

# КОМП'ЮТЕРНА ПРОГРАМА ЯК ОБ'ЄКТ ПІЗНАВАЛЬНИХ ДІЙ

## М.Б. Головін

Програмування є не тільки цікавим предметом вивчення, але й зручним об'єктом для психолого-педагогічних спостережень. Цей предмет міг би стати полігоном для відпрацювання нових технологій навчання. Унікальність програмування як полігону для психолого-педагогічних досліджень полягає в наступному.

- Комп'ютерна програма має зручну специфіку для спостереження за розумовими діями стосовно її освоєння. Програму можна представити як багаторівневу ієрархічну структуру. Окремі елементи цієї структури логічно зв'язані і можуть входити одна в одну, як складові частини.

- Заняття з програмування проводяться в дисплейному класі, і тому в розпорядженні викладача завжди є такі потужні технічні засоби навчання, як обчислювальні машини, об'єднані мережею.

- Навчання відбувається у файловій системі комп'ютера, стан якої легко можна контролювати за допомогою спеціальних навчальних програм. Це може забезпечити детальний моніторинг пізнавальних процесів, які відбуваються при взаємодії студента з комп'ютером.

- Навчальна програма, крім пасивного моніторингу, може забезпечити і активне керування пізнавальним процесом.

Для адекватного керування навчанням важливим є зворотній зв'язок між навчальними впливами на процес пізнання та відображенням цього процесу в файловій системі.

**Комп'ютерна програма як об'єкт пізнавальних дій** має зручну специфіку для дослідження. У її склад входять функціональні вузли і структури керування. Структури керування задають порядок виконання функціональних вузлів. Вузол виконує у програмі визначену йому функцію і має одне вхідне і одне вихідне сполучення з іншими програмними конструкціями. Функціональний вузол як об'єкт розумових дій може бути представлений як окремим оператором мови програмування, так і цілим логічно завершеним програмним фрагментом. У самому вузлі може бути своя локальна керуюча структура, що управляє функціональними вузлами нижчого рівня. Саме цей аспект дозволяє розглядати готову програму як багаторівневу логічну ієрархічну структуру.

У програмі дані проходять шлях від вхідних до вихідних через ланцюг проміжних параметрів. Ці програмні проміжні параметри для вузлів є локальними вхідними і вихідним. Шлях обробки інформації в програмі визначається її структурою керування і значеннями деяких проміжних параметрів.

Така структура, з одного боку, при її вивченні легко піддається аналітичному багатократному подрібненню на логічно закінчені функціональні вузли, з іншого – нові функціональні вузли при синтезі програми легко включаються в її текст як окремі логічно завершені модулі.

Керуючі структури будь-якої складності можна представити як сукупність слідувань і виборів. Цикл, наприклад, – це вибір із слідуванням на повторення. „Вибір” і „Слідування” є природними логічними діями, що використовують у майже не завуальованому вигляді конструкції процедурного мислення [1].

### ***Умови проведення педагогіко-психологічного експерименту***

Дослідження планувались у контексті реальних навчальних дій із освоєння аплікаційної програми. Ці навчальні дії студентів відбувались у внутрішньому розумовому, абстрактно-логічному режимі. Виявлення механізмів такої роботи є непростою задачею. Коректність уявлень про механізми пізнавальних процесів та їхній операціонізм може дозволити підвести теоретичну основу для вдосконалення сучасних методик навчання.

**Програма для вивчення** виглядала як суцільний текст з максимально довгими рядками і завуальованою ієрархічною програмною структурою.

**Демаскуючі завдання.** Для дослідження механізмів внутрішніх пізнавальних процесів у цій роботі використані спеціальні завдання. Передбачалося, що при виконанні завдань відбудеться

демаскування внутрішньої абстрактно-логічної роботи супутніми матеріально-практичними діями. Демаскуючі завдання були спрямовані на виявлення у програмі конкретних ключових функціональних вузлів, які знаходились на різних рівнях програмної ієрархії. У завданні вузли описувались їхніми функціями. Порядок опису функцій вузлів у списку був випадковим. Завдання, крім опису вузлів, які потрібно знайти, включали узагальнені коментарі щодо специфіки роботи всієї програми в цілому.

**Сукупні знання, на основі яких відбувалось освоєння аплікаційної програми,** складались із щойно згаданих узагальнених відомостей про роботу програми в цілому та знань, отриманих на попередніх заняттях з програмування. Це знання про специфіку роботи кожного окремого оператора, процедури, функції, що використовувались у програмі. Допускалось, щоб частина операторів, функцій та процедур була розглянута на лекційному занятті безпосередньо перед психологічним експериментом. Рівень цих окремих операторів – неподільних програмних одиниць – це нижній ієрархічний рівень програми. Узагальнені відомості про механізми роботи програми є верхнім рівнем ієрархії програми.

Процес освоєння програми вважався завершеним, а завдання виконаним, якщо студенти знаходили у програмі ключові вузли проміжних рівнів програмної ієрархії, що фігурували у завданні.

**Формалізація уявлень про пізнавальні дії** здійснювалась у термінології прийомів логічного мислення, таких як аналіз, синтез, дедукція, індукція, ідеалізація, абстрагування, конкретизація, узагальнення.

**Очікувались наступні результати експерименту.** Протягом надання тексту програми структурного вигляду передбачалось, що відбудеться демаскування початкових аналітико-дедуктивних розумових дій. Візуалізація дедуктивно-індуктивного характеру подальших логічних дій проявиться в тому, що вузли будуть пред'являтися студентами при виконанні завдання в порядку підвищення рівня їхньої ієрархії. Такий порядок можливий, якщо формування уявлення про функції кожного нового вузла N+1 рівня відбудеться в результаті зв'язування відповідної йому частини вузлів N-ного рівня. Механізм абстрактно-логічного покрокового зв'язування вузлів у новий вузол може мати місце тільки тоді, коли стануть відомими функції всіх вузлів, що потенційно ввійдуть у цей створюваний вузол.

Послідовність пред'явлення вузлів побічно може підтвердити наявність у розумових діях синтетичної складової. Синтетична дія може трактуватись тут як завершальна для індуктивно-дедуктивної.

#### **Модельні уявлення та експериментальні результати**

**На першому етапі дослідження, назвемо його аналітико-дедуктивним,** візуалізувалась структура програми.

Для виконання описаного вище завдання на співставлення вузлів з їхніми функціями абсолютна більшість студентів спочатку структурували програму. (Бралась до уваги робота тільки тих студентів, що швидко досягали позитивного результату в виконанні завдання). Візуалізація полягала в наступному. Суцільний текст програми розривався на короткі логічно завершені рядки. У кожному рядку – або оператор керування, або функціональний вузол. Усі керовані рядки стосовно керуючих зміщувались вправо. Рядки, що не були у відношеннях керуючий – керований, розміщувались строго один під одним у стовпці. Важливим моментом цього етапу є те, що в текст додавались тільки знаки „пробіл” та „кінець рядка”, які не впливають на змістову частину програми, тобто на її виконання.

Розглянемо пізнавальні дії аналітично-дедуктивного етапу за допомогою моделі пізнання, представленої в роботі [3] (рис.1). Репрезентація програми в когнітивній структурі **направляє дослідження на вибір** нової інформації в об'єкті. Цей вибір має два кроки: аналітичний і дедуктивний. Аналітичний виділяє із суцільного тексту програми черговий оператор. Дедуктивний крок встановлює місце вибраного оператора у вже існуючій програмній структурі. Цей вибраний в об'єкті (програмі) оператор **модифікує** репрезентацію програми у **когнітивній структурі**. Наступна ітерація цього циклу дозволяє виділити новий оператор і знайти його місце в ієрархічній програмній структурі. Кожна наступна ітерація неможлива без

попередньої. Описані дії поступово формують загальну ієрархічну картину. У цій картині проявляться функціональні вузли, що знаходяться в ієрархії вище рівня окремих операторів і нижче верхнього ієрархічного рівня. Функції цих вузлів ще невідомі, але їх можна побачити за формальними ознаками, зокрема за відступами операторів від краю екрану і за розмашуванням слів „Begin”, „End”.

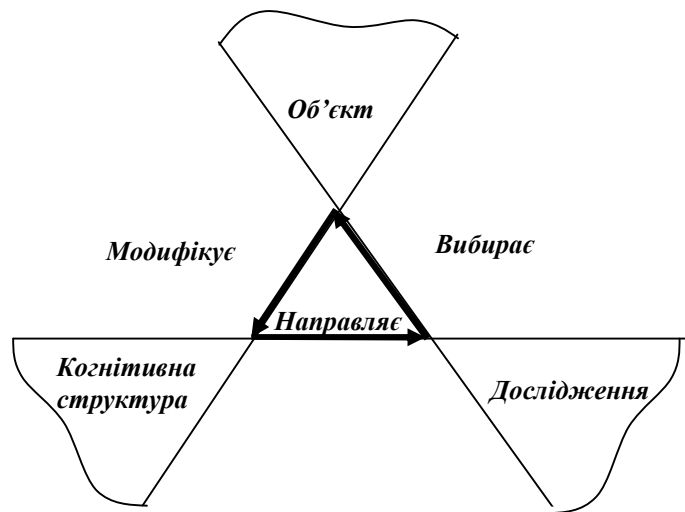


Рис.1. Процес пізнання об'єкта дослідження

Дії студентів щодо формування ієрархічної структури дійсно були взаємозв'язані. Знаходження місця в ієрархічній картині кожного конкретного вузла можливе лише тоді, коли знайдені місця в цій картині всіх вузлів, куди цей конкретний вузол входить.

Спостереження за роботою студентів показують, що вони мають намір розчленовувати програму до рівня, коли функції окремих вузлів стають зрозумілими. Межі вузлів, як правило, можна побачити за формальними ознаками. (У мові програмування Delphi це, зокрема, слова „Begin”, „End” на відповідних рівнях ієрархії).

Аналіз такого добре структурованого, ієрархічно багаторівневого об'єкта вивчення, як аплікаційна програма, здійснюється студентами з різною швидкістю. Студенти-експерти, що цікавляться предметом і багато часу приділяють складанню програм, значно швидше роблять аналіз, ніж звичайні студенти.

Спостереження за студентами-експертами та відслідковування їхніх поточних коментарів під час роботи з програмним продуктом призводять до наступних висновків. Під час аналітичної розумової роботи студенти-експерти не розчленовують програму до неподільного рівня – рівня окремих операторів. Вони бачать паттерни операторів у тексті програми.

Програмний паттерн – це стандартний програмний фрагмент, що вже неодноразово зустрічався експертові. Йому не треба детально досліджувати роботу паттерна. Він знає його функції і має уявлення про те, як спрацює фрагмент програми, що входить у паттерн, і без досліджень. Експертам не потрібно, як звичайним студентам, аналітично розчленовувати текст програми до рівня окремих операторів. Фрагменти програми, де застосовані нові для студентів-експертів процедури і функції, аналізуються до найнижчого рівня, як звичайними студентами. Приклад використання паттернів при розв'язуванні шахових задач наведено в роботі Р. Джона [1]. Там розглядалися паттерни експертів-шахістів. У роботі зазначалось, що в паттерни експертів-шахістів входили стандартні шахові ситуації. Показано, що шахові експерти знали рішення багатьох проблем, які гравці початківці мали розв'язувати як нові.

Враховуючи специфіку програмування, можна вважати, що тут паттерни можуть бути трьох типів: керування, функціональні вузли, комбіновані. У паттерн керування входять тільки керуючі структури, наприклад, вкладені цикли, або вкладені розгалуження. Функціональний вузол містить послідовність стандартних процедур, наприклад, щодо відкриття файлу. У комбінованому випадку важливими є і функціональна, і керуюча складова паттерну.

У кінці аналітико-дедуктивного етапу відомі функції вузлів верхнього та нижнього рівнів. Сформовані уявлення про формальний вигляд вузлів середнього рівня в програмі, але невідомі ще їхні функції. У цих умовах спонтанно формується пізнавальне завдання наступного етапу. Воно полягає в тому, щоб сформулювати уявлення про застосування окремих операторів, процедур і функцій у складі вузлів середнього рівня.

**На другому етапі дослідження, переважно дедуктивно-індуктивному,** зв'язуються вузли окремого ієрархічного рівня.

Дедукція, як відомо, – це ланцюг розумових логічних кроків виводу від початкових узагальнених посилок до конкретних наслідків. Початковими посилюваннями, в цьому випадку є текст програми. Він має узагальнений характер, оскільки, як правило, може працювати з цілим спектром конкретних вхідних параметрів. Конкретними наслідками є вихідні дані, отримані після виконання програми з відповідними вхідними параметрами. У розумових дедуктивних дослідженнях використовуються правила, які регламентують роботу структур керування з вузлами і обробку цими вузлами інформації. Ці дедуктивні дослідження полягають у відслідковуванні логічних зв'язків між вхідними і вихідними даними через ланцюги проміжних параметрів.

Модель, представлена на рис. 1, теж непогано описує дедуктивно-індуктивний етап роботи. Робота моделі тут теж циклічна і кожна її наступна ітерація базується на попередній і неможлива без неї. **Когнітивна структура направляє** дедуктивне **дослідження** з вивчення структур керування, що визначають порядок виконання окремих операторів. Проміжні параметри, що зв'язують вузли, теж є предметом дослідження. Поповнення уявлень про структури керування і зв'язки між параметрами відбувається циклічно. Кожна ітерація дозволяє **вибрати в об'єкті** порцію інформації і **модифікувати** нею репрезентацію програми. У складних програмах потрібно багато дедуктивних ітерацій циклу для формування уявлень про зв'язки рівня.

Частину досліджень в цьому циклі можна ідентифікувати як дедуктивні, адже вони відбуваються від загального до конкретного, а частину – як індуктивні. У процесі пізнавальних дій не завжди можна отримати значення деяких проміжних параметрів, що визначають зв'язки між вузлами програмного рівня. Значення таких параметрів конкретизуються припущеннями. Розумові дії в цьому випадку проходять в індуктивному режимі.

Спостереження показали, що в дедуктивному режимі студенти-експерти теж працювали швидше, ніж звичайні студенти. Аналіз коментарів студентів-експертів при їхній дедуктивній роботі з програмою показав наступне. Ці студенти діють швидше за рахунок того, що їм не треба досліджувати стандартні структури керування. У своїх дослідженнях програми вони використовують відповідні паттерни керування.

Логічне дедуктивне дослідження зв'язків функціональних вузлів приводить до подальшої диференціації репрезентативної картини аплікаційної програми у когнітивній структурі студента.

Бачення студентами ієрархічних рівнів у програмній структурі є суб'єктивним. Студенти-експерти схильні пропускати проміжні ієрархічні рівні у своїх коментарях до роботи. Студенти, що мають високу мотивацію в навчанні, хороші здібності до логічного та аналітичного мислення, але невелику практику в програмуванні, бачать багато проміжних рівнів програмної ієрархії. Ці рівні дійсно існують, але експерти на них не звертають уваги і маніпулюють цілими програмними паттернами як окремими програмними одиницями. Це значно зменшує кількість нових об'єктів у розумових діях експертів. Проявляється це у експертів, в порівнянні з іншими студентами, у вигляді сильно узагальнених коментарів до роботи. Спроби студентів з малою практикою в програмуванні пропускати ієрархічні рівні, як це роблять експерти, не приводить до позитивного навчального результату. При дедуктивному створенні репрезентації зв'язків між вузлами у студентів з малою практикою роботи виникають проблеми. Наявність проблем можна пояснити властивостями оперативної пам'яті [2]. Кількість нових вузлів, функціями яких одночасно треба маніпулювати в розумових діях, у цих випадках перевищує граничне число сім.

Важливим прийомом логічного мислення є абстрагування. Логічні дії стосовно зв'язків вузлів на кожному конкретному рівні ієрархічної програмної структури майже завжди супроводжуються абстрагуванням від вузлів, не задіяних у конкретному контексті розумових дій.

Формування уявлень у репрезентації програми про новий функціональний вузол із сукупності уявлень про вузли нижчих рівнів можна розцінювати як синтетичну розумову дію. Ця дія можлива тільки в ланцюгу з аналогічними попередніми синтетичними діями з формування уявлень про вузли нижчих рівнів. Формування репрезентації конкретного програмного вузла в когнітивній структурі індивідуума можливе, коли відомі функції всіх вузлів, з яких він має складатись. Виділення сукупності вузлів для розумового синтезу відбувається під час дедуктивного дослідження зв'язків всієї сукупності доступних для розгляду вузлів. При формуванні уявлень про функції нового вузла відбувається абстрагування від функцій вузлів нижнього рівня, що не ввійшли до нового вузла. Крім того, при формуванні уявлень про функції нового вузла узагальнюються уявлення про те, що відбувається в ньому. Частина програмних процесів не проявляється у вигляді проміжних параметрів на поточному ієрархічному рівні. Деякі з таких процесів можна вважати несуттєвими, технологічними. Функції вузла ідеалізуються за рахунок цих процесів.

**Третій етап дослідження переважно індуктивно-синтетичний.** На цьому етапі формуються уявлення про нові функціональні вузли вищого рівня.

Цей процес теж добре описується моделлю, представленою на рис.1. У когнітивній структурі на початку індуктивно-синтетичних дій сформовані загальні уявлення про роботу програми в цілому, про оператори найнижчого ієрархічного рівня, структури їх керування і проміжні параметри, що їх зв'язують.

Синтетичне **дослідження вибирає з об'єкта** (тексту програми) групу вузлів першого рівня і **модифікує** репрезентацію програми в **когнітивній структурі**, синтезуючи перший вузол другого ієрархічного рівня. Наступні ітерації індуктивно-синтетичного циклу формують уявлення про функції інших вузлів другого рівня. Необхідно відзначити, що всі індуктивно-синтетичні дії мають розумовий абстрактно-логічний характер. Вони формують уявлення про функції вузлів вищого рівня, спираючись на функції, структури керування і зв'язки вузлів нижчого рівня. При формуванні таких уявлень відбувається **узагальнення** функцій вузлів нижчого рівня. Це індуктивний шлях розумових дій.

Після закінчення формування уявлень про функції вузлів нового (другого) рівня індуктивно-синтетичний цикл змінюється на дедуктивно-індуктивний, який досліджує зв'язки нових вузлів другого рівня. Коли репрезентація про ці зв'язки сформується, знову включається індуктивно-синтетичний цикл розумових дій. Відбувається формування уявлень про функції вузлів третього рівня.

Спостереження показують, що коли поступає завдання знайти вузли із заданими функціями, студенти в першу чергу пред'являють вузли, що знаходяться на нижньому рівні ієрархії. Надалі вузли пред'являються в порядку збільшення рівня їхньої ієрархії. Порядок представлення вузлів корелює з їхнім положенням в ієрархічній структурі програми.

Дослідження коментарів, якими супроводжувались дедуктивні розумові дії студентів щодо з'ясування функції окремого конкретного вузла, що знаходиться на високому ієрархічному рівні, показують наступне. Студенти схильні використовувати функції вузлів нижніх рівнів, що мають внутрішню складну структуру, як одиничні елементи розумових дій.

Таким чином, знаходження функції конкретного вузла можливе лише тоді, коли знайдені функції всіх вузлів, що входять до цього конкретного вузла. Тому формування репрезентації ієрархічної програмної структури у когнітивній структурі студента може відбуватись тільки циклічно в ході послідовного переходу його уваги від вузла до вузла вгору ієрархічною структурою.

Механізм роботи моделі, представленої на рис.1, базується на передбаченні результатів дослідження [3]. Інтроспективні дослідження процесів передбачення в контексті освоєння програмних продуктів приводять до наступного тлумачення передбачення.

Передбачення відбувається миттєво. Часто це імпліцитна розумова дія, що веде до самовільного утворення гіпотези. Інколи передбачення має експліцитний характер. Це передбачення можна класифікувати як спонтанне індуктивне формування узагальнених розумових висновків за конкретизованими неповними даними. Місце таких спонтанних розумових індуктивних дій серед інших розумових дій, очевидно, у визначенні напрямку подальших деталізованих індуктивних, дедуктивних, аналітичних або синтетичних досліджень. Ці дослідження можуть закінчитись підтвердженням гіпотези або її спростуванням. У контексті визначення функцій вузла такі індуктивні дослідження відбуваються перед синтетичними розумовими діями.

**Узагальнення** спостережень за роботою студентів приводить до формування наступних уявлень про послідовність логічних дій з освоєння програми. Аналітико-дедуктивна робота над текстом програми віртуально пересуває увагу вниз ієрархічною програмною структурою. Результатом такої роботи є створення уявлень про окремі елементи програмної структури нижнього рівня. Подальша почергова дедуктивно-індуктивна та індуктивно-синтетична розумова робота приводить до віртуального пересування уваги вгору ієрархічними програмними рівнями. Процес пересування супроводжується послідовним (знизу-вгору) формуванням уявлень про функції вузлів на всіх рівнях програмної ієрархії.

Глибина занурення в програмну ієрархію для різних студентів різна. Наявність паттернів вузлів, що складаються з кількох операторів, у студентів-експертів знижує „глибину занурення” при аналітико-дедуктивному пересуванні вниз. Швидкість пересування вниз у цих студентів вища, бо вони швидше, за формальними ознаками рядків, створюють уявлення про ієрархію програми. Наявність паттернів структур керування пришвидшує пересування вгору ієрархічними рівнями. Таким чином, виграш по часу в освоєнні програми студентами-експертами складається з двох аспектів. Шлях, який вони проходять при русі вниз, а потім відповідно і вгору ієрархічними рівнями, менший, ніж у звичайних студентів. Крім того, швидкість при цих пересуваннях у них значно вища.

Здійснення в реальному часі описаних вище розумових дій над текстом програми приводить до поступової диференціації уявлень про те, як працює аплікаційна програма. Програмні фрагменти, що часто зустрічаються у аплікаційних програмах, запам’ятовуються разом з їхніми функціями як окремі програмні паттерни. Освоєння кожної наступної аплікації, що несе чергова лекційна або лабораторна програма, диференціює загальну когнітивну структуру [4].

#### **Література**

1. Джон Р. Андерсен Когнитивная психология. СПб.: Питер, 2002. – 492с.
2. Когнитивная психология. //Под ред. В.Н. Дружинина, Д.В. Ушакова. – М.: ПЕР СЭ, 2002. – 477 с.
3. Найсер У. Познание и реальность. Смысл и принципы когнитивной психологии. М.: Прогрес, 1981. – 225с.
4. Чуприкова Н. И. Психология умственного развития: Принцип дифференциации. –М.: Столетие, 1997. – 478 с.