f(a){\downarrow}$) з результатом r; запис $f(a){\uparrow}$ означає, що f може бути невизначеною на a.

Множину натуральних чисел будемо позначати Nat.

Під (частковим багатозначним) *предикатом* розуміємо довільну часткову багатозначну функцію вигляду $p: D \xrightarrow{m} Bool$. Тут $Bool = \{T, F\}$ — множина істиннісних значень.

КНС можуть використовуватися для побудови формальних моделей різноманітних класів програм та мов маніпулювання даними. Моделі програм, подані КНС, математично прості, але визначають семантику програм досить адекватно. Такі моделі параметричні, що дає змогу природним чином подавати програми різних рівнів абстракцій. Окрім того, є можливість подати на основі КНС поняття спеціальної абстрактної обчислюваності та різних аксіоматичних формалізмів. КНС можна класифікувати відповідно до рівнів абстракції їхніх параметрів: даних, функцій, композицій, імен.

Номінативні дані. Дані розглядаються на трьох рівнях: абстрактному, булевому і номінативному [3, 4]. Дані третього рівня називаються *номінативними* даними. Вони будуються індуктивно із множини імен V та множини базових значень W. Дамо рекурсивне визначення класу номінативних даних:

$$ND(V, W) = W \cup (V \xrightarrow{m} ND(V, W)).$$

Номінативні дані будемо записувати у вигляді $d = [v_i \mapsto a_i \mid i \in I]$. Відношення номінативної належності позначаємо $\in_n$. Таким чином, $v_i \mapsto a_i \in_n d$ означає, що $d(v_i){\downarrow} = a_i$. Клас $ND(V, W) \setminus W$ називається класом *власне номінативних даних*. Щоб підкреслити ієрархічну структуру номінативних даних, будемо їх також називати ієрархічними номінативними даними. Однорівневі однозначні номінативні дані називатимемо *іменними множинами.*

Операції та композиції над номінативними даними. Ми визначимо тільки такі операції, що необхідні для даної статті.

Основні унарні операції над номінативними даними з ім'ям v як параметром: *іменування* $\Rightarrow v$, *розіменування* $v\Rightarrow$, *перевірка* $v!$. При застосовуванні до даного d операція іменування $\Rightarrow v$ видає як результат номінативне дане, що складається з єдиної компоненти з іменем v та значенням d; операція розіменування $v\Rightarrow$ видає як результат одне (довільно вибране) значення імені v, якщо принаймні одна компонента з іменем v знаходиться в d; операція перевірки $v!$ видає як результат порожнє номінативне дане $\varnothing$, якщо в d міститься принаймні одна компонента з іменем v, та видає як результат d, якщо в d немає компонент з іменем v.

Основні бінарні операції над номінативними даними*: накладення* ∇, *об'єднання* $\cup$, *різниця* $\setminus$. При застосовуванні до номінативних даних d і d' операція накладення видає як результат нове номінативне дане, що має як компоненти усі компоненти d' та ті компоненти d, імена яких не містяться в d'; операція об'єднання видає як результат нове дане, що має усі компоненти аргументів; операція різниці $\setminus$ видаляє з d ті компоненти, що належать d'.

Введемо також спеціальну операцію *недетермінованого вибору* choice: при застосовуванні до довільного номінативного даного d choice видає як результат або d, або $\varnothing$.

Функції множини $Fn = ND(V, W) \xrightarrow{m} \xrightarrow{m} ND(V, W)$ називаються номінативними функціями.

Основні композиції, визначені на Fn, — це бінарні композиції, описані такими співвідношеннями:

1) *множення* (функціональна композиція): $(f \circ g)(d) = g(f(d))$.

Зауважимо, що спочатку застосовується f, потім застосовується g;

2) і*терація*. Рекурсивне визначення композиції ітерації таке:

$$(f{*}g)(d) = \begin{cases} d, & \text{якщо } f(d) = \varnothing, \\ (f * g)(g(d)), & \text{якщо } f(d) \neq \varnothing; \end{cases}$$

3) *накладення*: $(f \nabla g)(d) = f(d) \nabla g(d)$;

4) *сума*: $(f \cup g)(d) = f(d) \cup g(d)$;

5) *різниця*: $(f - g)(d) = f(d) \setminus g(d)$,

де $f, g \in Fn$, $d \in ND(V, W)$.

В усіх випадках, коли деяке застосування функції до певних даних у правій частині співвідношення невизначене, результат лівої частини співвідношення також невизначений.

Абстрактна обчислюваність над номінативними даними. Будучи формальними моделями програм, КНС дозволяють визначити і досліджувати основні властивості програм. Одна з таких властивостей — *обчислюваність* програми.

Традиційні мови програмування звичайно називають універсальними. Проте більш повне дослідження змушує засумніватися в цьому, воно вказує на низку труднощів, що виникають при такому трактуванні поняття обчислюваності програм. Як правило, під ним розуміють обчислюваність n-арних функцій, визначених на цілих числах або на словах. Таку обчислюваність можна назвати Тьюринговою. Але програми також працюють з іншими структурами даних, тому для цих структур мови програмування, що розглядаються як універсальні, вже такими не будуть, оскільки не можуть подавати усі обчислювані функції, визначені на цих структурах [3]. Отже, уточнення поняття обчислюваності програми — нетривіальна проблема, яка вимагає спеціального дослідження. Потрібне нам поняття обчислюваності мусить бути також застосовне до структур даних різних рівнів абстракції. Така обчислюваність називається *абстрактною*. Спеціальний тип абстрактної обчислюваності, орієнтований на композиційно-номінативні системи, називається *натуральною* обчислюваністю [6, 7]. Ми не будемо тут деталізувати це питання, подамо тільки один результат, що описує повний клас обчислюваних функцій над номінативними даними (теорема 5 [6]). Цей результат буде використовуватися при вивченні логік над номінативними даними.

Нехай $V = \{v_0, \ldots, v_m\}$ — скінченна множина ($m > 0$), W — абстрактна множина. В цьому випадку $ND(V, W)$ назвемо множиною V-скінченних W-абстрактних номінативних даних.

Теорема 1. Повний клас натурально обчислюваних часткових багатозначних функцій на множині $ND(V, W)$ V-скінченних W-абстрактних номінативних даних збігається з класом функцій, отриманих замиканням множини функцій $\{\Rightarrow v_0, \ldots, \Rightarrow v_m, v_0\Rightarrow, \ldots, v_m\Rightarrow, v_0!, \ldots, v_m!, \text{choice}\}$ за допомогою множини композицій $\{\circ, *, \mathrm{U}, -\}$.

Специфікації програм. Декларативні методи специфікації програм розглядаються як більш абстрактні, ніж імперативні. Існують різні підходи до специфікації програм. Тут ми приймемо традиційну точку зору, згідно з якою для специфікації програм використовуються предикати. Труднощі полягають у тому, що ми розглядаємо часткові багатозначні функції. Щоб подавати такі функції, необхідно розвинути спеціальні часткові логіки. У цьому напрямку зроблені тільки перші кроки.

Введемо наступне визначення. Нехай $p : D^2 \rightarrow Bool$ — бінарний предикат, $f \in D' \xrightarrow{m} D'$ — часткова багатозначна функція (тут $D' \subseteq D$). Тоді предикат p *реляційно подає* функцію f, якщо графік функції f збігається з множиною $\{(a,r) \mid p(a,r)=T, a,r \in D'\}$.

Композиційно-номінативні предикатні системи. КНС можуть також використовуватися для визначення класів предикатів. Відповідні КНС називаються *предикатними* (ПКНС) [8]. Для випадку ПКНС композиційні системи мають вигляд <D, Pr, C>, де D — множина даних, $Pr \subseteq D \xrightarrow{m} Bool$ — деяка множина (часткових багатозначних) предикатів, C — множина композицій на Pr.

Ми розглядаємо ПКНС на трьох рівнях, індукованих відповідними рівнями абстракції даних: ПКНС над абстрактними даними, нескінченними іменними множинами та ієрархічними номінативними даними. ПКНС є семантичною основою для спеціальних логік предикатів (композиційних логік). Логіки абстрактного рівня називаються *пропозиційними*. Інфінітарні пропозиційні логіки для різних класів часткових і недетермінованих предикатів були введені в [8]. У даній статті ми пропонуємо дві логіки предикатів: над іменними множинами та над номінативними даними.

2. Композиційна логіка над іменними множинами

Ця логіка буде аналогом класичної логіки предикатів, тому ми назвемо її неокласичною композиційною логікою предикатів (НКЛП), або неокласичною логікою.

Щоб мотивувати наші визначення, мусимо спочатку проаналізувати основні поняття класичної логіки [9], до яких можна віднести:

1) *мову* L_{CL}, що описує множину формул Fr_{CL} над множиною Vr предметних змінних та n-арною предикатною сигнатурою $\sigma_{CL}=(Ps, ar)$, де Ps — множина предикатних символів, $ar: Ps \to \to Nat$ — відображення арності;

2) відношення *вивідності* (довідності) $\vdash_{CL} \subseteq 2^{Fr_{CL}} \times Fr_{CL}$ формули з множини аксіом;

3) поняття *моделі* (інтерпретації) Int_{CL}, базоване на понятті алгебраїчної системи (АС) заданої сигнатури $A(\sigma_{CL})=(A, \mu_{CL}^{Ps})$, де відображення $\mu_{CL}^{Ps}: Ps \to \bigcup_{n \in Nat}(A^n \to Bool)$ узгоджується з відображенням арності ar;

4) відношення *істинності* $\models_{CL} \subseteq \subseteq Int_{CL} \times Fr_{CL}$ формули на моделі.

Проведений в [10] аналіз класичної логіки дозволяє зробити наступні висновки:

– до побудови класичної логіки застосовується синтактико-семантичний підхід;

– трактування пропозиційних зв'язок і атомарних формул орієнтоване на використання n-арних предикатів;

– семантика кванторів та формул задається за допомогою інтерпретацій.

З погляду композиційно-номінативного підходу семантичні аспекти важливіші за синтаксичні, тому синтактико-семантичний стиль побудови логіки повинен бути змінений на семантико-синтаксичний. Це означає, що ми спочатку фіксуємо рівень абстракції розгляду предметних областей; потім будуємо відповідні обраному рівню абстракції композиційні системи (алгебри), які задають семантичні аспекти, після цього визначаємо мову і відношення істинності, індуковане композиційними системами; нарешті, задаємо відношення виведення, яке узгоджується з властивостями композиційних систем, та відношення істинності.

Основним семантичним поняттям логіки є поняття предикату. Традиційно досліджувані предикати трактуються як тотальні n-арні предикати типу $A^n \to Bool$. Але якщо ми спробуємо подати значення формул як тотальні n-арні предикати, тоді зіштовхнемося з багатьма труднощами. Зокрема, таке подання не є композиційним за Фреге [8]. Зазначені факти дають підставу замість тотальних n-арних предикатів розглядати часткові предикати над іменними даними. Такі часткові предикати, визначені на множині ${}^{Vr}A = Vr \to A$, називаються *Vr-квазіарними*. Зауважимо, що фактично цей клас предикатів індукований семантикою Тарського класичної логіки.

Розглянемо наступний приклад. Нехай $Vr = \{x, y, z, ...\}$. Тотальний предикат на множині цілих чисел, описаний формулою $x > y$, може розглядатися як Vr-арний предикат, визначений на нескінченній іменній множині вигляду $[x \mapsto a_1, y \mapsto a_2, z \mapsto a_3, \ldots]$, де $a_1, a_2, a_3, \ldots$ — цілі числа. Зрозуміло, що для цього предикату тільки змінні x та y є істотними, інші змінні неістотні. Це означає, що ми можемо також поставити у відповідність формулі $x > y$ частковий Vr-квазіарний предикат замість тотального предикату. Цей частковий предикат буде визначений на всіх нескінченних та скінченних іменних множинах, які мають компоненти з іменами x та y, наприклад, на іменній множині $[x \mapsto a_2,\ y \mapsto a_1,\ z \mapsto a_4]$. На іменних множинах, що не мають компоненти з іменем x або y, цей предикат невизначений. Розглянутий предикат має наступні дві властивості. По-перше, якщо він визначений на деякій іменній множині d (в цьому випадку d містить компоненти з іменами x та y), тоді він буде також визначений на будь-якій іншій іменній множині d', яка є розширенням d. По-друге, цей предикат завжди буде визначеним на будь-якій максимальній іменній множині, тобто на іменній множині, в якій кожна змінна має значення.

Формалізація цих двох властивостей веде до наступних визначень [3, 11].

Предикат $p \in {}^{Vr}A \to Bool$ назвемо *еквітонним*, якщо $(p(d)\downarrow \& (d \subseteq d')) \Rightarrow$ $\Rightarrow (p(d')\downarrow = p(d))$ для будь-яких $d, d' \in {}^{Vr}A$.

Предикат $p \in {}^{Vr}A \to Bool$ назвемо *повнототальним*, якщо $p(d)\downarrow$ для будь-яких $d \in A^{Vr}$.

Еквітонні повнототальні предикати називаються *Vr-повними* предикатами.

Неважко показати, що формули класичної логіки мають як значення Vr-повні предикати. Це твердження є безпосереднім наслідком відомої леми збігу (coincidence lemma). Зазначений факт дуже важливий для побудови неокласичної композиційної логіки, яка базується на класі $Pc(Vr, A)$ Vr-повних предикатів. Для таких класів часткових предикатів пропозиційні зв'язки і квантори доцільно трактувати як композиції.

Композиції предикатів. Спочатку визначимо композиції диз'юнкції $\vee$ та заперечення $\neg$ на класі $Pr = D \to Bool$ наступними співвідношеннями (тут $p, q \in Pr$, $d \in D$):

$$(p \vee q)(d) = \begin{cases} T, \text{ якщо } p(d)\downarrow = T \text{ або } q(d)\downarrow = T, \\ F, \text{ якщо } p(d)\downarrow = F \text{ та } q(d)\downarrow = F, \\ \text{невизначене в усіх інших} \\ \text{випадках;} \end{cases}$$

$$(\neg p)(d) = \begin{cases} T, \text{ якщо } p(d)\downarrow = F, \\ F, \text{ якщо } p(d)\downarrow = T, \\ \text{невизначене, якщо } p(d)\uparrow. \end{cases}$$

Ці композиції є аналогами сильних зв'язок Кліні. Вони є композиціями абстрактного рівня.

На наступному рівні визначаємо D як ${}^{Vr}A$. На цьому рівні можна визначити нову параметричну композицію *реномінації* $\boldsymbol{R}^{v_1,\ldots,v_n}_{x_1,\ldots,x_n}$ та екзистенційний квантор (квантор існування) $\exists \boldsymbol{x}$ (тут $p, q \in Pr$, $d \in {}^{Vr}A$, $v_1, \ldots, v_n, x_1, \ldots, x_n \in Vr$, причому $v_1, \ldots, v_n$ — попарно відмінні імена):

$$\boldsymbol{R}^{v_1,\ldots,v_n}_{x_1,\ldots,x_n}(p)(d) = p([v \mapsto a \mid v \mapsto a \in_n d,\ v \notin \{v_1, \ldots, v_n\}] \nabla [v_i \mapsto a_i \mid x_i \mapsto a_i \in_n d,\ i \in \{1, \ldots, n\}]);$$

$$\exists \boldsymbol{x}(p)(d) = \begin{cases} T, \text{ якщо існує } b \in A : p(d \nabla x \mapsto \\ \mapsto b)\downarrow = T, \\ F, \text{ якщо } p(d \nabla x \mapsto a)\downarrow = F \\ \text{ для всіх } a \in A, \\ \text{невизначене в усіх інших} \\ \text{ випадках.} \end{cases}$$

Похідні композиції $\boldsymbol{\&}$, $\to$, $\leftrightarrow$, $\forall \boldsymbol{x}$ визначаються традиційним способом.

Ввівши позначення вигляду $\bar{y}$ для y_1, ..., y_n, замість $\boldsymbol{R}^{v_1,...,v_n}_{x_1,...,x_n}$ також писатимемо $\boldsymbol{R}^{\bar{v}}_{\bar{x}}$.

Нехай клас предикатів $Pr \subseteq {}^{Vr}A \to \to Bool$ замкнений відносно введених композицій. Композиційна система $<{}^{Vr}A$, Pr, $\{\vee, \neg, \boldsymbol{R}^{\bar{v}}_{\bar{x}}, \exists \boldsymbol{x}\}>$ називається базовою композиційною системою першого порядку і далі буде подаватися як алгебра $APFO(Vr, A, Pr) = <Pr, \vee, \neg, \boldsymbol{R}^{\bar{v}}_{\bar{x}}, \exists \boldsymbol{x}>$. Назвемо її *базовою предикатною алгеброю першого порядку*. Зазначені алгебри формують семантичну основу композиційної логіки першого порядку.

Неважко довести наступне твердження [11].

Теорема 2. Клас Vr-повних предикатів $Pc(Vr, A)$ замкнений відносно композицій $\vee$, $\neg$, $\boldsymbol{R}^{\bar{v}}_{\bar{x}}$, $\exists \boldsymbol{x}$.

Таким чином, $APFO(Vr, A, Pc(Vr, A))$ — предикатна алгебра першого порядку. Для цієї алгебри зберігаються основні властивості традиційних пропозиційних зв'язок і кванторів. Тому обмежимося розглядом дистрибутивної властивості реномінації:

$\boldsymbol{R}^{\bar{v}}_{\bar{x}}(\neg p) = \neg \boldsymbol{R}^{\bar{v}}_{\bar{x}}(p)$ — дистрибутивність реномінації щодо заперечення;

$\boldsymbol{R}^{\bar{v}}_{\bar{x}}(p \vee q) = \boldsymbol{R}^{\bar{v}}_{\bar{x}}(p) \vee \boldsymbol{R}^{\bar{v}}_{\bar{x}}(q)$ — дистрибутивність реномінації щодо диз'юнкції;

$\boldsymbol{R}^{\bar{v}}_{\bar{u}}(\exists \boldsymbol{x} p) = \exists \boldsymbol{x} \boldsymbol{R}^{\bar{v}}_{\bar{u}}(p)$ за умови $x \notin \{\bar{\boldsymbol{v}}, \bar{\boldsymbol{u}}\}$ — обмежена дистрибутивність реномінації щодо кванторів.

Обмежена дистрибутивносłь не дає змоги просунути реномінацію вглиб формули і перетворити формулу до класичноподібного вигляду. Щоб це стало можливим за умови $x \in \{v_1, ..., v_n, u_1, ..., u_n\}$, необхідно замінити кванторну змінну x на таку змінну z, яка не входить до $\{v_1, ..., v_n, u_1, ..., u_n\}$. Тоді вираз $\exists xp$ можна замінити виразом $\exists z R^{x}_{z} p$. Проте при обчисленні p така заміна змінює значення z, яке може бути істотним для p. Отже, щоб заміна була коректною, треба вимагати неістотності z для p. Загальне визначення неістотності таке.

Змінна $z \in Vr$ *неістотна* для предикату $p \in {}^{Vr}A \to Bool$, якщо для будь-яких $d, d' \in {}^{Vr}A$, які відрізняються тільки значеннями змінної z, із $p(d)\downarrow$ та $p(d')\downarrow$ випливає $p(d) = p(d')$.

Наступне питання: як можна одержати неістотні змінні? Видається розумним постулювати існування таких змінних для базових предикатів і потім виводити цю властивість для предикатів, які є значеннями структурованих формул. Змінні, неістотні для базових предикатів, називаються *синтетично* неістотними. Змінні, неістотність яких виводиться, називаються *аналітично* неістотними. Таким чином, постулюючи існування нескінченної множини синтетично неістотних змінних, ми можемо просувати реномінацію вглиб формули до предикатних символів. Отримані формули природно назвати *псевдокласичними*.

Основні визначення і властивості неокласичної композиційної логіки. Неокласична композиційна логіка предикатів — спеціальна композиційна логіка першого порядку. Основні поняття НКЛП будемо звичайно позначати, використовуючи CPL як нижній індекс.

Композиційно-номінативні предикатні системи. Композиційні логіки базуються на ПКНС (предикатних алгебрах) першого порядку, що індукують семантичні та синтаксичні властивості таких логік.

Мова. Алфавіт мови НКЛП складається з множини Vr предметних змінних, множини Ps предикатних символів (символів базових предикатів) та символів базових композицій $\vee$, $\neg$, $R^{\bar{v}}_{\bar{x}}$, $\exists x$. Для кожного символу Ps множина синтетично неістотних змінних визначається за допомогою тотального відображення $ne : Ps \to 2^{Vr}$. Пару σ_{CPL} = $= (Ps, ne)$ назвемо предикатною сигнатурою. Вимагається, щоб множина $\bigcap_{p \in Ps} ne(p)$, яку назвемо множиною тотально неістотних змінних, була нескінченною. Таке визначення сигнатури аналогічне визначенню сигнатури класичної логіки предикатів $\sigma_{CL}=(Ps, ar)$. Проте на противагу випадку класичної логіки для НКЛП сигнатура не впливає на множину формул, а обмежує тільки клас інтерпретацій. Тому множина формул НКЛП є множиною термів предикатної алгебри першого порядку. Позначимо її Fr_{CPL}.

Звичайним способом вводимо символи похідних композицій $\rightarrow$, &, $\leftrightarrow$, $\forall x$.

Інтерпретації задають значення предикатних символів, узгоджене з їх сигнатурою. Це означає: $\mu_{CPL}{}^{Ps}: Ps \rightarrow \rightarrow PC(Vr, A)$ є відображенням денотації (інтерпретації) предикатних символів таким, що для будь-якого $p \in Ps$ змінні множини $ne(p)$ є неістотними для предикату $\mu_{CPL}{}^{Ps}(p)$. Пара $E(\sigma_{CPL})=(A, \mu_{CPL}{}^{Ps})$ називається алгебраїчною системою Vr-повних предикатів, або неокласичною АС. Така алгебраїчна система — аналог класичної АС n-арних предикатів $A(\sigma_{CL})=(A, \mu_{CL}{}^{Ps})$. Якщо сигнатури зрозумілі з контексту, їх опускаємо. Відображення $\mu_{CPL}{}^{Ps}$ природним чином продовжується до відображення $\mu_{CPL}: Fr_{CPL} \rightarrow \rightarrow PC(Vr, A)$.

Відношення істинності $|=_{CPL}$. Формула $\Phi \in Fr_{CPL}$ *частково істинна* (неспростовна) в E, що позначаємо $E\,|=_{CPL}\Phi$, якщо для кожного $d \in {}^{Vr}A$ маємо $\mu_{CPL}(\Phi)(d)\downarrow \Rightarrow \mu_{CPL}(\Phi)(d) = T$. Формула Φ *всюди істинна*, що позначаємо $|=_{CPL}\Phi$, якщо $E\,|=_{CPL}\Phi$ для кожної неокласичної АС E.

Відношення вивідності $|-_{CPL}$. Відношення вивідності для НКЛП можна задати формально-аксіоматичними системами Гільбертівського типу [12]. Враховуючи властивості предикатних алгебр першого порядку, визначимо наступну множину логічних аксіом:

AxPr) $\neg\Phi \vee \Phi$ — пропозиційні аксіоми;

AxR∃x) $R_{x_1,\ldots,x_n}^{v_1,\ldots,v_n}(\Phi) \rightarrow \exists v_1 \ldots \exists v_n \Phi$ — аксіоми підстановки;

AxR¬) $R_{\bar{x}}^{\bar{v}}(\neg\Phi) \leftrightarrow \neg(R_{\bar{x}}^{\bar{v}}(\Phi))$ — R¬-дистрибутивність;

AxR∨) $(R_{\bar{x}}^{\bar{v}}(\Phi) \vee R_{\bar{x}}^{\bar{v}}(\Psi)) \leftrightarrow R_{\bar{x}}^{\bar{v}}(\Phi \vee \vee \Psi)$ — R∨-дистрибутивність;

AxRR) $R_{\bar{x}}^{\bar{v}}(R_{\bar{z}}^{\bar{u}}(\Phi)) \leftrightarrow R_{\bar{x}}^{\bar{v}} \circ_{\bar{z}}^{\bar{u}}(\Phi)$ — згортка реномінацій [10, 11];

AxR∃) $R_{\bar{u}}^{\bar{v}}(\exists x\Phi) \leftrightarrow \exists x R_{\bar{u}}^{\bar{v}}(\Phi)$ при $x \notin \notin \{\bar{v}, \bar{u}\}$ — обмежена R∃-дистрибутивність;

AxRT) $R_z^z(\Phi) \leftrightarrow \Phi$ — згортка пар однакових імен в реномінаціях;

Ax∃∃) $\exists x \exists y \Phi \rightarrow \exists y \exists x \Phi$ — комутативність ∃-префіксів;

AxN∃) $\exists x\Phi \rightarrow \forall x \exists x \Phi$ — аналітична неістотність квантифікованих імен;

AxNR) $\exists x R_z^x(\Phi) \rightarrow R_z^x(\Phi)$ при $x \neq z$ — аналітична неістотність верхніх імен в реномінаціях;

AxNP) $p \rightarrow \forall xp$, де $p \in Ps$, — синтетична неістотність імен для базових предикатів

(тут $\Phi, \Psi \in Fr_{CPL}$).

Множина правил виведення складається з таких правил:

П1) $\Phi \,|\!-\, \Psi \vee \Phi$ — правило розширення;

П2) $\Phi \vee \Phi \,|\!-\, \Phi$ — правило скорочення;

П3) $\Phi \vee (\Psi \vee \Xi) \,|\!-\, (\Phi \vee \Psi) \vee \Xi$ — правило асоціативності;

П4) $\Phi \vee \Psi,\ \neg\Phi \vee \Xi \,|\!-\, \Psi \vee \Xi$ — правило перетину;

П5) $\Phi \rightarrow \Psi,\ \Psi \rightarrow \forall x \Psi \,|\!-\, \exists x \Phi \rightarrow \Psi$ — правило $\exists_\forall$;

П6) $\Phi \rightarrow \forall y \Phi \,|\!-\, R_{z,\bar{x}}^{y,\bar{v}}(\Phi) \leftrightarrow R_{\bar{x}}^{\bar{v}}(\Phi)$ — правило згортки по неістотному імені

(тут $\Phi, \Psi, \Xi \in Fr_{CPL}$).

Розглянемо основні властивості НКЛП: коректність та повноту. Коректність можна довести, показуючи істинність аксіом в предикатних алгебрах першого порядку та збереження істинності правилами виведення. Повноту НКЛП можна довести безпосередньо. Таке доведення проводиться на основі секвенційних числень неокласичних логік, запропонованих та досліджених в [13, 14]. Вкажемо тут в певному розумінні більш сильний результат щодо взаємної трансляції (моделювання) класичної та неокласичної логік.

Універсальне моделювання класичної логіки в неокласичній. Зафіксуємо деяку мову класичної логіки першого порядку сигнатури σ_{CL}. Спочатку промоделюємо сигнатуру σ_{CL} у НКЛП. Виділимо в Vr зліченну послідовність різних змінних $u_1, ..., u_n, ...,$ що будуть грати роль стандартних імен 1, 2, ..., і збудуємо сигнатуру $\sigma_{CPL} = (Ps, ne)$, взявши $ne(p) = Vr \setminus \{u_1,..., u_n\}$, де $n = ar(p)$, для довільного $p \in Ps$. Тепер промоделюємо класичні АС в НКЛП. Взявши довільну класичну АС $A(\sigma_{CL}) = (A, \mu_{CL}{}^{Ps})$, збудуємо відповідну неокласичну АС $E(\sigma_{CPL}) = (A, \mu_{CPL}{}^{Ps})$, поклавши $\mu_{CPL}(p)(d)\downarrow = b \Leftrightarrow \mu_{CL}(p)(a_1, ..., a_n)\downarrow = b \;\&\; [u_1 \mapsto a_1, \dots, u_n \mapsto a_n] \subseteq d$ для довільних $p \in Ps$, $d \in {}^{Vr}A$, $b \in Bool$. Побудовану АС позначимо $nc(A)$. Нарешті, моделюємо формули класичної логіки в композиційній. Для цього збудуємо відображення $nc : Fr_{CL}(\sigma_{CL}) \to Fr_{CPL}(\sigma_{CPL})$, замінюючи в $\varphi \in Fr_{CL}$ всі атомарні формули вигляду $p(x_1, ..., x_m)$ формулами $R^{u_1,...,u_m}_{x_1,...,x_m}(p)$. Така трансляція зберігає істинність формул та всюди істинність класичних формул в обох логіках.

Всі аксіоми класичної логіки виводяться в НКЛП, правила виведення класичної логіки моделюються в НКЛП. Звідси отримуємо моделювання виведень формул класичної логіки в НКЛП. Отже, можемо говорити про ізоморфне вкладення класичної логіки в НКЛП.

Теорема 3. Нехай $A(\sigma_{CL})$ — довільна класична АС та $\varphi \in Fr_{CL}$. Тоді

1) $A(\sigma_{CL}) \models_{CL} \varphi \Rightarrow nc(A) \models_{CPL} nc(\varphi)$;

2) $\models_{CL} \varphi \Rightarrow \models_{CPL} nc(\varphi)$;

3) $\vdash_{CL} \varphi \Rightarrow \vdash_{CPL} nc(\varphi)$.

Обмежене моделювання неокласичної логіки в класичній. Універсального моделювання класу формул НКЛП в класі формул класичної логіки не існує. Причина полягає в тому, що множини істотних змінних для формул НКЛП можуть бути нескінченними. Проте можемо зробити обмежене моделювання, тому що будь-яку формулу НКЛП можна звести до еквівалентної їй псевдокласичної форми, використовуючи додаткові змінні.

Теорема 4. Нехай $\Phi \in Fr_{CPL}$. Тоді існує формула $\Phi^{(c)}$ в псевдокласичній формі така, що $\models_{CPL} \Phi \Leftrightarrow \models_{CPL} \Phi^{(c)}$ та $\vdash_{CPL} \Phi \Leftrightarrow \vdash_{CPL} \Phi^{(c)}$.

Всюди істинність $\Phi^{(c)}$ можна перевірити на її класичному аналогу $\varphi^{(c)}$. Крім того, можна моделювати виведення таких формул в обох логіках [12]. Отже, справджується

Теорема 5. Нехай $\Phi^{(c)} \in Fr_{CPL}$ та $\varphi^{(c)} \in Fr_{CL}$ — її класичний аналог. Тоді $\models_{CPL} \Phi^{(c)} \Leftrightarrow \models_{CL} \varphi^{(c)}$ та $\vdash_{CPL} \Phi^{(c)} \Leftrightarrow \vdash_{CL} \varphi^{(c)}$.

З теорем 3—5 і коректності та повноти класичної логіки випливає теорема коректності та повноти неокласичної композиційної логіки:

Теорема 6. Нехай $\Phi \in Fr_{CPL}$. Тоді $\models_{CPL} \Phi \Leftrightarrow \vdash_{CPL} \Phi$.

Підсумуємо відмінність та подібність класичної логіки і неокласичної композиційної логіки. Основна відмінність полягає в тому, що НКЛП будується в семантико-синтаксичному стилі на основі поняття композиційно-номінативної системи. Це веде до простішої семантики, що задається предикатною алгеброю першого порядку, та дозволяє безпосередньо використовувати потужний апарат алгебри для дослідження НКЛП. Пропозиційні зв'язки та квантори набувають явного семантичного статусу композицій. Клас моделей НКЛП істотно багатший за клас моделей класичної логіки. В той же час класична і композиційна логіки дуже близькі. Ми можемо говорити про ізоморфне моделювання класичної логіки в НКЛП та обмежене моделювання НКЛП в класичній логіці.

3. Композиційна логіка предикатів над номінативними даними

Предикатна алгебра першого порядку над номінативними даними. Неокласична композиційна логіка предикатів побудована згідно припущення, що дані були іменними множинами, тобто однорівневими номінативними даними класу ${}^{Vr}A$. На наступному рівні абстракції елементи A можна розглядати як ієрархічні номінативні дані над V і W. Ми також включаємо до A множину V, інакше прийдеться будувати двосортну аксіоматичну теорію номінативних даних. Проте ми воліємо розглядати односортну теорію, тому A визначимо як $V \cup ND(V, W)$. Припускаємо також, що V містить принаймні $m+1$ $(m>0)$ попарно різних імен $v_1, ..., v_m$. Побудовану композиційну логіку назвемо логікою номінативних даних (ЛНД).

Введемо спеціальні параметричні предикати типу $Pc(Vr, V \cup ND(V, W))$:

– предикат номінативної належності $x \mapsto y \in_n z$;

– предикат рівності $x = y$;

– характеристичний предикат $\widetilde{W}(x)$ для множини W;

– характеристичний предикат $\widetilde{V}(x)$ для множини V;

– характеристичний предикат $\widetilde{v}_i(x)$ для елемента $v_i \in V$, де $i \in \{0, ..., m\}$ (тут $x, y, z \in Vr$).

Такі позначення роблять формули будованої логіки дуже близькими до традиційної форми. Крім того, обмежимо у формулах явне використання реномінації.

Таким чином, розглядаємо алгебру $APND(Vr, V, W) = < Pc(Vr, V \cup ND(V, W))$, $\vee$, $\neg$, $\boldsymbol{R}_{\boldsymbol{x}}^{\bar{\boldsymbol{v}}}$, $\exists \boldsymbol{x}$, $x \mapsto y \in_n z$, $x = y$, $\widetilde{W}(x)$, $\widetilde{W}(x)$, $\widetilde{v}_0(x)$, …, $\widetilde{v}_m(x)>$.

Властивості відношення номінативної належності. Зрозуміло, що відношення номінативної належності подібне до відношення належності теорії множин. Тому при побудові аксіоматичної теорії номінативних даних можна керуватись ідеями, що застосовуються в теорії множин. Тут ми будемо орієнтуватись на теорію допустимих множин [15, 16], яка має деякі специфічні особливості.

По-перше, ця теорія використовує праелементи (базові елементи) для побудови структур даних. Відомо, що аксіоматична теорія множин ZF Цермело – Френкеля дозволяє робити праелементи непотрібними. ZF є елегантним та могутнім засобом організації математики, але вона занадто потужна для теорії програмування. Теорія допустимих множин більш слабка порівнянно з ZF стосовно принципів існування множин, вона, зокрема, не допускає застосування аксіоми побудови множини всіх підмножин. У той же час ця теорія дозволяє праелементи, які інтерпретуються як базові дані. Це дає змогу природним чином будувати складні структури даних, що добре узгоджується з практикою програмування.

По-друге, ця теорія не настільки потужна, як ZF, але вона достатньо сильна, щоб породжувати функції, які використовуються в теорії баз даних і програмуванні.

Враховуючи зазначені моменти, будуємо ЛНД у стилі теорії допустимих множин.

Аксіоматична теорія номінативних даних. Деякі зауваження стосовно позначень. По-перше, використовуємо традиційний запис $\varphi(x)$ для позначення формули $R_x^z(\varphi)$, утвореної з φ перейменуванням z на x, якщо ім'я z зрозуміле з контексту і не вимагає пояснень. По-друге, припускаємо, що змінні a, b, c приймають значення в класі власне номінативних даних, тобто справджуються формули $\neg\widetilde{W}(a) \& \neg\widetilde{V}(a)$, $\neg\widetilde{W}(b)$ & & $\neg\widetilde{V}(b)$, $\neg\widetilde{W}(c) \& \neg\widetilde{V}(c)$. Це дає змогу легко вказувати належність до номінативних даних чи до базових даних. Наприклад, формула $\exists a \exists x \exists y (x \mapsto y \in_n a)$ стверджує, що існують непорожні номінативні дані.

Введемо поняття обмежених кванторів. Записи

$$\forall x \mapsto y \in_n a\ \varphi \text{ та } \exists x \mapsto y \in_n a\ \varphi$$

розглядаємо як скорочені позначення формул $\forall x \forall y (x \mapsto y \in_n a \rightarrow \varphi)$ та $\exists x \exists y (x \mapsto y \in_n a \,\&\, \varphi)$ відповідно.

Строге включення визначаємо формулою $a \subset b \leftrightarrow (a \neq b) \,\&\, \forall x \forall y (x \mapsto y \in_n a \rightarrow x \mapsto y \in_n b)$.

Поняття Δ_0-формули та Σ-формули вводимо традиційним [15, 16] способом (див. також [1, 2]). Клас Σ-формул буде використовуватися при вивченні обчислюваності в ЛНД.

Спеціальні аксіоми ЛНД розділені на три групи. Перша група описує властивості рівності:

AxRf) $x = x$ — рефлексивність рівності;

AxEq) $x_1 = y_1 \,\&\, \ldots \,\&\, x_n = y_n \to R^{v_1,\ldots,v_n}_{x_1,\ldots,x_n}\varphi \leftrightarrow R^{v_1,\ldots,v_n}_{y_1,\ldots,y_n}\varphi$ — заміна рівних.

Друга група аксіом описує властивості власне номінативних даних:

AxEx) $\forall x \forall y (x \mapsto y \in_n a \leftrightarrow x \mapsto y \in_n b) \to a = b$ — екстенсійність;

AxFn) $(\forall a (\forall x \mapsto y \in_n a\, \varphi(x) \,\&\, \varphi(y) \to \varphi(a))) \to \forall a\, \varphi(a)$ — фундованість (індукція за $\in_n$);

AxIn) $(\forall a (\forall b \subset a\, \varphi(b) \to \varphi(a))) \to \forall a\, \varphi(a)$ — індукція за включенням;

AxSp) $\exists b \forall x \forall y (x \mapsto y \in_n b \leftrightarrow x \mapsto y \in_n a \,\&\, \varphi(a))$ — виділення;

AxNm) $\exists c (x \mapsto y \in_n c)$ — іменування;

AxUn) $\exists c (a \subseteq c \,\&\, b \subseteq c)$ — об'єднання;

AxNT) $\exists a \exists x \exists y (x \mapsto y \in_n a)$ — нетривіальність власне номінативних даних;

ANV) $(x \mapsto y \in_n u) \to \widetilde{V}(x)$ — тільки елементи V можуть використовуватися як імена.

Третя група аксіом описує властивості множин базових імен і даних:

AxAt) $\widetilde{V}(z) \vee \widetilde{W}(z) \to \forall x \forall y (x \mapsto y \notin_n z)$ — атомарність базових імен і даних;

AxCN) $(\widetilde{v}_i(x) \,\&\, \widetilde{v}_i(y) \to x = y$, де $i \in \{0, \ldots, m\}$ — єдиність опису імені характеристичним предикатом;

AxDN) $\exists u_0 \ldots \exists u_{m-1} \exists u_m (\widetilde{v}_0(u_0) \,\&\, \ldots \,\&\, \widetilde{v}_{m-1}(u_{m-1}) \,\&\, \widetilde{v}_m(u_m) \,\&\, u_0 \neq u_1 \,\&\, \ldots \,\&\, u_0 \neq u_m \,\&\, \ldots \,\&\, u_{m-1} \neq u_m)$ — існування $m+1$ різних імен.

Теорема 7. Аксіоматична теорія номінативних даних несуперечлива.

Для доведення теореми треба показати, що клас предикатів над $A = V \cup ND(V, W)$ — модель для цієї теорії, звідки теорія несуперечлива.

ЛНД досить потужна, вона дає змогу подавати всі натурально обчислювані функції над номінативними даними.

Подання обчислюваних функцій. Покажемо, що кожна натурально обчислювана функція над номінативними даними може бути реляційно подана предикатом, описаним деякою Σ-формулою. Такі предикати назвемо Σ-предикатами. Для доведення цього твердження будуємо подання всіх основних функцій, зазначених в теоремі 1, та всіх функцій, отриманих за допомогою композицій, наведених в цій теоремі. Строгий запис всіх предикатів технічно досить складний, тому допускаємо деякі синтаксичні вільності, а саме, "подаємо" функції записами вигляду $f(x) = y \equiv \varphi(x, y)$. Зауважимо, що тут формула $f(x) = y$ розглядається як скорочення формули $\varphi(x, y)$. Введення констант обґрунтовується формулами вигляду $\exists! x\, k(x)$. Зокрема, формула $\exists! a\, \forall x \mapsto y \in_n a\, (x \neq x)$, яка засвідчує існування єдиного елемента, що є власне номінативним даним і не містить жодних компонент, описує порожнє номінативне дане $\varnothing$. Тоді $z = \varnothing$ є скороченням формули $\forall x \mapsto y \in_n z\, (x \neq x)$. Нові предикатні символи вводяться записами вигляду $Q(x) \equiv \psi(x)$. Можна використовувати попередньо подані символи. Звичайно, коректність таких введень мусить бути доведена.

Існування імен обґрунтовується виразами вигляду $\exists! x\, \widetilde{v}_i(x)$ для імені v_i, $i \in \{0, \ldots, m\}$.

Подаємо параметричні іменування, розіменування і функції перевірки такими виразами:

$\Rightarrow v_i(x) = y \equiv (\exists z \mapsto z' \in_n y\ (z = v_i \,\&\, z' = x)) \,\&\, (\forall z \mapsto z' \in_n y\, (z = v_i \,\&\, z' = x))$;

$v_i \Rightarrow (x) = y \equiv (\exists z \mapsto z' \in_n x\, (z = v_i \,\&\, z' = y)$;

$v_i!(x) = y \equiv (\exists z \mapsto z' \in_n x\ (z = v_i \rightarrow y = \varnothing)\ \&$

$\&\ (\forall z \mapsto z' \in_n x\ (z \neq v_i) \rightarrow y = x).$

Операція вибору: $\mathrm{choice}(x) = y \equiv (y = \varnothing \vee y = x)$.

Покажемо, як можна подавати функції, побудовані за допомогою композицій.

Нехай функції f та g подані формулами $P(x, y)$ та $Q(x, y)$, тоді для множення, сумування та різниці цих функцій можна використати наступні вирази:

$(f \circ g)(x) = y \equiv \exists z P(x, z)\ \&\ G(z, y);$

$(f \cup g)(x) = y \equiv \exists a \exists b (P(x, a)\ \&\ G(x, b)\ \&\ (\forall z \mapsto z' \in_n y\ (z \mapsto z' \in_n a \vee z \mapsto z' \in_n b))\ \&\ (\forall z \mapsto z' \in_n a\ (z \mapsto z' \in_n y))\ \&\ (\forall z \mapsto z' \in_n b\ (z \mapsto z' \in_n y)));$

$(f - g)(x) = y \equiv \exists a \exists b (P(x, a)\ \&\ G(x, b)\ \&\ (\forall z \mapsto z' \in_n y\ (z \mapsto z' \in_n a\ \& \vee z \mapsto z' \notin_n b))\ \&\ (\forall z \mapsto z' \in_n a\ (z \mapsto z' \notin_n b \rightarrow z \mapsto z' \in_n y))).$

Для ітерації ситуація складніша. Спочатку ми повинні подати множини, ординали, натуральні числа і послідовності номінативних даних, потім подати індексні операції над послідовностями та операцію наступника для натуральних чисел.

Множина $\{d_1, ..., d_n\}$ подається як номінативне дане $[v_0 \mapsto d_1, \ldots, v_0 \mapsto d_n]$. Послідовність $<d_1, ..., d_n>$ — як номінативне дане $[v_0 \mapsto [v_0 \mapsto 1, v_1 \mapsto d_1], \ldots, [v_0 \mapsto [v_0 \mapsto n, v_1 \mapsto d_n]]$.

Усі необхідні побудови можуть бути зроблені в стилі [15,16] з використанням тільки Σ-формул. Отже, справджується

Теорема 8. Клас натурально обчислюваних функцій над класом номінативних даних може бути поданий за допомогою Σ-предикатів ЛНД.

Таким чином, побудовані композиційні логіки номінативних даних досить потужні, вони дозволяють подати клас всіх натурально обчислюваних функцій, отже, такі логіки неповні згідно теореми Гьоделя про неповноту.

Теорема 9. Аксіоматична теорія номінативних даних неповна.

Висновки

У цій статті ми запропонували дві спеціальні логіки, орієнтовані на специфікації програм. Вони побудовані в семантико-синтаксичному стилі і є композиційними за Фреге. Часткові предикати над номінативними даними використовуються для подання денотатів формул. Перша логіка (неокласична композиційна логіка) була визначена як логіка еквітонних предикатів над нескінченними іменними множинами, вона зберігає основні властивості класичної логіки предикатів. Друга логіка (над скінченними номінативними даними) є конкретизацією композиційної логіки предикатів. На основі цієї логіки визначається клас багатозначних натурально (абстрактно) обчислюваних функцій над номінативними даними.

1. *Аксиоматические* системы спецификаций программ над номинативными данными / Н.С. Никитченко, Л.Л. Омельчук, С.С. Шкильняк, О.И. Янченко // Пробл. программирования. — 2000. — № 1–2. — С. 259–272.
2. *Нікітченко М.С., Шкільняк С.С., Омельчук Л.Л.* Програми над ідентифікованими даними та їх Σ-визначеність. — Київ, 1999. — Деп. в ДНТБ України 01.10.99, N 242–Ук99. — 82 с.
3. *Nikitchenko N.* A Composition Nominative Approach to Program Semantics: Technical Report IT–TR: 1998–020. — Denmark: Technical University of Denmark, 1998. — 103 p.
4. *Никитченко Н.С.* Композиционно–номинативный подход к уточнению понятия программы // Пробл. программирования. — 1999. — № 1. — С. 16–31.
5. *Редько В.Н.* Композиции программ и композиционное программирование // Программирование. — 1978. — № 5. — С. 3–24.
6. *Nikitchenko N.S.* Abstract Computability of Non-deterministic Programs over Various Data Structures // Perspectives of Systems Informatics: Proc. of A.Ershov Fourth Intern. Conf., Novosibirsk, 2001. — LNCS 2244 (2001). — P. 471–484.
7. *Басараб И.А., Никитченко Н.С., Редько В.Н.* Композиционные базы данных. — Киев: Либідь. — 1992. — 191 с.
8. *Никитченко Н.С.* Предикатные композиционно–номинативные системы // Пробл. программирования. — 1999. — № 2. — С. 3–19.
9. *Шенфилд Дж.* Математическая логика. — М.: Наука, 1975. — 527 с.
10. *Никитченко Н.С., Шкильняк С.С.* Неоклассические логики предикатов // Пробл.

программирования. — 2000. — № 3–4. — С. 3–17.

11. *Нікітченко М.С., Шкільняк С.С.* Алгебри еквітонних функцій та їх властивості // Вісн. Київ. Ун. Сер.: фіз.-мат. науки. — 1998. — Вип 2. — С. 222–232.
12. *Нікітченко М.С., Шкільняк С.С.* Чисті композиційно-номінативні числення // Там же. — 2000. — Вип 3. — С. 290–303.
13. *Нікітченко М.С., Шкільняк С.С.* Неокласичні логіки та секвенційні числення. — Деп. в ДНТБ України 22.07.2002, № 114-Ук2002. — 46 с.
14. *Шкільняк С.С.* Неокласичні секвенційні числення. — Вісн. Київ. Ун. Сер.: фіз.-мат. науки, 2002. — Вип 4. — С. 261–274.
15. *Barwise J.* Admissible sets and structures. — Berlin: Springer–Verlag, 1975. — 394 p.
16. *Ершов Ю.Л.* Определимость и вычислимость. — Новосибирск: Науч. кн., 1996. — 300 с.

Отримано 03.09.03

Про авторів

Нікітченко Микола Степанович
доктор фіз.-мат. наук, зав. кафедрою теорії програмування

Шкільняк Степан Степанович
канд. фіз.-мат. наук, доцент кафедри теорії програмування

Місце роботи авторів:
Київський національний університет ім. Тараса Шевченка

вул. Володимирська, 60, Київ, Україна
Тел. (044) 259 0519

E-mail: nikitchenko@unicyb.kiev.ua
sssh@unicyb.kiev.ua