

Психолого-педагогічні аспекти навчання основ програмування

Одним з основних завдань освітньої системи є виховання творчої людини. Для цього необхідно впроваджувати такі форми організації навчальної діяльності, які сприяють різнобічному розвитку кожного студента. З метою розв'язування цього важливого завдання психологи й дидакти досліджують загальні психологічні закономірності розумового процесу (О.М. Леонтьєв [1]); структуру мислення (Л.С. Виготський [2], Г.С. Костюк [3], С.Л. Рубінштейн [4] та ін.); системи розумових дій, що складають основу мислення (П.Я. Гальперін [5], Є.М. Кабанова-Меллер [6], Н.Ф. Тализіна [7] та ін.).

У роботах з психології ([4], [8-9] та ін.) розвиток студента розглядається як процес, у якому поєднуються біологічні, фізіологічні, психічні і соціальні аспекти. Результатом є формування індивідуальності, суттю якої є єдність і взаємозв'язок властивостей особистості людини й суб'єкта діяльності.

Дослідження А.М. Алексюка [10], Б.Г. Ананьєва [11], М.В. Буланової-Топоркової, А.В. Духавневої, С.І. Самигіна [12] та ін. присвячені вивченню особливостей розвитку особистості студента. Для студента, як людини, характерні психічні властивості індивіда, від яких залежить перебіг психічних процесів, виникнення психічних станів; суспільні відносини, зумовлені належністю до певної соціальної групи; якості, які зумовлені спадковістю і вродженими задатками: тип вищої нервової діяльності, безумовні рефлекси, інстинкти, фізична сила і т.п.; щодо морального розвитку у цьому віці, то він характеризується з одного боку, посиленням свідомих мотивів поведінки, закріпленням таких якостей, як цілеспрямованість, рішучість, наполегливість, самостійність, ініціатива, вміння володіти собою; підвищенням інтересу до моральних проблем (мети і образу життя, обов'язку тощо), а з іншого боку, недорозвиненість свідомої регуляції поведінки, наявність внутрішньої невпевненості у собі; щодо діяльності студента, то вона вимагає інтенсивності функціонування психіки, високої інтелектуальної напруженості, перевантажень [12].

Ці самі дослідники виділяють три основні підходи до пізнавальної діяльності студентів:

1) широкий підхід до цілей і завдань навчання, тобто орієнтація на широку спеціалізацію, на різнобічну професійну підготовку, коли інтереси студентів спрямовані на більш широке поле знань, ніж передбачено програмою;

2) чітка орієнтація на вузьку спеціалізацію, при якій пізнавальна діяльність студентів характеризується виходом за рамки навчальної програми в межах окремих дисциплін;

3) засвоєння знань і здобуття вмінь лише в межах навчальної програми.

Психологи розрізняють два основних види діяльності, пов'язані з пізнавальними процесами людини – учіння і навчання. Згідно з [13], «учіння – цілеспрямований процес засвоєння учнями знань, оволодіння уміннями й навичками», «навчання – цілеспрямований процес формування і засвоєння знань, умінь, навичок і способів пізнавальної діяльності людини».

Польський вчений О.В. Долженко вважає [14], що «... навчання детерміноване метою, змістом і діями, за допомогою яких суб'єкт, який навчається, здобуває певні знання, уміння й навички; воно розгортається в результаті власної активності суб'єкта, що ніхто інший «за нього» не виявить, носить, головним чином, процесуальний характер, може протікати в різних формах, базується на пізнанні (прямому або непрямому) та індивідуальному досвіді, викликає зміни в поведінці».

У роботі [15] зазначено, що «розрізняють дві основні спрямованості процесу навчання: вивчення понять і формування вмінь (навичок)». При вивченні понять основна увага надається поданню і роз'ясненню нового концептуального і фактичного навчального матеріалу: означенню понять, опису їх властивостей і відношень, загальному і розгорненому опису процесів і т.п. При формуванні вмінь головна увага надається створенню проблемних ситуацій, що повторюються і поступово ускладнюються і вимагають від студента виконання певних дій.

Л.М. Фрідман [16] виділяє наступні етапи навчальної діяльності: 1) початково-мотиваційний, що містить такі основні навчальні дії: а) усвідомлення учнем основної навчально-проблемної ситуації, що вводить його в предмет майбутньої роботи з вивчення навчального матеріалу; б) формулювання основного навчального завдання; в) самоконтроль і самооцінка можливостей майбутньої діяльності щодо вивчення навчального матеріалу; 2) операціонально-пізнавальний, на якому учень вивчає навчальний матеріал, опановує знання, уміння й навички в процесі виконання основних навчально-пізнавальних завдань; 3) контрольно-оцінювальний, на якому учень узагальнює вивчений навчальний матеріал, проводить коригування помилкових дій, аналізує роботу, що виконана, оцінює свою діяльність.

Умовою успішного навчання у ВНЗ є інтерес студентів до навчального предмету, перебігу навчання і його результату. Цей інтерес пов'язаний з багатьма факторами: змістом навчального предмету, рівнем його складності; організацією процесу навчання; системою контролю й оцінювання знань; особистісними якостями самого викладача; системою цінностей студента, його найближчого оточення; взаєминами в студентському колективі. Тому одним з важливих завдань дидактики вищої школи є активізація навчально-пізнавальної діяльності студентів ([17-19] та ін.), яка «розуміється як цілеспрямована діяльність викладача, спрямована на розробку і використання такого змісту, форм, методів, прийомів і засобів навчання, які сприяють підвищенню пізнавального інтересу, активності, творчої самостійності студентів у засвоєнні знань, формуванні навичок і вмінь, застосуванні їх на практиці» [19]. У курсі з основ програмування творчий підхід студентів до навчальної діяльності формується шляхом залучення їх до розв'язування різноманітних задач, що поступово ускладнюються, на лабораторних заняттях, під час самостійної роботи, при підготовці рефератів і доповідей, при виконанні курсових робіт, розробці колективних проектів.

Точкою відліку початку підвищеного інтересу психологів щодо програмування є вихід першого видання книги Вейнберга „The Psychology of Computer Programming” більше 30 років тому. Предметом дослідження психології програмування є діяльність програміста, психологічні закономірності приймання і опрацювання повідомлень, функціонування психічних механізмів пам'яті і мислення у програміста, його працездатності, питання професійного відбору [20].

У дослідженнях J.G. Greeno [21], які засновані на інформаційному підході до психології навчання, запам'ятовування і розв'язування задач, запропоновано схему взаємодії компонентів пам'яті, які використовуються у роботі програміста і до яких відносять короткотривалу пам'ять, довготривалу пам'ять, робочу пам'ять. Згідно з цим дослідженням, постановка задачі (проблеми), яка повинна бути запрограмована, надходить до короткочасної пам'яті, у якій заздалегідь аналізуються відомості, що надійшли. Постійні знання програміста (знання синтаксису мови, методів побудови алгоритмів, опису структур даних і т.п.) знаходяться в довготривалій пам'яті. У робочій пам'яті відомості з короткочасної і довготривалої пам'яті об'єднуються в нові структури. У процесі розв'язування задачі, тобто генерації програми, нові відомості, що є у короткочасній і довготривалій пам'яті, і поняття, що відносяться до проблеми, об'єднуються в робочій пам'яті, а результат використовується для генерації розв'язку або, у разі навчання, запам'ятовується в довготривалій пам'яті для подальшого використання. Як відмічено в [22]: «Хороші програмісти завжди трошки ледачі. Вони сидять і чекають осяяння, замість того, щоб втілювати в життя ту ідею, що першою прийшла їм на думку». Це відбувається завдяки тому, що досвідчений програміст зберігає в пам'яті складні багаторівневі знання про поняття і методи програмування і в процесі осмислення задачі визначає, які із цих знань слід застосувати. Частина цих знань називають семантичними знаннями [22]. Ці знання належать до загальних істотних для програмування понять, але не пов'язані з конкретною мовою програмування. До семантичних знань входять визначення понять нижнього рівня, наприклад: як зберігаються дані в пам'яті комп'ютера, як виконується оператор розгалуження, як відбувається виклик підпрограми, які бувають типи даних. До них входять визначення понять середнього рівня, наприклад, обмін значень двох змінних, алгоритм пошуку більшого з двох значень, а також складніші алгоритми, такі як алгоритми сортування, бінарний пошук, обхід бінарного дерева пошуку т.п. Ці знання зберігаються у вигляді загальних змістових фреймів, значною мірою не залежних від знань синтаксису конкретних мов програмування або засобів програмування, таких як пакети підпрограм або середовища програмування.

У довготривалій пам'яті зберігаються й синтаксичні знання. Ці знання більш конкретні й докладні, а тому легше забуваються у порівнянні з семантичними знаннями. До синтаксичних знань входять такі елементи як правила опису операторів надання значення, розгалуження, повторення, зарезервовані слова, імена бібліотечних функцій і стандартних модулів і т.п.

На Рис. 1 зображено схему взаємозв'язків компонентів пам'яті при розв'язуванні задач з програмування.

Програмістам, психологам і педагогам відомо, що звичайно важко вивчити першу мову програмування, а другу подібну вивчити відносно легко тому, що легше вивчити нове синтаксичне подання, якщо вже відома відповідна семантична структура.

Завдання:
 Написати програму для розв'язування наступної задачі: Дано натуральне число n . Видалити із запису числа n цифри 0 і 5, залишивши порядок інших цифр без змін. Наприклад, із числа 59015509 треба отримати число 919.

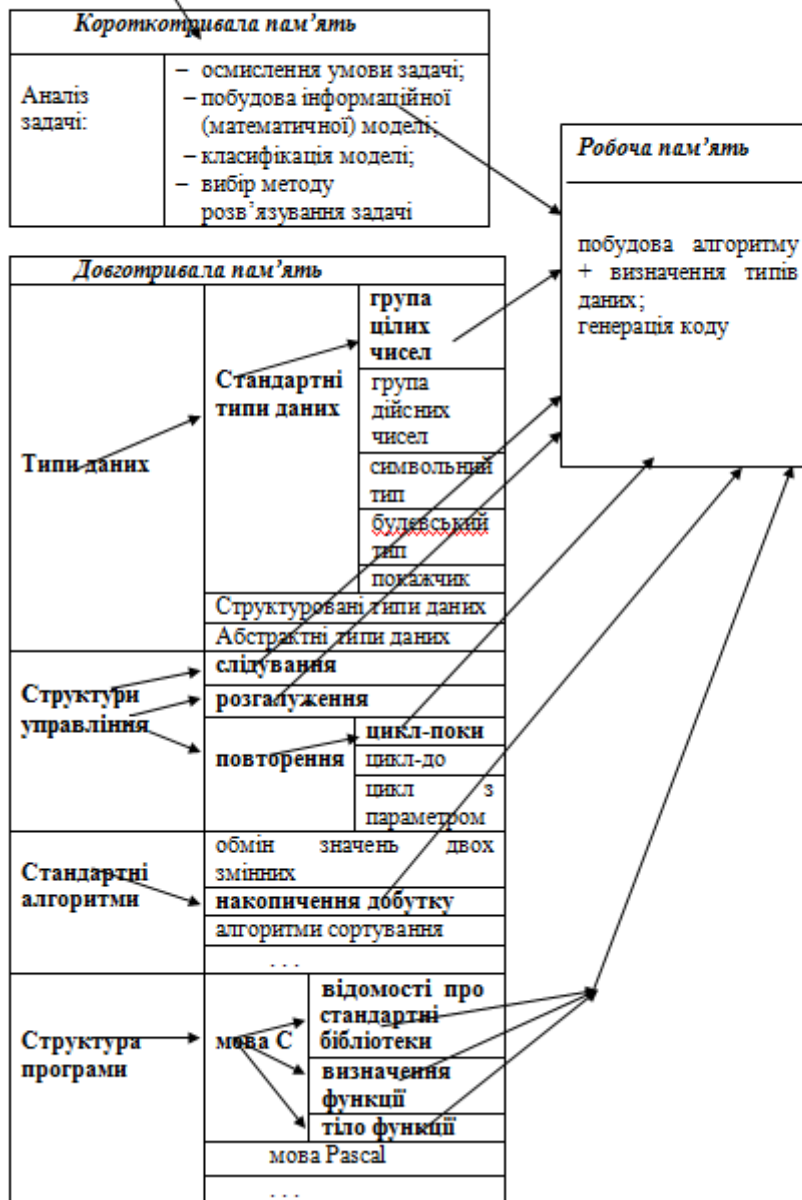


Рис. 1. Схема функціонування компонентів пам'яті при розв'язуванні задач з програмування.

Б. Шнейдерман [22] вказує на відмінності між семантичними і синтаксичними знаннями в довготривалій пам'яті програміста. Семантичні знання утримуються в процесі, який є інтелектуальною потребою осмисленого навчання, що включає розв'язування задач і вивчення навчальної літератури (підручників, довідників), що допомагає закріпити нові поняття сумісно з наявними семантичними знаннями. Синтаксичні знання, що носять інструктивний характер, заучуються механічно і не дуже перетинаються із вже наявними системами семантичних знань. Отриманню нових синтаксичних знань можуть перешкоджати одержані раніше синтаксичні знання, оскільки відбувається додавання нових відомостей, а не їх інтеграція. Дослідження психологів підтверджують, що при вивченні нових мов програмування програмісти дуже легко розуміють семантику основних конструкцій мови, але часто помиляються при кодуванні. Наш досвід роботи показує, що, наприклад, студенти, які вже мають досвід написання програм мовою Pascal, при програмуванні мовою C не ставлять крапку з комою перед службовим словом „else” в операторі розгалуження й не звертають увагу на розходження ідентифікаторів, у які входять однакові символи (букви алфавіту) в одній і тій самій послідовності, але в різних регістрах (рядкові й прописні).

Дослідники припускають, що процес написання програми розв'язування поставленої задачі складається із двох фаз. На першій фазі відомості про умову задачі надходять до системи мислення програміста і через короткотривалу пам'ять потрапляють до робочої пам'яті, де вони аналізуються. Аналогічно з довготривалою пам'яті програміста пересилаються узагальнені відомості (семантичні і

синтаксичні) для подальшого аналізу задачі. На другій фазі, при побудові загального плану розв'язування задачі, відбувається її покрокове уточнення. Спочатку розв'язок задачі уявляється в узагальнених термінах. Потім це уявлення розвивається до більш докладних планів аж до генерації конкретних кодів, що відображають найдрібніші деталі.

Однією з характеристик професійних якостей програміста є розуміння програм. Розуміння програм необхідне при налагодженні і при модифікації програм, а також при навчанні. Б. Шнейдерман у своїх дослідженнях припускає, що для розуміння програми програміст будує багаторівневу внутрішню семантичну структуру, використовуючи свої знання синтаксису мови програмування. На найвищому рівні програміст повинен розуміти, яка задача розв'язується за допомогою програми. На більш низьких семантичних рівнях програміст може розуміти знайомі набори операторів. Можлива ситуація, коли програміст може зрозуміти деталі нижнього рівня, не розуміючи загальної задачі. „Головне полягає в тому, що програмісти виробляють внутрішню семантичну структуру для подання синтаксису програми, але не запам'ятовують і не розуміють програму як впорядкований набір рядків” [22]. Процес запам'ятовування, за допомогою якого програміст перетворює програму у внутрішню семантичну структуру, має „кусковий” характер [22]. Замість того, щоб аналізувати програму символ за символом, програміст розпізнає призначення груп операторів, потім збирає ці фрагменти разом для формування більш великих фрагментів доти, поки не буде зрозуміла вся програма. Тому, що внутрішня семантична структура програми розробляється самим програмістом, ці знання не швидко забуваються і доступні для різних цілей: використання інших структур даних для розв'язування задачі, написання програми розв'язування задачі іншою мовою програмування, роз'яснення призначення програми кому-небудь.

У процесі навчання майбутніх інженерів-програмістів також є синтаксичні і семантичні аспекти. Так, у курсах навчання другої або третьої (не першої) мови програмування однієї парадигми основна увага приділяється синтаксичним аспектам семантичних понять. Курси з розробки й аналізу алгоритмів зосереджені на семантичних знаннях.

Згідно [22], семантичні і синтаксичні знання розділені. Багаторівневі семантичні знання, отримані в основному завдяки цілеспрямованому навчанню, знаходять своє втілення в багаторівневому методі розробки внутрішньої семантики для деякої конкретної задачі. Відносні синтаксичні знання, сформовані, в основному, завдяки механічному заучуванню, цілком обмежені мовою. Семантичні знання необхідні для аналізу задачі, у той час як синтаксичні знання використовуються на етапах кодування і виконання.

Багато вчених (Ю.І. Машбиць [23], Н.Ф. Тализіна [7] та ін.) підкреслюють неприпустимість зведення навчальних цілей тільки до засвоєння предметних знань і вмінь. На їхній погляд у систему цілей необхідно включати вміння вчитися. Спираючись на цю точку зору, вважаємо, що у процесі навчання основ програмування треба визначити систему завдань, за допомогою якої можна буде сформувати у студентів знання, уміння й навички, необхідні й для їх подальшого навчання й у наступній професійній діяльності. Система завдань з програмування як засіб навчання є одним з найважливіших компонентів методичного забезпечення в навчанні основ програмування ([24]).

Також необхідно вирішити питання про складність змісту навчального матеріалу та його обсяг. Зміст навчального матеріалу, з одного боку, повинен бути максимально наближеним до сучасного рівня розвитку комп'ютерних наук, з іншого боку, повинен бути доступним студентам. Також треба систематизувати й визначити послідовність його подання. Важливо визначити міру складності навчального матеріалу, тому що занадто спрощений або занадто складний зміст ведуть до зниження інтересу й пізнавальної активності студентів у процесі навчання. Обсяг навчального матеріалу не повинен бути надмірним, щоб не перевантажувати студентів у процесі навчання.

Так само важливим є визначення необхідної кількості завдань для оволодіння навчальним матеріалом і формування практичних умінь і навичок. Ю.К. Бабанський відзначає, що «спеціальні дослідження показали, що якщо дотримуватися принципів систематичності і послідовності, то для формування визначених вмінь і навичок досить виконати п'ять вправ при середньому рівні складності навчального матеріалу. Для визначення кількості вправ у навчально-методичних комплексах необхідна побудова цих вправ таким чином, щоб на мінімальному числі вправ учні опанували б загальним принципом виконання подібного роду дій» [25]. Зведення кількості завдань до мінімуму приводить до того, що їх виявляється недостатньо для забезпечення цілей навчання. З іншого боку, занадто велика кількість завдань ставить перед студентом нездоланий бар'єр.

Зазначимо, що для забезпечення якісного засвоєння студентами знань з основ програмування слід не тільки нарощувати число досліджуваних понять, але і вивчати ті самі поняття на різних рівнях деталізації, наукової строгості. Дидактичні принципи систематичності, послідовності навчання при побудові курсу з основ програмування доцільно реалізовувати у вигляді дидактичної спіралі – багаторазове повернення на новому рівні [26]. Мається на увазі оволодіння студентами знаннями й

уміннями в контексті, що ускладнюється, за допомогою збагачення, розвитку й узагальнення досліджуваних понять.

Література

1. Леонтьев А.Н. Деятельность. Сознание. Личность. – М.: Политиздат, 1975. – 304 с.
2. Выготский Л.С. Избранные психологические произведения / Выготский Л.С. – М.: Учпедгиз, 1956. – 426 с.
3. Костюк Г.С. Навчально-виховний процес і психічний розвиток особистості / Під ред. М.М. Проколієнко. – К.: Рад. шк., 1989. – 608 с.
4. Рубинштейн С.Л. Основы общей психологии. – 2-е изд. – М.: Учпедгиз, 1947. – 704 с.
5. Гальперин П.Я. Развитие исследований по формированию умственных действий / П.Я. Гальперин // Психологическая наука в СССР. – М.: Изд-во АПН РСФСР, 1959. – Т. 1. – С. 441–469.
6. Кабанова-Меллер Е.Н. Психология формирования знаний и навыков у школьников. Проблема приемов умственной деятельности. М.: Изд-во АПН РСФСР, 1962. – 376 с.
7. Талызина Н.Ф. Управление процессом усвоения знаний. – М.: Изд-во МГУ, 1975. – 343 с.
8. Ананьев Б.Г. Избранные психологические труды: в 2 т. – / Б.Г. Ананьев – М.: Педагогика, 1980. – Т. 2. – 1980. – 288 с.
9. Ананьев Б.Г. Избранные психологические труды: в 2 т. / Б.Г. Ананьев – М.: Педагогика, 1980. – Т. 1. – 1980. – 232 с.
10. Алексюк А.М. Педагогіка вищої освіти України / А.М. Алексюк. – К.: Либідь, 1998. – 560 с. – (Історія. Теорія: Підручник).
11. Ананьев Б.Г. Человек как предмет познания / Б.Г. Ананьев – Л.: Изд-во Ленинград. ун-та, 1968. – 339 с.
12. Педагогика и психология высшей школы: Учеб. пособие для студентов и аспирантов вузов // С.И. Самыгин, М.В. Буланова-Топоркова, А.В. Духавнева и др. – Ростов на Дону: Феникс, 1998. – 544 с.
13. Український педагогічний словник / [авт.-уклад. Гончаренко С.]. – Київ: Либідь, 1997. – 376 с.
14. Куписевич Ч. Основы общей дидактики: Пер. с польск. О. В. Долженко. – М.: Высш. шк., 1986. – 368 с.
15. Экспертно-обучающие системы / Петрушин В.А.; Отв. ред. А.М. Довгяло; АН УССР, Ин-т кибернетики. – Киев: Наук. думка, 1992. – С. 196.
16. Фридман Л.М. Педагогический опыт глазами психолога: Книга для учителя. – М.: Просвещение, 1987. – 224 с.
17. Аврамчук Л.А. Формування активної пізнавальної діяльності студентів / Аврамчук Л.А. // Педагогіка і психологія. – 1997. – № 3. – С. 122–125.
18. Іваськів І.С. Активізація навчально-пізнавальної діяльності учнів на основі систем штучного інтелекту при навчанні інформатики в старшій школі: Дис. канд. пед. наук: 13.00.02 / НПУ ім. М.П. Драгоманова. – К., 2000. – 250 с.
19. Слєпкань З.І. Наукові засади педагогічного процесу у вищій школі. – К.: НТУ, 2000. – 210 с.
20. Преподавание информационных технологий в России. Открытая всероссийская конференция. – <http://www.sparc.spb.su/oio/cs/>. <12.04.07>
21. Greeno, J.G., The structure of memory and the process of problem solving, University of Michigan: Human Performance Center Technical Report 37, 1972. – p. 128.
22. Шнейдерман Б. Психология программирования: Человеческие факторы в вычислительных и информационных системах. Пер. с англ. – М.: Радио и связь, 1984. – 304 с.
23. Машбиц Е.И. Психолого-педагогические проблемы компьютеризации обучения. – М.: Педагогика, 1988. – 192с.
24. Гришко Л.В. Удосконалення системи завдань з основ програмування як засіб підвищення практичної значущості результатів навчання майбутніх інженерів-програмістів. Вісник Черкаського університету. Серія Педагогічні науки. – Випуск №101. – Черкаси, 2007. – С. 40-45.
25. Бабанский Ю.К. Дидактические проблемы совершенствования учебных комплексов //– М.: Просвещение, 1980. – С. 17–33.
26. Computing Curricula 2001. – [Електронний ресурс]. – Режим доступу: <http://www.computer.org/education/cc2001>.