

РОЛЬ САМОСТІЙНОЇ РОБОТИ З ПРАКТИЧНОГО ПРОГРАМУВАННЯ В СИСТЕМІ ПІДГОТОВКИ СТУДЕНТІВ ФІЗИКІВ

Головін М.Б.

Розглянуто навчальне програмування в контексті структурної організації ментального відображення об'єктів цієї діяльності.

Educational programming in the context of structural organization of this activity objects' mental reflection are considered.

Практичне програмування є надзвичайно важливим полем навчальної діяльності студентів фізиків. Оригінальні фізичні дослідження, як правило, вимагають не стандартно скомпонованого обладнання. Програмне забезпечення для такої специфічної фізичної установки, зазвичай ексклюзивне. Ніхто більш тонко, ніж сам дослідник, не розуміє специфіку експерименту, не може налаштувати установку і написати для неї програмне забезпечення. Іншою важливою областю, де може бути застосоване програмування фізиком, є модельні дослідження, математична обробка результатів експерименту та обчислювальні задачі.

Моделювання фізичних процесів є цікавим завданням для фізика як спеціаліста. Ці завдання дозволяють поглибити знання в області фізики, математики та програмування. Вони вимагають творчого креативного мислення. Багато студентів фізиків вибирають теми курсових, дипломних і магістерських робіт таким чином, щоб ця навчальна діяльність була пов'язана з програмуванням. Виконання великих програмних проектів в режимі самостійної роботи вимагає специфічних умінь та навичок. Однак, ця діяльність абсолютно неможлива, якщо на молодших курсах при виконанні базових завдань не була проведена належна практика з програмування.

Специфіка програмування в тому, що зразу після засвоєння порції декларативних знань передбачається використання їх в практичній діяльності максимально наближеній до професійної. Це вимагає швидкого включення нової порції зовнішньої інформації в систему існуючих у суб'єкта діяльності знань. Не всі навчальні дисципліни надають таку можливість. Слово „розуміння” набуває тут конкретного звучання. Якщо суб'єкт навчання може застосовувати новий навчальний об'єкт в діяльності, то він розуміє його сутність. Аналогічно, алгоритмічна конструкція з багатьох елементів є зрозумілою, якщо вона може бути успішно модифікована, модернізована, виправлена, інтегрована з іншим подібним утворенням. Ця сфера знань абсолютно не приймає навчальних процесів, що будуються на вербальному декларативному „зазубрюванні” догматів. Практика самостійного практичного використання знань впродовж програмування надзвичайно важлива.

Розв'язування задач з програмування, як і розв'язування задач природничого циклу, зокрема з фізики, вимагає крім причинно-наслідкового мислення, витонченої розгалуженої системної структури професійних знань. Діяльність первинна в цій сфері. Вона формує процеси сприйняття, спосіб диференціації та структурування знань, особливості зв'язків у понятійній сфері, математичний апарат, пріоритети в навчанні.

Існує специфіка практичної навчальної діяльності стосовно складних добре формалізованих штучних об'єктів, таких зокрема, як електронні пристрої та програмні засоби. Ця специфіка полягає в тому, що людина не може оперувати одночасно всіма багаточисельними функціональними вузлами складного об'єкта, які вона сприймає і розрізняє. Суб'єкт навчання змушений у процесі освоєння, створення, модифікації, відлагодження об'єкта переводити увагу і проводити абстрактно – логічні дії для зв'язування цих вузлів. У результаті у довготривалій пам'яті поступово формується ментальна структура ієрархічного типу - пізнавальна (когнітивна) схема об'єкта.

Ієрархічність схеми надає суб'єкту діяльності важливу спроможність - здатність до масштабування уявлень про об'єкт у процесі діяльності.

Актуальною проблемою в контексті самостійної практичної навчальної діяльності та її оптимізації у сфері високих технологій є деталізація уявлень про когнітивну схему об'єкта діяльності та про розумові процеси, що формують цю ментальну структуру. Цікавим питанням також є формалізація оптимізованого протоколу пересування вже сформованим ментальним відображенням складних об'єктів з метою їх відлагодження або модернізації. Швидкий серфінг такою ментальною структурою забезпечує швидку практичну діяльність стосовно об'єкта. Іншим важливим моментом цікавим для розгляду є акумулювання конкретизованих ментальних структур зв'язаних з окремими об'єктами (процесами, феноменами) в узагальненій структурі знань суб'єкта діяльності.

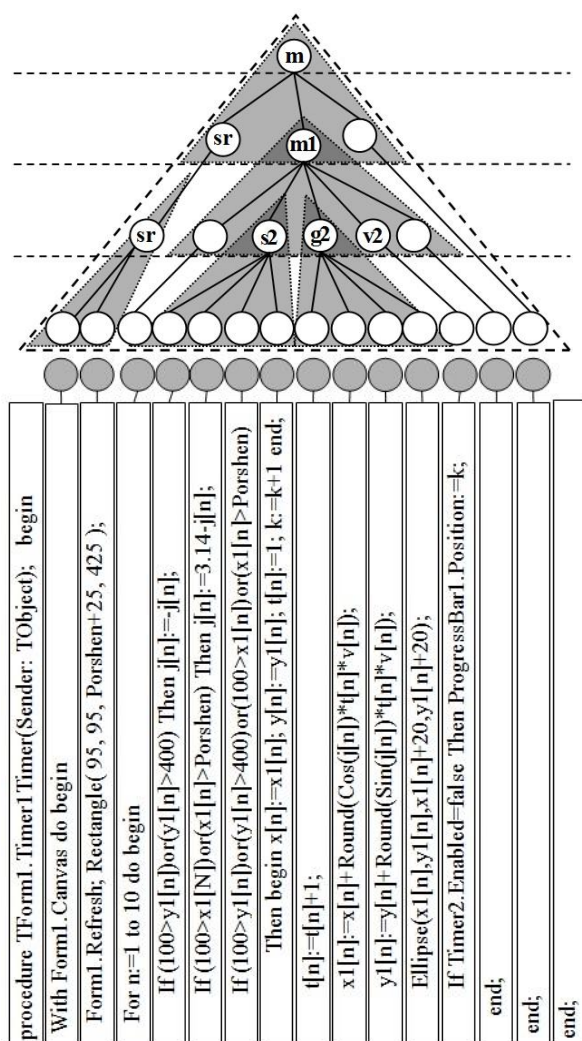


Рис.1 Когнітивна схема навчального об'єкта та відповідний йому текст фрагменту програми.

Метою цієї роботи є формалізація уявлень про ментальні процеси, що відбуваються впродовж самостійної тривалої практичної навчальної діяльності стосовно складних об'єктів (на прикладі програмування) на основі доктрин когнітивної (пізнавальної) психології. Мається на увазі формування ментальних схем складних об'єктів, пересування цими структурами в процесі діяльності, акумулювання їх в більш узагальнені ментальні утворення.

Пізнавальна (cognitive) психологія є цікавою основою для модельних підходів до розумових процесів. Базисною відправною точкою досліджень в чисельних течіях когнітивної психології (Ф. Бартлетт, С. Палмер, У.Найссер, Э. Рош, М. Минський, Б. Величковський і ін. [1]) є думка, що інтелектуальна діяльність детермінується структурною організацією пізнавальної сфери. Ця ідея є ключовою і у даній роботі. Згідно з ідеями відомого генетичного підходу до психічних структур, жодна з них ніколи не є радикально новою, кожна є модифікацією попередньої в часі структури. Найбільш контрастну оцінку дає У. Найссер. Він вважає, що „ті види інформації, для яких у нас нема схем, ми просто не сприймаємо” [2].

Розглянемо еволюцію ментального відображення модельного процесу з фізики в контексті самостійної практичної реалізації моделі засобами програмування. Проілюструємо, формування ментальної структури на прикладі ізотермічного процесу в ідеальних газах. Зафіксуємо завдання на створення моделі ізотермічного процесу, як початкову проблемну ситуацію, яка спричинює подальші дії.

Рис.1 є платформою для теоретичного розгляду розумових процесів, супутніх самостійному виконанню завдання з практичного програмування. Ментальна частина

структури - когнітивна схема об'єкта, окреслена у великому трикутнику пунктирною лінією. У вершині трикутника - об'єднуючий вузол. Він позначає функції об'єкта як цілісного утворення. Вузли основи позначають відображення в пам'яті реальних неподільних елементів об'єкта - операторів програми. Реальні оператори зображені як рядок тонованих кружечків зовні трикутника.

Всередині великого трикутника є тоновані трикутники меншого розміру. Це трикутники зосередження уваги. Вони окреслюють функціональні вузли, на яких одномоментно сконцентрована увага. У процесі логічних дій поступово відкидаються „зайві” або вводяться „недостаючі” вузли. Межі уваги балансують. Увага завжди концентрується на логічно закінчених структурах.

Початок роботи над створенням навчальної моделі пов'язаний з вибором середовища моделювання і вибором процесу, що буде моделюватись. Модельний процес цілісно, разом з середовищем моделювання позначимо буквою *m*. Майбутній модельний об'єкт-програму, що вирішує проблему моделювання ізотермічного процесу позначимо *m1*. Будемо вважати, що студент має знання з предметної області та з програмування в цілому достатні для проведення робіт.

Нехай, модель реалізується на комп'ютері. Оператори програми, які спільні для всього середовища, позначимо *sr*. Вони майже зразу доступні для реалізації. Уявлення про них існують у суб'єкта навчання.

Деталізація фізичної проблеми відбувається в кілька етапів конкретизації. Перехід з поточного рівня ментальних уявлень на більш конкретизований супроводжується трьома базовими логічними діями: абстрагуванням, аналізом і дедукцією [3, 4]. Ці дії циклічно повторюються при подоланні кожного рівня конкретизації. Під абстрагуванням будемо розуміти нехтування в моделі деталями, які не є суттєвими у роботі моделі. Так, зокрема, на перших етапах конкретизації відбувається абстрагування від одного з вимірів трьохвимірного простору. Моделювання надалі відбувається в площині екрану комп'ютера.

На другому рівні конкретизації реалізується аналіз двовимірної проекції процесу на площині. Аналіз проблеми створення модельного об'єкта зв'язаний з виділенням його взаємодіючих функціональних вузлів. У результаті отримуються три фізичні субстанції: ідеальний газ (*g2*), посудина з поршнем (*s2*), вимірювальні пристрої (*v2*). Подальші дедуктивні дії визначають функції вузлів у процесі. Газ тисне на стінки. Стінки утримують газ. Вимірювальні пристрої визначають тиск при зміні об'єму.

У модельному експерименті зміна тиску спричинюється зміною об'єму. Вимірювач тиску реалізується рахівником, що визначає тиск сукупною кількістю зіткнень атомів зі стінками. Такий візуалізатор при відомій кількості зіткнень реалізується вже на другому рівні конкретизації.

На третьому рівні конкретизації деталізується газ і посудина. Аналіз газу як об'єкту приводить до представлення його сукупністю однакових точкових рухомих атомів (існує розподіл атомів за швидкостями). Наступні дедуктивні дії визначають функції кожного елемента, отриманого в результаті аналізу, в контексті їх сукупності. Обговоримо результати аналітико-дедуктивних дій на поточному рівні конкретизації. Ознака сукупності - тиск. Тиск в моделі визначається кількістю атомів і кількістю зіткнень. Кількість зіткнень при постійній кількості атомів (герметична посудина) та постійній температурі є функцією швидкості окремих атомів. Рухомий атом є функціональним вузлом третього рівня деталізації. Функціональна ознака такого вузла - швидкість. Якщо представити таку ситуацію, що в посудині один атом, то його швидкість визначить тиск на стінки. Подальша деталізація уявлень про рухому частинку, що створює тиск, приводить

до представлення її сукупністю параметрів. Час вільного пробігу до зіткнення з стінкою – t ; швидкість руху – v ; поточні координати – xI, yI ; координати початку вільного пробігу – x, y . Забігаючи вперед можна сказати, що такі параметри повинні бути актуальними для кожного з N атомів газу.

На третьому рівні конкретизації посудина представляється трьома окремими нерухомими стінками і однією рухомою – поршнем. Стінка повинна відбити атом, якщо поточні його координати співпадуть з межами посудини. Модель працює в двовимірному просторі. Тому зручно реалізувати межі посудини наступними умовами відбивання. Ліва стінка $100 > xI$, права стінка $xI > Porshen$, верхня $100 > yI$, нижня $yI > 400$ (розміри посудини – $100, 100, Porshen, 400$).

Суттєвим моментом з психолого-педагогічної точки зору є перехід від деталізації в термінах фізичних об'єктів до деталізації в термінах мови програмування. У цілому, мета всієї попередньої пізнавальної (когнітивної) діяльності полягає у створенні ментальної орієнтовної основи для матеріалізованих дій написання окремих операторів програми. Як вказувалось вище, протокол розумових дій у великій мірі задається можливостями нашої уваги. У колі уваги в кожній окремий момент часу може знаходитись обмежена кількість функціональних вузлів (операторів) об'єкта (програми). Як вже зазначалось, на рис.1 вузли об'єкта, які одночасно знаходяться в полі уваги позначено тонованими трикутниками. Ці трикутники накладаються. Логічний зв'язок між розумовими діями здійснюється через спільні вузли.

Подальша еволюція уявлень про модельний об'єкт пов'язана з узагальненням. Крок узагальнення ментальною структурою об'єкта вверх (перехід від нижнього трикутника зосередження до верхнього) пов'язаний з реалізацією індукції, синтезу, абстрагування [3,4]. Індуктивні дії стосовно вузлів об'єкта в полі уваги формують цілісні уявлення про відповідний фрагмент програми, як про монолітне утворення. Синтез в цьому контексті є завершальною дією. Далі відбувається випробовування фрагмента. Якщо уявлення про відповідний вузол, утворене при конкретизації, співпало з синтетичним і корелює з результатами випробовування, то попередня розумова діяльність вважається успішною, відбувається абстрагування і перехід до вузлів верхнього трикутника зосередження уваги.

Зрозуміло, що розмір фрагментів програми, які випробовуються поступово збільшується впродовж пересування ментальною структурою об'єкта вверх. Перевантаження поля уваги не відбувається через використання синтетичних уявлень поряд з уявленнями про елементарні вузли об'єкта (оператори).

Розглянемо реально видимі навчальні процеси, що відбуваються впродовж створення модельної програми. Реперними подіями будемо вважати випробування програми. Випробувальні запуски фрагментів програми є цікавими тому, що кожний фрагмент програми, що випробовується, повинен бути логічно завершеним і дієздатним (без помилок). Логічна завершеність передбачає визначення локальної проблеми, яку цей фрагмент вирішує, відповідних вхідних і вихідних даних. Спостереження за студентами та їх коментарями впродовж роботи свідчить про те, що перед випробуванням формується узагальнене лаконічне уявлення про фрагмент програми. Будемо називати це уявлення концептом. Відповідне уявлення про конструкцію фрагмента, який може бути оглянутий за один сеанс зосередження уваги, будемо називати конструктором. Графічно формалізовані конструкти зображені на рис.1 в трикутниках зосередження уваги.

Перше випробовування стосується пересування атома прямолінійною траєкторією. Локальною метою тут є реалізація фрагмента програми, що відповідний конструкту з вузлом $g2$ у вершині рис.1. Після отримання рухомого об'єкта студенти переходять, як правило, до випробувань концепцій, пов'язаних з реакцією віртуального рухомого атома

газу на віртуальну границю - стінку посудини. Фактично реалізується спрощений конструкт з вершиною $s2$. Перевірка взаємодії фрагментів програми відповідних конструктам $g2$ і $s2$ приводить до випробування концепції зв'язаної з конструктом $m1$. Спостереження за роботою студентів свідчать про те, що в них існує схильність розв'язувати простіший окремих випадок роботи програми, а потім його ускладнювати. Нижче розглянутий один з початкових варіантів фрагменту програми, що реалізує взаємодії одинарного рухомого об'єкту і вертикальної перешкоди зліва.

```

procedure TForm1.Timer1Timer(Sender: TObject);
begin
  Form1.Refresh;
  With Form1.Canvas do
    begin
      t:=t+1;
      If (100>X1) Then begin j:=3.14-j; x:=x1; y:=y1; t:=1 end;
      x1:=x+Round(cos(j)*t*v);
      y1:=y+Round(sin(j)*t*v);
      Ellipse(x1, y1, x1+20, y1+20 );
    end;
end;
end;

```

Надалі прописуються всі границі посудини. Дієдатність кожної границі перевіряється. Параметри окремого атома ($t, v, x, y, x1, y1$) представляються масивами відповідних параметрів і вводиться додатковий цикл.

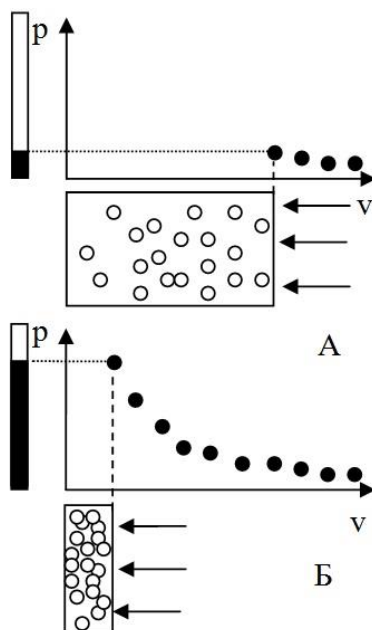


Рис.2 Модельний експеримент. Побудова моделюючою програмою четвертої (А) і десятої (Б) точки ізотерми, відповідно.

Фрагмент програми набуває вигляду, що відповідає конструкту $m1$, який зображений на рис.1. З фрагменту програми видно, що кількість зіткнень фіксується параметром k . В якості візуалізатора використаний компонент ProgressBar. Графік можна вивести за допомогою компонента Chart, для фіксації часу накопичення зіткнень можна використати додатковий компонент Timer.

Розподіл молекул за швидкостями можна грубо змоделювати застосувавши функцію, що генерує випадкові числа. На рис.2 показано інтерфейс описаної вище програми. Нагадаємо, що метою цієї роботи було не створення досконалої моделі, а ілюстрація закономірностей розумових дій впродовж діяльності по самостійному створенню складних модельних об'єктів.

Мета викладача в контексті самостійної практичної роботи полягає в тому, щоб пояснити студентам методологію самостійної роботи.

Студенти ж повинні уникати спорадичних безсистемних дій стосовно програми. Вони повинні уявляти шляхи, якими йде мислення при роботі стосовно складних об'єктів.

Важливим психологічним моментом в роботі з програмами є те, що вони вимагають під час свого створення, відлагодження, модернізації глибокого відстороненого абстрактно-логічного, причинно-наслідкового мислення, яке періодично підкріплюється випробуваннями програм. Сеанс випробовування фрагмента програми має багато спільного з фізичним експериментом.

Звуженість платформи професійних знань, міфологічність мислення, недостатність навичок та умінь, надія на „авось”, на вищі сили є недопустимими і шкідливими при роботі з високотехнологічними об'єктами. Програмування завжди пов'язане з ситуативною неоднаковістю локальних проблем. Вирішення яких спричинює побудову ексклюзивних програмних конструкцій. Така робота формує здатність до креативності розумових дій. У цьому сенсі програмування, як навчальний предмет, формує важливу життєву здатність до строгого причинно-наслідкового мислення, яке поступово вихолощує міфологічну складову.

Становлення спеціаліста з програмування проходить через кілька рівнів розвитку в сенсі структурно-організаційного протоколу розумової роботи. На першому етапі, студент може зв'язати невелику кількість операторів, функції яких одночасно можуть утримуватись ним у полі уваги. На другому етапі формується вміння переносити увагу між фрагментами програми, що знаходяться на одному, або на суміжних рівнях програмної ієрархії. Третій етап передбачає вміння робити далекі транзитні переходи структурою об'єкта і дає можливість створювати і відлагоджувати програмні структури високої складності і великого розміру. Спостереження за великою кількістю студентів, що вивчають програмування приводить до думки, що вміння програмувати формується на протязі років і має системний послідовний характер. Це спосіб мислення. Тут як ні в жодній іншій діяльності діє дещо радикальна але очевидно вірна теза, навчити неможливо можливо тільки навчитись.

Література

1. Холодная М.А. Психология интеллекта: парадоксы исследования. — СПб.: Питер, 2002. - 272с.
2. Найссер У. Познание и реальность. Смысл и принципы когнитивной психологии. М.: Прогрес, 1981. - 225с.
3. Головін М.Б. Вплив когнітивних процесів супутніх діяльності на розподіл швидкостей виконання завдань з програмування. // Проблеми педагогічних технологій. Збірник наукових праць. Випуск 1-4. Луцьк, 2007. – С.160-172.
4. Головін М.Б. Специфіка формування пояснень до прикладів – апікацій у навчальному матеріалі з програмування //Проблеми сучасного підручника, Збірник наукових праць. Випуск 7. Київ-Луцьк, 2007. – С.339-350.