

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ
ІМЕНІ АДМІРАЛА МАКАРОВА
МИКОЛАЇВСЬКА ОБЛАСНА ДЕРЖАВНА АДМІНІСТРАЦІЯ
ПІВДЕННИЙ НАУКОВИЙ ЦЕНТР НАН УКРАЇНИ
ГОЛОВНЕ УПРАВЛІННЯ ДЕРЖАВНОЇ СЛУЖБИ З НАДЗВИЧАЙНИХ СИТУАЦІЙ УКРАЇНИ
У МИКОЛАЇВСЬКІЙ ОБЛАСТІ
ДП «ДОСЛІДНО-ПРОЕКТНИЙ ЦЕНТР КОРАБЛЕБУДУВАННЯ»
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ОДЕСЬКА НАЦІОНАЛЬНА АКАДЕМІЯ»
ОДЕСЬКИЙ НАЦІОНАЛЬНИЙ МОРСЬКИЙ УНІВЕРСИТЕТ
ХАРБІНСЬКИЙ ІНЖЕНЕРНИЙ УНІВЕРСИТЕТ (КИТАЙ)
УНІВЕРСИТЕТ НАУКИ І ТЕХНОЛОГІЙ ЦЗЯНСУ (КИТАЙ)
ГДАНСЬКИЙ ПОЛІТЕХНІЧНИЙ УНІВЕРСИТЕТ (ПОЛЬЩА)
ЗАХІДНО-ПОМЕРАНСЬКИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ (ПОЛЬЩА)
КОШАЛІНСЬКИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ (ПОЛЬЩА)
БАТУМСЬКИЙ НАВЧАЛЬНО-НАВІГАЦІЙНИЙ УНІВЕРСИТЕТ (ГРУЗІЯ)
МОРСЬКЕ ІНЖЕНЕРНЕ БЮРО
АТ «ЗАВОД «ЕКВАТОР»
КОМПАНІЯ «АМІКО ГРУПП»
ДП «ДЕЛЬТА-ЛОЦМАН»
ТОВ "ЮСК СЕРВИС"
ТОВ «ЕВЕРІ»

ІННОВАЦІЇ В СУДНОБУДУВАННІ ТА ОКЕАНОТЕХНІЦІ

МАТЕРІАЛИ

X міжнародної науково-технічної конференції

Том 2.

26 – 28 вересня 2019 р.

*Національний університет кораблебудування
імені адмірала Макарова
просп. Героїв України, 9 м. Миколаїв*

УДК 001.895:629.5
И66

ОРГАНІЗАТОРИ

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ КОРАБЛЕБУДУВАННЯ
ІМЕНІ АДМІРАЛА МАКАРОВА
МИКОЛАЇВСЬКА ОБЛАСНА ДЕРЖАВНА АДМІНІСТРАЦІЯ
ПІВДЕННИЙ НАУКОВИЙ ЦЕНТР НАН УКРАЇНИ
ГОЛОВНЕ УПРАВЛІННЯ ДЕРЖАВНОЇ СЛУЖБИ З НАДЗВИЧАЙНИХ СИТУАЦІЙ УКРАЇНИ
У МИКОЛАЇВСЬКІЙ ОБЛАСТІ
ДП «ДОСЛІДНО-ПРОЕКТНИЙ ЦЕНТР КОРАБЛЕБУДУВАННЯ»
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ОДЕСЬКА НАЦІОНАЛЬНА АКАДЕМІЯ»
ОДЕСЬКИЙ НАЦІОНАЛЬНИЙ МОРСЬКИЙ УНІВЕРСИТЕТ
ХАРБІНСЬКИЙ ІНЖЕНЕРНИЙ УНІВЕРСИТЕТ (КИТАЙ)
УНІВЕРСИТЕТ НАУКИ І ТЕХНОЛОГІЙ ЦЗЯНСУ (КИТАЙ)
ГДАНСЬКИЙ ПОЛІТЕХНІЧНИЙ УНІВЕРСИТЕТ (ПОЛЬЩА)
ЗАХІДНО-ПОМЕРАНСЬКИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ (ПОЛЬЩА)
КОШАЛІНСЬКИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ (ПОЛЬЩА)
БАТУМСЬКИЙ НАВЧАЛЬНО-НАВІГАЦІЙНИЙ УНІВЕРСИТЕТ (ГРУЗІЯ)
МОРСЬКЕ ІНЖЕНЕРНЕ БЮРО
АТ «ЗАВОД «ЕКВАТОР»
КОМПАНІЯ «АМІКО ГРУПП»
ДП «ДЕЛЬТА-ЛОЦМАН»
ТОВ "ЮСК СЕРВИС"
ТОВ «ЕВЕРІ»

**Матеріали публікуються за оригіналами, наданими авторами.
Претензії до організаторів не приймаються.**

Відповідальний за випуск:
Блінцов Володимир Степанович

И66 **Інновації** в судобудуванні та океанотехніці : Матеріали X Міжнародної науково-технічної конференції. у 2 томах. Т. 2. — Миколаїв : НУК, 2019. — 448 с.

ISBN 978-966-321-370-5

У збірнику наведені матеріали X Міжнародної науково-технічної конференції «Інновації в судобудуванні та океанотехніці». Збірник становить інтерес для наукових працівників, викладачів, інженерів та студентів.

УДК 001.895:629.5

ISBN 978-966-321-370-5

© Національний університет кораблебудування
імені адмірала Макарова, 2019

Vacuumization high performance databases of commercial projects

Text of the annotation. The causes of bloating, vacuumization methods and restructuring high-performance databases are considered. Considered the advantages and disadvantages of various vacuumization methods.

Keywords: database; vacuumization; transaction

Дудченко Олег Миколайович, Михаліченко Павло Євгенович

Вакуумізація високопродуктивних баз даних комерційних проектів.

Текст анотація. Розглянуто причини здуття, способи вакуумізації та реструктуризації високопродуктивних баз даних. Розглянуто різноманітні способи вакуумізації.

Ключові слова: база даних; вакуумізація; транзакція.

УДК 004.023:1.159.956

ДОСЛІДЖЕННЯ РОЛІ ІНТУЇЦІЇ ТА ЕВРИСТИКИ В ПРОГРАМУВАННІ

Бузунова М. І., ст. гр. 5141м, Патлайчук О.В., к.ф.н., доц.

Національний університет кораблебудування імені адмірала Макарова,

Україна, м. Миколаїв,

buzunovamargarita@gmail.com, volnistik@gmail.com

Анотація. Основним предметом дослідження в статті є пограмістська евристика як сукупність методів вирішення нестандартних завдань в програмуванні, тобто таких завдань, для яких не існує відомих алгоритмів. Як специфічний механізм мислення, що породжує елементи здогади за основою евристики, береться інтуїція. Стаття ґрунтується на висновках, отриманих Ж. Адамаром, Н. Вінером, Р. Декартом, А. Пуанкаре, і Ж. Піаже.

Ключові слова: евристика, програмування, інтуїція, філософія.

Вступ Постановка проблеми та її актуальність. Актуальність дослідження полягає в необхідності філософського осягнення і обґрунтування нових відкриттів в програмуванні, де чималу роль завжди відігравала інтуїція. Очевидно, на зараз програмування зберігає багато рис з математики попередніх епох, але має і свої особливості. Вона полягає в надзвичайно високому рівні абстракцій і ідеалізацій науки, значну складність нових моделей та алгоритмів.

Стан наукової розробки проблеми. Поняття «інтуїція» розглядалась в працях філософів античної епохи: Платон, Аристотель. В епоху середньовіччя: Ф. Аквінський, У. Оккам. В епоху нового часу: Р. Декарт, Б. Спіноза, Г. Лейбніц, І. Кант, І.Г. Фіхте, Ф. Шеллінг, Г. Гегель, А. Шопенгауер, Ф. Ніцше, А. Бергсон, Н.О. Лоський, С.Л. Франк, Е. Гуссерль. Роль інтуїції в науковому пізнанні, було представлено в роботах філософів (в більшості математичному): Рене Декарт, Ж. Адамар, Папп Олександрійський, Дж. Пойа, А. Пуанкаре.

Мета дослідження: дослідження та вияв ролі інтуїції в програмуванні.

Дослідження

У програмістському пізнанні часто інтуїтивно вбачається саме дедуктивний метод програмування, істинність якого в перший момент очевидна тільки його творцеві як суб'єкту відкриття: цей творець, на відміну від інших програмістів, особистісним чином, «бачить» дедуктивно-алгоритмічний ланцюг. Програмісти прагнуть до розгортання дедуктивного методу для доказу основного виведення, тобто отримання вихідних даних, тому для них дедукція має інший напрямок покрокового поширення очевидною істинністю – таке, як і передбачено у Декарта, тобто від посилок до висновку. Тому для першовідкривача такий шлях представляється як розгортання дедуктивного ланцюга з судження, отриманого інтуїтивно, тобто за допомогою виконання процедури, зворотної декартової.

Інтуїтивне програмування – це основа творчого процесу в програмуванні. Вона проявляється, перш за все, як акт схоплювання значення важливості структури завдання або проблеми; все це відбувається без опори на розгорнуті аналітичні засоби. Дія інтуїції завжди ґрунтується на знанні предмета, її «включення» неможливо без наполегливої зосередженості на проблемі. Взагалі рівень обізнаності, необхідний для «включення» інтуїції, не піддається жодним вимірам, а його досягнення неможливо без найсерйозніших зусиль, які в цьому сенсі

ніколи не будуть надмірними. Психологічно інтуїтивного знання властива переконаність, яка дозволяє сприймати його найсерйознішим чином, незважаючи на відсутність обґрунтування, і зобов'язує вченого, дослідника наполегливо прагнути до пошуку цього обґрунтування. Важливо розуміти, що інтуїція не може бути вільною від самої упередженої перевірки і сумнівів. Правильність або помилковість результатів роботи механізму інтуїції в програмуванні встановлюється, в кінцевому рахунку, не самою інтуїцією, а методами логічного тестування.

Необхідність логічного тестування – не єдина особливість програмістської інтуїції в порівнянні з іншими видами наукової інтуїції, наприклад, в порівнянні з фізичною інтуїцією, тобто з інтуїцією, яка бере участь у фізичному пізнанні. Основна відмінність інтуїціями полягає в різному їх підставі при формуванні загальних і необхідних суджень, тобто при так званому укладанні по індукції. У судженні по індукції в програмуванні спирається тільки на могутність розуму, оскільки є підтвердження одного з властивостей самого розуму, а індукція, що застосовується в фізичних науках, спирається на віру в загальний порядок Всесвіту – порядок, який знаходиться поза нами, і тому «завжди недостовірний» [1].

В програмуванні можливий «уявний експеримент», або евристичний здогад, що є результатом дії діяльності механізму математично-програмісткою інтуїції. Така унікальна особливість підкреслює важливість дослідження природи інтуїції в програмуванні як рушійною силою програміста творчості. Програмістську інтуїцію в цьому аспекті можна охарактеризувати як осяяння, оскільки саме так прийнято називати момент «спрацювання» механізму інтуїції під час кульмінації творчого процесу.

Осяяння – це центральний етап дослідницького процесу в програмуванні. Його можна розглядати як якісний стрибок в ході пізнання, як вирішальний момент. Саме в осяяння полягає сенс і мета програміста творчості. Як правило, момент осяяння непередбачуваний, але його в ході евристичного, тобто творчого процесу його можна передчувати. Феномен раптового осяяння є об'єктом найпильнішої уваги філософів і методологів.

Ж. Адамар запропонував нині загальноприйняту в сучасній науці і філософії модель евристичного, тобто творчого процесу в науці [2]. Він виділяє наступні етапи творчого процесу в науці: по-перше, етап підготовки, коли відбувається вивчення солідних масивів літератури, що має те чи інше відношення (необов'язково пряме) до поставленого завдання або досліджуваної проблеми. По-друге, це етап інкубації, коли підсвідомість активно працює над накопиченим на етапі підготовки матеріалом. Робота на етапі інкубації відбувається не усвідомлення, і тільки рідкісні думки можуть прорватися в власне свідомість. Розробник при цьому взагалі може відключитися від роботи над основною проблемою – відпочивати або займатися підготовкою матеріалу по іншій проблемі. Ж. Адамар висуває в зв'язку з цим гіпотезу відпочинку або забуття як необхідну на цьому етапі. Далі слід безпосередньо саме осяяння як центральний етап евристичного процесу. «Спалах» осяяння – результат роботи підсвідомості. Осяяння непередбачуване. Неможливо точно визначити, коли ж воно станеться, та й чи відбудеться взагалі. Осяяння може бути відсунута за часом від попереднього етапу творчого процесу – етапу інкубації. Точніше, етап інкубації з перервами може розтягнутися надовго, оскільки осяяння – результат прихованої діяльності підсвідомості.

По суті справи, осяяння і становить основну загадку евристичного процесу. Суть осяяння – «прорив» з підсвідомості в область власне свідомості, який ми відчуваємо, як спонтанну вербалізацію. Зрозуміло, безпосередньо під час осяяння не можна вести мову про повноцінну теорію або строгих формулах, проте в результаті осяяння відбувається найголовніше – дослідник отримує якийсь первинний результат-естафету, який як би передається з підсвідомості в область свідомості для подальшої роботи над цими результатами з метою їх остаточного завершення, що і відбувається зазвичай на останньому етапі евристичного процесу – етапі перевірки. Як відкриття нового знання неможливо без участі інтуїції, так і спрацювання механізму інтуїції на етапі осяяння під час творчого процесу неможливо без евристичного впізнавання. Евристичне впізнавання можна розглядати і як дуже точний вид ізоморфізму, і як виявлення аналогії. Таким чином, дослідження Ж. Адамара можна застосувати до програмістської творчості, значить, і наукової творчості взагалі, ні в якому разі не може бути зведена до механічного перебору всіляких варіантів. Цінність творчого мислення якраз і полягає в тому, що воно дозволяє уникнути механічного перебору можливих варіантів і «підводить» суб'єкта евристичного процесу до правильного рішення.

Звісно ж, що при застосуванні синтаксису однієї мови програмування так само здійснюється евристичне впізнавання суб'єктом програміста пізнання вже відомого йому синтаксису. Це так зване евристичне впізнавання застосування: йому неодмінно має передувати евристичне впізнавання. Евристичне впізнавання застосування можна розглядати як певне «вторинне» евристичне впізнавання. Воно відповідає програмістській інтуїції, що розуміється як наочне споглядання. В ситуації рішення будь-якої нестандартної задачі, евристичне впізнавання

відкриття і евристичне впізнавання застосування як би переплітаються в єдиному творчому процесі. Їх кваліфікація як різних типів евристичного впізнавання необхідна тільки на філософсько-методологічному рівні. Для філософії і методології науки важливий, перш за все, той факт, що «вивчаючи процес мислення, ми маємо право розраховувати на проникнення в саму сутність людського розуму» [1]. Це означає, що методологія програмування дозволяє проникнути безпосередньо в роботу самого мислення, в його саму суть; і, перш за все, в цьому полягає цінність дослідження творчих аспектів програміста пізнання.

Категорії інтуїції в науці і філософії традиційно протистоять категорія логіки, а інтуїтивного судження – алгоритм, поняття якого має величезне значення в програмуванні. Концептуально алгоритм можна розглядати як аналог деякого процесу людської діяльності. Алгоритм в програмуванні визначається як чітка послідовність дій, коли при заданих умовах в результаті виконання дій гарантовано отримуємо рішення деякої задачі. При цьому випадки, коли рішення не існує, обумовлені особливо. Однак і в програмуванні, і в інших науках, що спираються на математичний апарат, завжди знайдеться достатня кількість найважливіших завдань, вирішення яких не може бути знайдено за допомогою вже відомих алгоритмів. Внаслідок цього в програмуванні завжди будуть мати велике значення методи, які на відміну від алгоритмічних не гарантують отримання рішення в результаті виконання певної послідовності дій, але, разом з тим, дозволяють наблизитися до вирішення поставлених завдань. Ці методи називаються евристичними. До них мислення програміста вдається як би спонтанно: евристичні методи, як кажуть, «приходять в голову» під час напружених роздумів над завданням. «Евристичні методи не існують, а виробляються по ходу рішення» [3]. Навіть застосовуючи широко відомих евристичних методів [4], призводить до елементів незворотності в цілях застосування цих методів до вирішення конкретного завдання. Цінність подібних методів перш за все в їх гнучкості, що дозволяє отримувати рішення нестандартної задачі (наприклад, в штучному інтелекті) за відсутності для неї алгоритмічного рішення.

Традиційно під програмістською евристикою розуміють сукупність евристичних засобів, тобто методів, функцій програмування, що застосовуються для вирішення складних завдань, для яких не існує стандартних алгоритмів. «Евристичний» в даному контексті означає «незворотний», що містить елемент здогадки, і внаслідок цього повністю неалгоритмічний [5]. Не можна сприймати програмістську евристику тільки як деякий звід правил, яких слід дотримуватися при написанні коду, щоб отримати результат за більш короткий проміжок часу тим же алгоритмічним способом, без будь-якого якісної зміни звичного підходу написання коду програми. Таке уявлення про методи програміста пізнання помилково. Евристика дійсно буває різною залежно від ступеня складності завдання – її може бути «більше» або «менше» в залежності від ступеня складності завдання, але без виникнення здогадки, значить, і без участі інтуїції евристичний процес неможливий. Евристика має величезне значення в програмуванні. Рішення серйозного завдання для програміста, не обходиться без застосування евристичних методів і функцій.

Отже, евристичні методи, на відміну від алгоритмічних, дозволяють відступити від жорсткої схеми в рішенні нестандартної задачі. Корисність програмістської евристики полягає перш за все в тому, що її застосування дозволяє значно скоротити число перебираються алгоритмів при пошуку рішення нестандартної задачі. Вкрай важливим є факт, що таке скорочення досягається в результаті інтуїтивного вибору, в основі якого лежить евристичне впізнавання аналогій, тобто елементів знайомого в новому. Функція здогадки якраз і полягає в виявленні цих аналогій. Гіпотеза і аналогія – це «ключ» до евристики, її визначальна ознака. Тому принципово недостатньо визначати евристику тільки як систему правил, що визначають таку тактику вирішення завдання, яка істотно обмежує перебір методів її рішення, оскільки вказане скорочення числа передбачуваних методів вирішення задачі можливо тільки завдяки виникненню здогади. Строгий метод в програмуванні на відміну від евристичного носить алгоритмічний характер і є стандартним. Застосування суворого методу не пов'язане з виникненням здогадки, яка, навпаки, є необхідним елементом евристичного методу в програмуванні. При вирішенні будь-якої нестандартної задачі мислення програміста може продукувати цілу серію припущень – в залежності від складності поставленого завдання.

Основна праця Паппа Олександрійського – «Математичне збори» - містить великі коментарі до робіт класиків давньогрецької геометрії, зокрема, до Евкліду. У своєму трактаті «Мистецтво вирішувати завдання» Папп Олександрійський виділив як мінімум два евристичних прийому: регресивне міркування до початку, або навпаки – прогресивне міркування; а також пов'язаний з цим дією прийом до визначення мети і даних за допомогою виведення при цьому наслідків; процедура другого прийому відзначалася Паппом неявно. За весь час свого існування в якості апарату рішення задач різного характеру, програмування з математики накопичила неосяжну кількість різних евристичних прийомів, методів і функцій. При вирішенні особливо складних завдань використовуються в різних поєднаннях найрізноманітніші евристичні прийоми та методи. Чим складніше завдання, тим, зрозуміло, різноманітніше застосовується при її вирішенні еристика [6].

Відзначимо той примітний факт, що практично кожен автор «бачить» евристичні методи по-своєму і дає їх оригінальну класифікацію. Зрозуміло, що пошукова евристика строго індивідуальна. Накопичивши достатній досвід, людина часто неусвідомлена слід цього досвіду. Пошукова евристика зосереджує в собі досвід наукової творчості, і цей досвід воістину безцінний. Програміст, на основі систематизації своєї індивідуальної пошукової евристики, виробляє свою власну неповторну евристичну стратегію.

При використанні евристичних стратегій Дж. Пойа, застосовуються різні евристичні методи і функції. Це означає, що поняття евристичної стратегії є узагальнюючим і організуючим по відношенню до понять евристичної функції або методу. Евристичні стратегії завжди мають певну послідовність, оскільки кожна з них переслідує певну мету. Цікаво, що для Дж. Пойа характерний кілька своєрідний підхід до визначення поняття «евристичний». Міркування евристичного характеру, що містять здогад, він визначає, як «правдоподібні». Виникнення здогади викликає інтерес в основному тому, що здогад неможливо планувати, передбачати.

Евристика по суті своїй, за визначенням, не може бути універсальним алгоритмом. Цю думку підтверджує і звернення до робіт засновника кібернетики Н. Вінера, який вважав, що «послідовності думок не узгоджуються з законами логіки» [7]. Як би не була ефективна евристична стратегія того чи іншого програміста, необхідно розуміти, що ніяка евристика не може скасувати необхідність часто болісного пошуку здогади, і будь-яка евристика містить звернення до інтуїції. В цілому можна зробити висновок про необхідність втручання інтуїції в процес програмування в наступних ситуаціях: 1) при конкретизації евристичних функцій і методів, тобто при їх інтерпретації в контексті умов конкретного завдання; 2) при поширенні вже відомої евристики на інший тип завдань, нехарактерний для даної евристики; 3) при перенесенні на програмування евристики, що застосовується в інших науках, наприклад, у фізиці; 4) при зміні евристичних прийомів [4].

Важливо відзначити, що, незважаючи на очевидну важливість дослідження значення інтуїції як рушійної сили евристичного процесу в програмуванні, одним тільки цим її роль у розвитку програмування не вичерпується.

У програмуванні також застосовується топологічна інтуїція як окремий тип інтуїції, відмінний від інтуїції власне програмісткою [7]. Звісно ж, що використання нового терміна правомірно розглядати як вираз філософсько-методологічного прагнення до повного відділення топологічної інтуїції від інтуїції в програмуванні, причому незважаючи на те, що топологія за визначенням є областю математики.

Всю важливість ідей топології для сучасної математики, а також для програмування добре усвідомлював, французький математик другої половини двадцятого століття А. Вейль. За його висновками вся сучасна математика в цілому являє собою химерне переплетення ідей топології і алгебри. Специфіка топології в області математики і програмування пов'язана з певними особливостями перетворення фігур в просторі. Основними поняттями в топології є поняття гомеоморфізму і конгруентності. Кажуть, що фігури конгруентні, якщо відстані між їх відповідними точками рівні. Гомеоморфізм в топології є відображення без розривів, склеювання, а також без утворення складок. Основною ідеєю топології є ідея безперервності.

Специфіка топологічної інтуїції пов'язана з формами і розташуванням фігур в просторі. Це означає, що топологічна евристика, тобто евристика, що застосовується в топології, повинна бути пов'язана з топологічними перетвореннями фігур, тобто з властивостями фігур, які не змінюються при гомеоморфізмі, і повинна включати в себе такі прийоми, як розтягування і стиснення фігур. Сама топологія «працює» в просторі, і передбачає введення таких понять, як вузли, розшарування, накриття.

Топологічна евристика – це дійсно якісно нова евристика, що активно застосовується в програмуванні, наприклад, для відображення безперервних 3D об'єктів, яка на відміну від колишньої евристики, не пов'язаної з просторовими перетвореннями, проводить досить вільні перетворення фігур в просторі, нехай і обмежені певними умовами.

До освіти топології як області в програмуванні просторова інтуїція відбувалася головне на художньому рівні, де її прояви, звичайно ж, вимагали виняткової конкретності, і не могли бути абстрактними, як в топології. Ця особливість топологічної інтуїції дозволяє припустити, що, в кінцевому рахунку, всі види інтуїції, властиві людині, взаємопов'язані, і людина здатна в силу історичної логіки пізнання усвідомити цей зв'язок.

В даному контексті неминує виникає питання про можливість проведення повної і остаточної раціоналізації результатів діяльності інтуїції в програмуванні – тобто повного і остаточного перекладу інтуїтивних здогадок, отриманих в результаті діяльності інтуїції під час евристичного процесу в дискурсивні, тобто доказові, міркування – тим більше, що останнім етапом евристичного процесу в програмуванні є тестування, під час якого, за визначенням, здійснюється максимально можливий переклад інтуїтивних міркувань в доказові. Ця особливість евристичного процесу в програмуванні дозволяє певною мірою говорити про деяку

схожість структури інтуїтивних і дискурсивних міркувань. Однак ця подібність відноситься скоріше до змісту міркувань, ніж до їх форми

Висновки та результати досліджень

Робота інтуїції необхідна, по-перше, для отримання інтуїтивних суджень, і, по-друге, для перекладу інтуїтивних суджень в дискурсивні. Це означає, що така раціоналізація є ні що інше, як творчий процес, при якому здійснюється подальший переклад деяких «порцій» інтуїтивних міркувань в міркування дискурсивні. Кожна «порція» інтуїтивних міркувань при цьому перетворюється в дискурсивні під час єдиного повторення докази. В цілому ж вся раціоналізація по відношенню до одного конкретного доказу являє собою розумовий творчий процес, який можна охарактеризувати як багатшаровий. Багатшаровість як властивість програміста обґрунтування означає, що, при раціоналізації докази, до вже наявних не раціоналізованим залишкам роботи інтуїції завжди будуть додаватися нові, такі ж нераціональні, тобто інтуїтивні.

Діяльність інтуїції в процесі процедури програміста обґрунтування носить двоїстий характер: з одного боку, сприяє раціоналізації інтуїтивних елементів, а з іншого боку – її ж і ускладнює, оскільки продукує нові. Контекст обґрунтування, як це впливає з проведених міркувань, також не вільний від втручання інтуїції. Тому контекст програміста обґрунтування не є суто логічним і алгоритмічним. Загадка інтуїції існує, і вона аж ніяк не відкривається в процесі дослідження її специфіки.

REFERENCES

- [1] Poincare, A. (1990). *About science*. M.: Science.
- [2] Hadamard, J. (1970). *The study of the psychology of the process of invention in the field of mathematics*. M.: Soviet radio.
- [3] Polya, D. (1957). *Mathematics and plausible reasoning*. M.: Science.
- [4] Ilyasov, I.I. (1992). *The system of heuristic methods for solving problems*. M.: ROU.
- [5] Polani, M. (1985). *Personal knowledge*. M.: Progress.
- [6] Van der Warden, B.L. (1959). *Awakening science*. M.: GIFML.
- [7] Wiener, N. (1958). *Cybernetics*. M.: Science.

Buzunova M.I., Patlaychuk O.V.

Investigation of the role of intuition and heuristics in programming

Abstract. *The main subject of the research in the article is programmatic heuristics as a collection of methods for solving non-standard problems in programming, that is, such tasks for which there are no known algorithms. As a specific mechanism of thinking, generates elements of guessing on the basis of heuristics, intuition is taken. The article is based on the conclusions obtained by J. Hadamard, N. Winner, R. Descartes, A. Poincare and J. Piaget.*

Keywords: *heuristics, programming, intuition, philosophy.*

Бузунова М.И., Патлайчук О.В.

Исследование роли интуиции и эвристики в программировании

Аннотация. *Основным предметом исследования в статье является программистская эвристика как совокупность методов решения нестандартных задач в программировании, то есть таких задач, для которых не существует известных алгоритмов. Как специфический механизм мышления, порождает элементы догадки по основанию эвристики, берется интуиция. Статья основывается на выводах, полученных Ж. Адамара, Н. Винером, Р. Декартом, А. Пуанкаре и Ж. Пиаже.*

Ключевые слова: *эвристика, программирование, интуиция, философия.*