

МОДЕЛЬНИЙ РОЗГЛЯД ПІЗНАВАЛЬНИХ ПРОЦЕСІВ, СУПУТНИХ НАВЧАЛЬНОМУ ПРОГРАМУВАННЮ

Микола Головін, Ніна Головіна, Неля Головіна

Східноєвропейський національний університет імені Лесі Українки, м. Луцьк, Україна,
NickHolovin@gmail.com

У статті розглянуто авторську оригінальну психологічну модель ментальних пізнавальних процесів, що забезпечують навчальне програмування. Уявлення про програму, що формуються в процесі навчальних дій, подано у вигляді ієрархічної структури. Вона розростається невеликими, логічно завершеними фрагментами. Модельний механізм формування цієї структури в процесі навчання окреслено лаконічною схемою, у якій задіяно три циклічні процеси. Їх ментальний механізм пояснено термінами формальної логіки. Представлена модель формування знань є універсальною. Вона може бути платформою для розробки методик навчання в будь-якій галузі знань. Навчальне програмування вибрано для ілюстрації моделі, тому що ця сфера знань має витончену ієрархічну структуру, елементи якої є кроками алгоритму, що можуть бути швидко перевірені на дієздатність. У цьому проявляється спорідненість цих пізнавальних схем із когнітивними картами. Це особливо добре помітно під час моделювання фізичних процесів, що мають візуальний складник свого розвитку на екрані. Саме тому приклади з програмування простих фізичних процесів розглянуто в цій роботі. Цікаві ментальні процеси, що супроводжують ускладнення програми в процесі її написання, тобто переходи від простішої до складнішої моделі фізичного процесу. Програмний фрагмент, у якому уособлюється ускладнення моделі фізичного процесу, логічно завершений. Він може бути охоплений полем уваги та усвідомлений. На прикладі ускладнення моделі фізичного процесу розглянуто механізми ускладнення структури уявлень про програму.

Ключові слова: пізнавальні процеси, психологічна модель, довготривала пам'ять, короткочасна пам'ять, увага, свідомість, цикли пізнавальних дій, формальна логіка, структура декларативних знань, пізнавальні структури та схеми, методика викладання інформатики.

Mykola Holovin, Nina Holovina, Nelia Holovina. Computer Modeling of Physical Processes and Specifics of Relevant Educational Activities. The article discusses the unique and original psychological model of mental cognitive processes that serve as a foundation for the educational programming. The understanding of the program that is formed in the process of educational action is presented in the form of hierarchical structure. This structure develops through the small, logically finished fragments. Model mechanism of formation of this structure in the process of

education is depicted through laconic scheme of three cyclical processes. Mental mechanism of these cycles is explained through the terms of formal logics. The presented model of knowledge formation is universal. It can be used as a platform for development of educational methodology in any sphere of science. The educational programming was chosen for illustration of the model, as this field of science has elaborated hierarchical structure and elements of this structure are the elements of algorithm that can be fast checked for efficiency. The interrelatedness of these schemes is reflected particularly well while modeling the physical processes that have visual component of own development on the screen. All the examples of programming of simple physical processes are discussed within this work.

Key words: cognitive processes, psychological model, long-term memory, short-term memory, attention, consciousness, cycles of cognitive actions, formal logics, structure of declarative knowledge, cognitive structures and schemes, methodic of informatics teaching.

Николай Головин, Нина Головина, Неля Головина. Модельное рассмотрение познавательных процессов, сопутствующих учебному программированию. В статье рассматривается авторская оригинальная психологическая модель ментальных познавательных процессов, обеспечивающих учебное программирование. Представления о программе, которые формируются в процессе учебных действий, подаются в виде иерархической структуры. Она разрастается небольшими, логично завершенными фрагментами. Модельный механизм формирования этой структуры в процессе обучения очерчен лаконичной схемой, в которой задействованы три циклические процесса. Ментальный механизм этих циклов объяснен терминами формальной логики. Представленная модель формирования знаний является универсальной. Она может быть платформой для разработки методик обучения в любой области знаний. Учебное программирование выбрано для иллюстрации модели потому, что эта область знаний имеет изящную иерархическую структуру, элементы которой являются шагами алгоритма, что могут быть быстро проверены на работоспособность. В этом проявляется родство таких познавательных схем с когнитивными картами. Это особенно хорошо видно при моделировании физических процессов, имеющих визуальную составляющую своего развития на экране. Именно поэтому примеры по программированию простых физических процессов рассмотрены в этой работе. Интересны ментальные процессы, сопровождающие осложнения программы в процессе ее написания, то есть переходы от простой к сложной модели физического процесса. Программный фрагмент, в котором олицетворяется усложнения модели физического процесса, логически завершен. Он может быть охвачен полем внимания и осознан. На примере усложнения модели физического процесса рассматриваются механизмы усложнения структуры представлений о программе.

Ключевые слова: познавательные процессы, психологическая модель, долговременная память, кратковременная память, внимание, сознание, циклы познавательных действий, формальная логика, структура декларативных знаний, познавательные структуры и схемы, методика преподавания информатики.

Постановка наукової проблеми та її значення

Навчальне програмування як діяльність вимагає витонченого причинно-наслідкового та абстрактно-логічного мислення. Програми як об'єкти ментальної пізнавальної й матеріалізованої діяльності є складно організованими, добре структурованими. Як на локальному рівні, так і на рівні окремих укрупнених функціональних блоків, програми мають численні логічні внутрішні та зовнішні зв'язки. Цікавою галуззю програмування є комп'ютерне моделювання фізичних процесів. Програмування в галузі фізики накладає на модель строгі рамки реального процесу, сприяє розвитку дослідницьких умінь і навичок, наближає процес навчання до наукового пошуку. Моделювання фізичних процесів у сенсі необхідних для цього знань вимагає трьох важливих компонентів (програмування, математику та фізику).

Навчальна діяльність, що включає практичне програмування, специфічна. Тут контрастно проявляються особливості ментальних процесів людини, способи мислення, формування знань. Цьому сприяє висока формалізація знань та одночасно висока структурність і зв'язність в усіх трьох згаданих вище напрямках (інформатика, математика, фізика). Висока контрастність предмета вивчення дає можливість максимально формалізувати сам процес навчання та оптимізувати його. Актуальним напрямом покращення методів навчання є дослідження процесу навчання на базисі когнітивної психології, де ментальні пізнавальні процеси, притаманні людині, розглядаються на детальному рівні.

Важливими для навчання є такі концепції когнітивної психології:

– жодна пізнавальна схема людини не є абсолютно новою, кожна є модифікацією або трансформацією попередніх ментальних схем (Холодная, 2002);

– схема усвідомлюється та модифікується маленькими порціями, що корелюють із розміром короткочасної пам'яті людини, поля уваги (7 ± 2) (Miller, 1956);

– розвиток пізнавальних структур відбувається в напрямі їх деталізації (Чуприкова, 1997).

Мета цієї роботи – модельне дослідження трансформацій пізнавальних схем упродовж навчального програмування з метою покращення процесів навчання.

Дослідження трансформацій пізнавальних схем у сфері програмування та фізики дає можливість екстраполювати виявлені закономірності в інші області, де структура знань не має такої строгої конструкції

й ментальні процеси не проявляються контрастно. У цьому сенсі комп'ютерне моделювання виступає як полігон для досліджень методик навчання.

Виклад основного матеріалу й обґрунтування отриманих результатів дослідження

Комп'ютерна програма та її структура

Розглянемо особливості формування структури знань на прикладі виконання завдання, що містить моделювання простого фізичного процесу (рух тіла в полі тяжіння та відбивання його від перешкоди), мовою Java Script (JS). Текст відповідної готової програми, що є результатом виконання завдання, відображено на рис. 1.

Текст програми структурований. Структура реалізована виділенням блоків, логічно завершених фрагментів у тексті програми. Виділення має вигляд зміщення праворуч блоку від рядка, що ним керує, та вирівнюванням самого блоку по лівому краю. Таким чином формується ієрархічна деревоподібна конструкція, що може мати багато рівнів укладеності. Адже в кожний блок у тілі програми може бути вкладено інший блок або кілька.

Запуск моделі, зображеної на рис. 1 можна реалізувати рядком:

```
<body onload="Model()"> <div id="Tilo"> </div> <div id="rectangle"></div> </body>
```

```
<style type="text/css">
  #Tilo{position:absolute;width: 50px;height: 50px;background: red;
  -moz-border-radius: 50px;-webkit-border-radius: 50px;border-radius:50px;}
  #rectangle {position:absolute; width: 700px;height: 500px; border: 2px solid grey;}
</style>
<script type="text/javascript">
  function RuhTila(X0, Y0, D, kulja) // Рух тіла
  {
    var T=0; Vo=0;Ypoch=100; Y=Y0;
    kolo = document.getElementById(kulja)
    setInterval(function() // Циклічне обчислення координати тіла та його візуалізація
    {
      if (Y>500) // Відбивання тіла від перешкоди вниз
      {Vo=T*9.8; T=0; Ypoch=500}
      Y=Ypoch-Math.round(Vo*T)+Math.round(9.8*Math.pow(T, 2)/2)
      T=T+0.1; kolo.style.left=X0; kolo.style.top=Y
    }, 40);
  }
  function Model()
  {
    RuhTila(300, 300, 50,'Tilo')
    rectangle.style.left=50; rectangle.style.top=50;
  }
</script>
```

Рис. 1. Програма мовою Java Script, що моделює рух тіла в полі тяжіння та відбивання його від перешкоди

Пізнавальна схема програми, як відображення її структури

Пізнавальна структура ключового фрагмента програми – функції `RuhTila()` (рис. 1) у завершеному вигляді після виконання завдання вкладається в чотири конструкти (рис. 2.8).

Під конструктом розумітимемо логічно завершений фрагмент програми, який можна охопити полем уваги та усвідомити одномоментно. Функції такого фрагмента можуть бути лаконічно сформульовані. Конструкт у контексті пізнавальної схеми означає логічно завершену частину цілісної пізнавальної конструкції, що має свій функціональний сенс у цілому. Сукупність конструктів програми зв'язана в ієрархічну структуру. Зсув блоку тексту програми праворуч униз показує, що відмічений таким чином текст перебуває під управлінням рядка зверху ліворуч. Програмні блоки, відповідні конструктам, мають логічну завершеність і кількість компонентів у них укладається в магічне число Міллера 7 ± 2 (Miller, 1956). Якщо ж у блоці більше компонентів, то він, як правило, у ментальній роботі учня представляється, як кілька конструктів, розміщених на одному рівні вкладеності. Поточне поле уваги учня позначено на рисунку виділеним трикутником (рис. 2).

Відповідність між текстом програми на рис. 1 та пізнавальною схемою на рис. 2.8 така. На рис. 2.8 кружечками позначені окремі компоненти пізнавальної схеми. Компонентами першого типу можуть бути конкретні засоби програмування, зокрема такі, як оператори, функції, процедури. Другий тип компонентів – це синтетичні програмні блоки, що включають багато конкретних компонентів.

Конструкт що міститься на першому рівні, має шість компонентів, на другому – п'ять, на третьому – два конструкти по три компоненти. Перший рівень: `T=0; Vo=0; Ypoch=100; Y=Y0; kolo = document.getElementById (kulja); set Interval (function ()`. Другий рівень: `if (Y>500)....; Y=Ypoch...; T=T+0.1; kolo.style.left=X0; kolo.style.top=Y`. Третій рівень має два конструкти по три компоненти відповідно: `Y=Ypoch-Math.round (Vo*T)+Math.round (9.8*Math.pow (T, 2)/2)` та `Vo=T*9.8; T=0; Ypoch=500`.

Компонент конструкту у вершині трикутника називатимемо його концептом. Він цілісно, концептуально описує функцію всього конструкту. Зрозуміло, що концепт завжди має більш узагальнений сенс, порівняно з конкретизованими компонентами, якими він реалізується. Концепт, що входить у склад двох конструктів на стороні верхнього конструкту, розуміється цілісно, а на стороні нижнього – диференційовано. Перехід від концепту до компонентів у конструкті

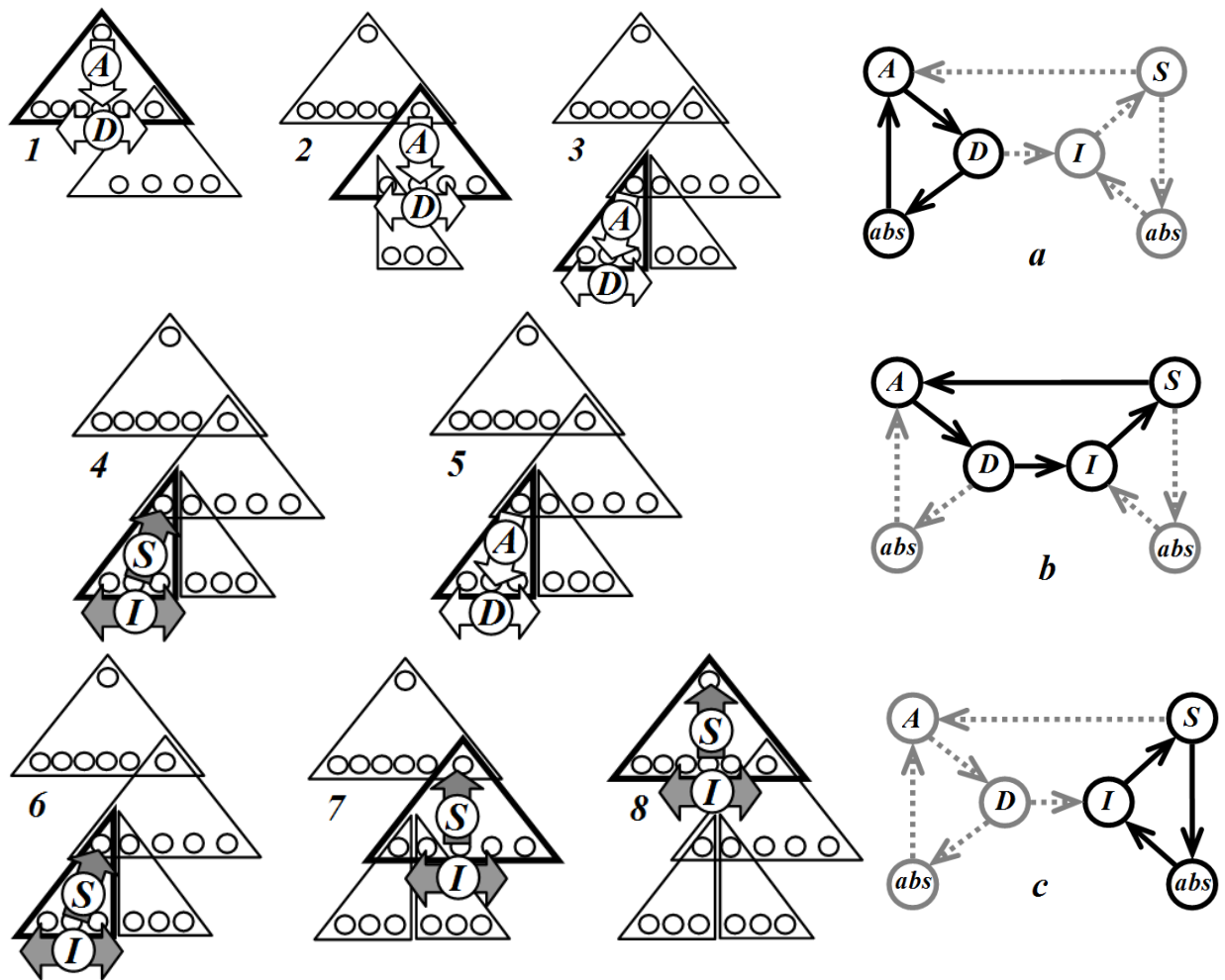


Рис. 2. Модельний механізм ментальних пізнавальних процесів людини під час виконання завдання на програмну реалізацію відбивання від перешкоди тіла під час руху його в полі тяжіння. Ліворуч – еволюція пізнавальних схем (від першої до восьмої) у довготривалій пам'яті в процесі навчальних дій. Праворуч – цикли навчальних дій (a, b, c) в термінах формальної логіки, що спричиняють цю еволюцію схем. Стрілками чорного кольору позначено домінуючі дії в циклах, а сірими – другорядні. Позначення на схемах: A – аналіз; D – дедукція; I – індукція; S – синтез; abs – абстрагування.

здійснюється в результаті дедуктивних дій, а зворотний перехід вимагає індуктивної діяльності. Переключення уваги через концепт від верхнього до нижнього конструкту – конкретизація, а від нижнього до верхнього – узагальнення. В обох випадках відбувається абстрагування. Переключення уваги є елементарним кроком в абстрактно-логічних діях. Саме через концепти відбуваються абстрактно-логічні транзити уваги: униз – конкретизація, уверх – узагальнення. Зрозуміло, що транзит уваги без утрати логіки думок може відбуватися не лише між

двома сусідніми конструктами, але й між кількома суміжними. Такі транзити уваги детально розглянуто в роботі (Головін, 2008).

Технологічно важливим для методики викладання є контроль переміщення поля уваги учнів у процесі пояснення нового матеріалу або за підтримки їхніх дій на лабораторному занятті. Мінімальний крок у процесі переміщення уваги є переведення зосередження уваги на суміжний із поточним конструкт через спільний для цих конструктів компонент – концепт.

Еволюція пізнавальної схеми в процесі виконання завдання проходить проміжні стани від рис. 2.1 (статична куляка на екрані) до рис. 2.8 (рух у полі тяжіння та відбивання). Однак уважатимемо, що опорною програмою є програма, яка моделює руху тіла в полі тяжіння (рис. 2.2). Пізнавальну схему цієї опорної програми потрібно сформулювати в процесі усвідомлення завдання. Завдання ж полягає в програмній реалізації відбивання (рис. 2.8).

Пізнавальна схема опорної програми (рис. 2.2) має міститися в довготривалій декларативній пам'яті суб'єкта навчання (учня або студента) на момент початку активних дій із написання нового програмного коду. Дії, зображені на рис. 2.1.–2.2. та на рис. 2а, є діями з освоєння опорної програми.

Ця опорна програма представлена в довготривалій пам'яті трьома конструктами, укладеними один в одній. Верхній конструкт охоплює всю програму. Концепт цього конструкту лаконічно може бути позначений у мисленні, наприклад, словами «рух тіла». У тексті програми (рис. 1) це відповідний такому конструкту програмний блок, що позначається «Ruh Tila(...)».

Конструкт, розміщений нижче, відповідає за циклічне обчислення координати тіла й виведення його на екран. Скорочено у внутрішній мові суб'єкта мислення відповідний концепт може позначатися, наприклад, як «цикл руху». У програмі він позначений «setInterval (...)».

Найнижчий конструкт схеми опорної програми включає компоненти формули, що відповідальні за обчислення координат тіла при рівноприскореному русі. Концепт, що об'єднує у внутрішній мові ці компоненти, може звучати як «координати тіла». У програмі він має вигляд:

$$Y=Y_{\text{poch}}-\text{Math.round}(V_0*T)+\text{Math.round}(9.8*\text{Math.pow}(T, 2)/2).$$

Написання цієї формули вимагає окремого зосередження уваги. Тому в структурі функції RuhTila() формула представляється окремим конструктом.

Будемо вважати, що конструкт, відповідний формулі, складається з трьох компонентів, хоча кількість компонентів мислення може бути індивідуальна для учнів із різною підготовкою. Деякі школярі із низьким рівнем підготовки можуть сприймати цю конструкцію як таку, що має до дев'яти компонентів.

Суть виконання завдання у формалізмі пізнавальних схем полягає у формуванні в опорній пізнавальній схемі людини ще одного (четвертого) конструкту. Він відповідає реалізації однократного повного пружного відбивання кульки. Відбивання вимагає написання розгалуження:

$$\text{if } (Y > 500 - D) \{ V_0 = T * 9.8; T = 0; Y_{\text{roch}} = 500 \},$$

де 500 – числове значення, що вказує на рівень перешкоди, від якої здійснюється відбивання тіла (кульки) без урахування її розмірів.

Концепт цього нового конструкту може в думках учня мати одне з таких лаконічних вербальних позначень: «перешкода», «відбивання», «земля». У програмі цей концепт видимий, як «if(...)». У конструкті концепт конкретизується у вигляді трьох компонентів, що розміщені за розгалуженням. Вони спрацьовують за виконання умови.

Примітка. Програма, представлена вище, реалізовує рівноприскорений рух, однократне відбивання та рівносповільнений рух. Багатократне відбивання кульки вимагає ще одного розгалуження, що забезпечує правильний перерахунок змінних при повороті у верхній точці. Воно реалізується $\text{if } (Y < 100) \{ V_0 = 0; T = 0; Y_{\text{roch}} = 100 \}$

Модельний механізм пізнавальних процесів у динаміці. Розглянемо крізь призму моделі ментальну роботу людини, що приводить до появи в її пізнавальній схемі нового конструкту. Як зазначено вище, цей конструкт відповідальний за програмний блок, що реалізовує відбиття тіла від перешкоди. Нехай виконання завдання починається на початку заняття, а програма, що моделює рух тіла в полі тяжіння, дається учню в цілісному вигляді, як опорна. Припустимо також, що школяр розв'язував уже задачі, подібні до опорної. Пізнавальну схему до рівня, зображеного на рис. 2.2 (рух тіла в полі тяжіння), потрібно модифікувати на основі наявних у довготривалій пам'яті пізнавальних схем й актуалізувати.

Актуалізація пізнавальної схеми опорної програми полягає в аналітико-дедуктивних діях, прикладених до тексту цієї опорної програми. Дії відбуваються за схемою рис. 2а. Пізнавальна схема опорної програми в цьому конкретному випадку формується двома етапами, зобра-

женими на рис. 2.1 та 2.2. Специфіка кожного з них розкривається схемою перцептивного циклу Найссера (рис. 3) й полягає в такому (Найссер, 1981).



Рис. 3. Схема перцептивного циклу Найссера

Актуалізація опорної програми починається з **дослідження** її тексту. Ґрунтується це на вже наявних у довготривалій пам'яті учня **когнітивних (пізнавальних) схемах** та уявленнях про функції програми. Ці схеми спрямовують дослідження. Учень може **вибрати** в тексті опорної програми не більше ніж 7 ± 2 зв'язані між собою програмні компоненти (програмних блоків, окремих операторів, функцій, процедур). Далі дедуктивними діями він зв'язує частини в цілісний механізм, з'ясовуючи при цьому призначення кожного з компонентів. У цьому полягає окремий крок у **модифікації** пізнавальної схеми опорної програми та її актуалізації. Перший оберт за схемою на рис. 3 відповідає усвідомленню верхнього конструкту в пізнавальній схемі (рис. 2.1). Увагу зосереджено на верхньому конструкті. Зрозуміло, що модифікована пізнавальна схема (рис. 2.1) спрямовує дослідження на нову групу зв'язаних компонентів програми. Другий оберт на рис. 3 відповідає усвідомленню конструкту, розміщеного нижче на один щабель у пізнавальній схемі (рис. 2.2). Відбувається абстрагування від компонентів верхнього конструкту та **вибір** нових компонентів у тексті програми, що відповідні конструкту, розміщеному на один щабель нижче (рис. 2.2). Абстрагування та подальші аналітико-дедуктивні дії учня (рис. 2а) зв'язують частини програми, відповідні цьому нижчому конструкту, у цілісний механізм, з'ясовуючи при цьому призначення кожного з компонентів. Відбувається подальша

актуалізація та **модифікація** пізнавальної **схеми**. Тепер уже два зв'язані конструкти включені в схему й містяться в довготривалій пам'яті. Нова пізнавальна схема **спрямовує** зусилля на виконання завдання – написання нового фрагмента коду програми. Кожен новий оберт схемою, представленою на рис. 3, спрацьовує в режимі передбачення результатів вибору.

Потрібно відзначити, що на рис. 2 не показано, що учень детально розбирається з конструктом, відповідальним за обчислення координат тіла під час руху в полі тяжіння. Стосовно конструкту на найнижчому щаблі (рис. 2.2) відсутні аналітико-дедуктивні дії. У цих діях нема потреби. Адже для реалізації відбивання достатньо бачити блок, що відповідає за обчислення координат, цілісно та записати поряд із ним на одному рівні вкладеності програмний блок, відповідальний за відбивання.

Завершуючи обговорення ментальних дій, пов'язаних з актуалізацією опорних пізнавальних схем, потрібно зазначити таке. Якщо пізнавальна схема опорної програми вже міститься в декларативній пам'яті учня, то він у процесі перегляду її тексту тільки актуалізує вже наявну пізнавальну схему. Якщо ж учень дивиться на текст опорної програми, а відповідної пізнавальної схеми ще нема, то в процесі вивчення тексту опорної програми він модифікує одну з підходящих наявних пізнавальних схем для поточної проблемної ситуації. Надалі ця модифікована та актуалізована пізнавальна схема є базисом для розв'язування поставленої задачі.

Модифікація пізнавальної схеми при генерації нового конструкту
На рис. 2.3 представлено аналітико-дедуктивні дії в ментальній діяльності людини, які стосуються **генерації конструкту** як групи логічно зв'язаних нових (таких, що ще не існують в опорній програмі) компонентів програми. Якісна відмінність цього кроку від кроків на рис. 2.1 та рис. 2.2 полягає в тому, що тут відбувається не дослідження наявних програмних механізмів, а їх утворення. Утворення нового фрагмента програми супроводжується виникненням нового конструкту (ментальних уявлень про новий фрагмент програми). Процес утворення ґрунтується, напередбаченні успішного результату застосування однієї чи кількох модифікацій стереотипних пізнавальних схем, що були використані суб'єктом навчання в подібних проблемних ситуаціях колись раніше.

Із рис. 2.1–2.3 видно, що людині у своїх ментальних діях потрібно триразово повторити цикл аналітико-дедуктивних дій, щоб виконати завдання. Цей цикл можна зобразити схемою, як на рис. 2а. Видно, що

кожний виток цього циклу супроводжується абстрагуванням. У процесі абстрагування відбувається неповна очистка поля уваги (свідомості) від компонентів попереднього конструкту. У свідомості ланцюг роздумів підтримується через низку концептів.

На рис. 2.4 зображено синтез нового фрагмента програми, що відбувається в контексті індуктивних дій людини стосовно тексту нового програмного блоку. Спрацьовує цикл, який включає у свій склад абстрагування, синтез, індукцію, зображений на схемі рис. 2 с. У діях, показаних на рис. 2.3; 2.4, ментальні дії супроводжуються матеріалізованими діями. Матеріалізовані дії полягають у написанні нового програмного фрагмента та перевірці виправленої програми на дієздатність. Якщо перевірка роботи блоку виявила правильну роботу фрагмента й усієї програми в цілому, то відбувається завершення дій стосовно виконання завдання. Якщо ж у процесі перевірки з'ясувалося, що фрагмент працює неправильно, то далі виконуються повторні аналітико-дедуктивні (рис. 2.4) та індуктивно-синтетичні дії (рис. 2.5) без зміни поля уваги. Ці дії з виправлення програмного фрагмента можуть повторюватися багатократно за схемою рис. 2b (Головін, 2008: 312), (Головін, 2008: 7). У цій діяльності ментальні дії також супроводжуються матеріалізованими діями, які включають написання нового програмного фрагмента, випробовування його в роботі. Так формується нова реалізація програмного фрагмента.

Якщо щойно написаний фрагмент вимагає змін у тілі програми не лише на рівні щойно програмно реалізованого конструкту, тоді може спрацювати довгий цикл дій, що вимагає багатократної зміни поля уваги. Послідовність дій у довгому циклі включає кроки вверх конструктами (рис. 2.6, рис. 2.7, рис. 2.8) за схемою рис. 2с, а потім цими ж конструктами униз за схемою рис. 2а. Такі дії можуть уключати локальні зацикльовки дій без зміни поля уваги на окремому конструкті за схемою рис. 2b.

Представлена модель формування знань є універсальною. Вона може бути платформою для розробки методик навчання в будь-яких галузях знань, у тому числі гуманітарних. Навчальне програмування вибрано для ілюстрації моделі тому, що ця сфера знань має витончену ієрархічну структуру та елементи цієї структури є кроками алгоритму, які можуть бути швидко перевірені на дієздатність. У цьому проявляється спорідненість цих пізнавальних схем із когнітивними картами. Така спорідненість особливо добре проявляється під час моделювання

фізичних процесів, що мають візуальну складову свого розвитку на екрані. Саме тому приклади з програмування простих фізичних процесів розглянуто в цій роботі.

Перевірку дієздатності цієї моделі, що ґрунтується на аналізі статистичних розподілів швидкостей виконання завдань з інформатики великими групами студентів, представлено у роботах (Головін, 2011), (Головін, 2015). Засвоєння фактичного навчального матеріалу, що розглянутий у цих роботах, забезпечувалося виконанням пакетів завдань. Розв'язання кожного з них реалізовувалося побудовою невеликої ієрархічної конструкції, що калібрована за кількістю елементів та складністю. Завдання на програмування вимагають додання більших за розміром і складніших за змістом ієрархічних структур. Аналіз статистичних розподілів швидкостей навчальних дій під час виконання завдань на програмування є найближчою метою авторів роботи.

Висновки й перспективи подальших досліджень

Розкрито особливості ментальних та матеріалізованих дій людини в термінах формальної логіки. Проведено аналіз трансформації пізнавальних схем у процесі навчання.

Навчальні дії можуть бути добре керовані в кожний окремий поточний проміжок часу, якщо вони стосуються невеликих упорядкованих, логічно завершених фрагментів пізнавальної схеми об'єкта вивчення, кожен із яких може одномоментно бути охоплений полем уваги суб'єкта навчання.

Знання представленого лаконічного модельного механізму ментальних пізнавальних процесів людини викладачем та його врахування в процесі лекційного розтлумачення аплікаційних програм або в процесі підтримки учнів на лабораторних заняттях із програмування дає змогу уникнути хаотичності, надлишковості або недостатності навчальних дій учителя.

Література

1. Холодная М. А. Психология интеллекта: парадоксы исследования / М. А. Холодная. – Санкт-Петербург : Питер, 2002. – 272 с.
2. Miller George A. The Magical Number Seven, Plus or Minus Two / A. Miller George // The Psychological Review. – 1956. – Vol. 63. – Issue 2. – P. 81–97.
3. Чуприкова Н. И. Психология умственного развития: принцип дифференциации / Н. И. Чуприкова. – Москва : Столетие, 1997. – 478 с.

4. Головін М. Б. Зміст підготовки висококваліфікованого фахівця з інформаційних комп'ютерних технологій у контексті когнітивних процесів (на прикладі програмування) / М. Б. Головін // Інформаційні технології в освіті. – Вип. 2. – Херсон, 2008. – С. 66–73. URL: <http://ite.kspu.edu/issue-2/p-66-73>

5. Найссер У. Познание и реальность. Смысл и принципы когнитивной психологии / У. Найссер. – Москва : Прогрес, 1981. – 225 с.

6. Головін М. Б. Практична навчальна діяльність з програмування в контексті теорії про поетапне формування розумових дій та понять / М. Б. Головін // Сучасні інформаційні технології та інноваційні методики навчання в підготовці фахівців: методологія, теорія, досвід, проблеми. – 2008. – Вип. 20. – С. 312–317. URL: <http://esnuir.eenu.edu.ua/bitstream/123456789/12915>

7. Головін М. Б. Розумові та матеріалізовані дії, їх вербальне супроводження в контексті практичного навчального програмування / М. Б. Головін // Вісник Черкаського університету. – 2008. – Вип. 136 : Педагогічні науки. – С. 7–13. URL: <http://esnuir.eenu.edu.ua/bitstream/123456789/12867>

8. Головін М. Б. Дослідження процесів навчання на основі аналізу моментів статистичних розподілів швидкостей навчальних дій (на матеріалах вивчення інформатики) / М. Б. Головін // Психологічні перспективи. – Вип. 18. – Луцьк, 2011. – С. 62–72. URL: <http://esnuir.eenu.edu.ua/bitstream/123456789/1743/1/Holovin.pdf>

9. Головін М. Б. Формування знань з інформатики в процесі конструювання ієрархічних схем та статистичний аналіз цих навчальних дій / М. Б. Головін // Науковий часопис НПУ імені М. П. Драгоманова. – Серія № 2. Комп'ютерно-орієнтовані системи навчання – № 16 (23). – 2015. – С. 20–26. URL: <http://enpuir.npu.edu.ua/handle/123456789/10653>

References

1. Holodnaya M. A. (2002). *Psihologiya intellekta: paradoksy issledovaniya*. [Psychology of Intellect: paradoxes of research.] SPb: Piter. 272 p. [in Ukrainian].

2. Miller George A. (1956). The Magical Number Seven, Plus or Minus Two // *The Psychological Review*, 63, Issue 2, 81–97.

3. Chuprikova N. I. (1997). *Psihologiya umstvennogo razvitiya: Printsyp differentsiatsii*. [Psychology of cognitive development: Principle of differentiation] М.: Stoletie. 478 p. [in Russian].

4. Holovin N. B. (2008). Zmist pidgotovky vysokokvalifikovanogo fahivtsya z informatsijnyh kompyuternyh tehnologij u konteksti kognityvnyh protsesiv (na prykladi programuvannya). [Preparation of a highly qualified specialist in information computer technologies in the context of cognitive processes (ex. programming)]. *Informatsijni tehnologii v osviti – Informational technologies in education*, 2, 66–73. <http://ite.kspu.edu/issue-2/p-66-73>. [in Ukrainian].

5. Naysser U. *Poznanie i real'nost'*. *Smysl i printsipy kognitivnoi psihologii*. [Cognitive perception and reality. Sense and principles of cognitive psychology]. М.: Progres. 225 p. [in Russian].

6. Holovin N. B. (2008). Praktychna navchal'na diyal'nist' z programuvannya v konteksti teotii pro poetapne formuvannya rozumovyh dij ta ponyat'. [Practical

educational activity in programming in the context of theory of stage formation of cognitive activities and notions]. *Suchasni informatsijni tehnologii ta innovatsijni metodyky navchannya v pidgotovtsi fahivtsiv: metodologiya, teoriya, dosvid, problemy – Modern informational technologies and innovational methodic of teaching in preparation of professionals: methodology, theory, experience, problems*. 20. 312–317. <http://esnuir.eenu.edu.ua/bitstream/123456789/12915> [in Ukrainian].

7. Holovin N. B. (2008). Rozumovi ta materializovani dii, ih verbal'ne suprovodzhennya v kontexti praktychnogo navchal'nogo programuvannya. [Cognitive and materialized activities, their verbal accompaniment in the context of practical educational programming.] *Visnyk cherkas'kogo universytetu – Cherkassy University magazine. Pedagogical sciences*. 136. Pedagogichni nauky. 7–13. <http://esnuir.eenu.edu.ua/bitstream/123456789/12867> [in Ukrainian].

8. Holovin N. B. (2011). Doslidzhennya protsesiv navchannya na osnovi analizu momentiv statystychnyh rozpodiliv shvydkostei navchal'nyh dij (na materialah vyvchennya informatyky). [Investigation of learning process based on analyse of statistic moment velocity distributions of learning actions (on the materials of computer science learning). *Psychologichni perspektyvy – Psychological perspectives*, 18. 62–72. <http://esnuir.eenu.edu.ua/bitstream/123456789/1743/1/Holovin.pdf>. [in Ukrainian].

9. Holovin N. B. (2015). Formuvannya znan' z informatyky v protsesi konstruyuvannya ierarhichnyh shem ta statystychnyi analiz tsyh navchal'nyh dij. [Formation of knowledge of informatics in the process of construction of hierarchical schemes and the statistical analysis of these educational activity] *Naukovyi chasopys NPU imeni Dragomanova – M. P. Dragomanov NPU Scientific magazine, 2. Kompyuterno orientovani systemy navchannya – Computer Oriented systems of education № 16 (23)*. 20–26. <http://enpuir.npu.edu.ua/handle/123456789/10653> [in Ukrainian].

Received: 24.04.2018

Accepted: 16.05.2018