

5. Бахарев А. Т. Оптимизация многопороговых моделей / А. Т. Бахарев // Пробл. случайного поиска. – Рига – 1975, вып. – С. 209–214.
6. Blum A. Training a 3-Node Neural Network is NP-Complete / A. Blum, R. Rivest // Neural Networks. – 1992, Vol 5. – PP. 117-127.
7. Гэри М. Вычислительные машины и труднорешаемые задачи / М. Гэри, Д. Джонсон. – М.: Мир, 1982. – 416 с.
8. Rivest, R. Learning decision lists / R. L. Rivest // Machine Learning – 1987. – # 2 – PP. 229-246.
9. Anthony, M. Discrete Mathematics of Neural Networks / M. Anthony. – Philadelphia: SIAM, 2001. – 132 с.

Костюченко А.

**МЕТОДИЧНІ ПІДХОДИ ДО ВИКЛАДАННЯ ПРОГРАМУВАННЯ  
В ПЕДАГОГІЧНИХ ВНЗ ЯК ОДИН З ПОЧАТКОВИХ КРОКІВ  
ДО СТВОРЕННЯ ВІЛЬНОПОШИРЮВАНИХ ПРОГРАМНИХ ЗАСОБІВ  
НАВЧАЛЬНОГО ПРИЗНАЧЕННЯ**

**Постановка проблеми.** Майже одночасно з появою в школі комп'ютерів були створені програми, призначені для навчання школярів програмування. Потім з'явилися програми для навчання іншим предметам. За такими програмами та інструментальними засобами закріпився термін „педагогічні програмні засоби” (ППЗ). Можна виділити кілька основних проблем використання ППЗ у навчальному процесі, першою з яких є те, що програмні засоби спочатку створюються, а потім розпочинаються способи знайти їм місце в навчальному процесі, причому при створенні таких програмних засобів розробники не завжди узгоджують свої дії з вчителями практиками. Другою проблемою є те, що більшість програмних засобів, що пропонуються для підтримки викладання тих чи інших предметів, працюють під пропрієтарними операційними системами, що пов'язано з тотальним «майкрософтоцентризмом» кінця 90-х – початку 2000-х років. Третьою проблемою можна назвати платність таких програмних засобів. Тобто використання пропонованих педагогічних програмних засобів може бути досить дорогим задоволенням для сучасних українських вчителів і шкіл. Зважаючи на вищесказане, постає задача створення власних програмних засобів, які можуть використовуватися в навчальному процесі і вільно розповсюджуватися серед українських шкіл, оскільки вільне програмне забезпечення містить у собі чималу кількість важливих, у тому числі й стратегічних переваг, чи не найголовнішою серед яких є можливість суттєвого заощадження бюджетних коштів. Така задача може бути покладена на вчителів інформатики, що в свою чергу потребує своєрідного навчання студентів педагогічних ВУЗів відповідних спеціальностей.

**Аналіз публікацій з проблеми дослідження.** Класифікацією методів навчання займалися Д.О. Лордкіпанідзе, Є.Я. Голант, Н.М. Верзілін, І.Я. Лернер, М.І. Махмутов, Ю.К. Бабанський, А.М. Алексюк та інші. Разом з тим окремі методичні підходи до навчання програмування зустрічаються в роботах: М.М. Скаткіна, Є.М. Кабанової-Меллер, Г.С. Костюка, А.П. Ершова.

**Мета статті** - огляд методичних підходів, які можуть використовуватися при викладанні програмування в педагогічних ВНЗ.

**Виклад основного матеріалу дослідження.**

**Класичні методи навчання програмування.** Перш ніж розпочати огляд методичних підходів, потрібно розглянути поняття класичних методів навчання та їх класифікації, оскільки саме вони використовуються в будь-яких методичних підходах до навчання, в тому числі і навчання програмування.

Методи навчання (від гр. *methodos* — спосіб пізнання, шлях дослідження) — це упорядковані способи діяльності вчителя й учнів, спрямованих на засвоєння учнями системи знань, набуття умінь і навичок, їх виховання і загальний розвиток [3].

Під методом навчання у ВНЗ розуміють впорядковані способи взаємопов'язаної діяльності викладача і студента, спрямовані на досягнення поставлених цілей вивчення конкретної наукової дисципліни.

Класифікації методів навчання відрізняються за ознакою, покладеною в основу кожного з них. Спільним для всіх підходів є те, що в кожному з них відображається три групи ознак, які характеризують: навчально-пізнавальну діяльність, педагогічну діяльність, предмет спільної діяльності учасників педагогічного процесу.

Розглянемо основні класифікації методів навчання з точки зору застосування їх при вивченні курсу програмування. Серед яких можна виділити три: за способом передачі відомостей від викладача до студента, за основними видами дидактичних проблем, що вирішуються на заняттях, за характером пізнавальної діяльності студентів.

I. За способом передачі відомостей від викладача до студента виділяють такі методи: словесні, або вербальні — бесіда, пояснення, розповідь, лекція, інструктаж; наочні або демонстраційні — ілюстрування та демонстрування; практичні — вправи, лабораторні та практичні роботи, дослідні роботи. Під час навчання курсу програмування словесні методи використовуються при викладанні лекційного матеріалу. В наочних методах головну роль відіграє демонстрація та візуалізація викладачем структур, алгоритмів, що дозволяють в наочній формі динамічно відображати деталі їх роботи. Практичні методи частіше всього застосовуються при розв'язуванні задач, виконанні лабораторних та практичних робіт. При цьому наголос робиться на практичні методи, оскільки під час їх застосування студенти отримують не лише нові знання, але й практичні навички.

II. За основними видами дидактичних проблем, що вирішуються на заняттях, можна виділити такі методи: методи набуття знань, методи формування вмінь, методи застосування знань, методи творчої діяльності, методи перевірки компетентностей.

III. Класифікація методів навчання програмування за характером пізнавальної діяльності студентів: пояснювально-ілюстративний, або інформаційно-рецептивний; репродуктивний; проблемний; частково-пошуковий, або евристичний метод навчання; метод проектів або дослідницький метод навчання; метод активного навчання; метод інтерактивного навчання. Інформаційно-рецептивний і репродуктивний методи забезпечують засвоєння студентами передбачених робочою програмою фактів, методів, алгоритмів, тощо і формування у них вмінь і навичок їх застосування, що є передумовою успішного навчання і творчої діяльності студентів, тоді як методи проблемного навчання — метод проблемного викладу, евристичний і дослідницький методи формують риси творчої особистості студентів.

**Задачний підхід.** На сучасному етапі навчання програмування найпоширенішим є задачний підхід. М.М. Скаткін відзначає, що “сучасна психологія і дидактика розкрили значну роль задач у навчанні як одного з найважливіших чинників збудження і підвищення пізнавальної і практичної активності учнів”.

Як і в більшості предметів природничо-математичного циклу, в програмуванні уміння та навички розв'язування задач формуються безпосередньо у процесі їх розв'язування на основі застосування знань відповідних теоретичних основ.

Особливістю розв'язування задач з програмування є те, що це розв'язування передбачає реалізацію алгоритму деякою конкретною мовою програмування, що вимагає від студента не лише знання відповідних мовних конструкцій, а і вміння працювати в певному середовищі програмування.

Як показує досвід, у процесі навчання основ програмування можна виділити чотири стадії вміння і здатності розв'язування задач студентами: репродуктивне програмування, осмислене програмування, доказове програмування, творче програмування.

Такий підхід сприяє підвищенню інтересу студентів до навчання, надає їм впевненості у свої можливостях активізує їх пізнавальну активність, стимулює навчальну діяльність. У той же час розв'язання задачі за готовим алгоритмом, запропонованим викладачем, не вимагає від студента самостійного мислення. А навички вміння розв'язувати задачі набуваються тільки в результаті самостійної праці.

**Методичний підхід до парного програмування.** В процесі навчання програмування доцільно використовувати метод «парного програмування», суть якого полягає в тому, що під час виконання завдань два студента одночасно працюють за одним комп'ютером. При цьому один із студентів управляє комп'ютером і, в основному, думає над кодуванням у деталях. Другий студент зосереджений на задачі в цілому і постійно переглядає код, що набирається першим студентом. У цьому випадку студенти, що працюють у парі, постійно спілкуються з приводу навчального завдання, здійснюють взаємонавчання і взаємоконтроль.

Для підходу парного програмування обов'язковою є умова обміну ролями студентів у процесі роботи, тобто періодично студенти, що працюють у парі, повинні мінятися місцями за комп'ютером.

**Колективний проект як методичний підхід до активного навчання програмування.** В більшості випадків сучасне програмування є колективним, і вклад кожного окремого програміста в спільний проект впливає на успіх всієї команди. Необхідно розуміти важливість формування і розвитку у студентів таких важливих якостей, як уміння працювати в колективі і співпрацювати з іншими колективами, наприклад, колективом вчителів-предметників з інших дисциплін чи колективом методистів.

Учасники команди працюють над спільною задачею, шукають шляхи її розв'язання. Кожен сумлінний учасник несе особисту відповідальність за код програми і вважає доцільним знаходити і відстоювати кращі розв'язки проблеми. Так, між студентами можуть виникати суперечки, а, як відомо, в суперечках народжується істина.

Такий вид навчальної діяльності також спонукує студентів до самостійної творчої роботи і глибокого усвідомленого вивчення курсу програмування.

**Підхід з максимальним охопленням матеріалу** полягає в розгляді більшої кількості навчального матеріалу без глибокого занурення в нього. Застосування цього підходу дає студентам більш цілісний погляд на дисципліну, що вивчається, уявлення про низку важливих тем, замість того, щоб відразу занурювати їх у деталі однієї конкретної проблеми. Студенти, які навчаються за таким підходом, одержують більш широке уявлення про програмування, що дозволяє їм рухатися далі, впевнено і цілеспрямовано, опановуючи інші комп'ютерні курси. До недоліків такого підходу можна віднести те, що розгляд широкого спектру теоретичних питань з програмування на ранній стадії навчання, що передбачає такий підхід, ускладнює засвоєння навчального матеріалу студентами молодших курсів.

**Підхід з орієнтацією на алгоритми** при опануванні основних концепцій програмування передбачає використання псевдокоду замість реальної мови програмування. Застосування цього підходу мінімізує зусилля студента, що йдуть на вивчення специфічних синтаксичних конструкцій конкретної мови програмування. Натомість від студентів вимагається обґрунтування і роз'яснення алгоритмів, які вони створюють. Це дозволяє студентам працювати з широким діапазоном типів даних і базових структур. Після того, як студенти опанують основні типи алгоритмів і типи даних, вони можуть починати використовувати одну з мов програмування.

**Підхід з орієнтацією на апаратну частину** передбачає вивчення основ інформатики і, зокрема, основ програмування, починаючи з апаратно-логічного рівня. Зазначається, що таким чином студенти можуть вивчити інформатику більш глибоко і послідовно. Тільки після сформування у студентів розуміння структурних особливостей апаратної частини, машинної логіки і математики, здійснюється перехід до розгляду

програмування мовами високого рівня. З погляду розробників такий підхід більш придатний для студентів, що вважають за краще розуміти процес обчислення на ЕОМ у всіх його деталях.

#### Джерела

1. Дидактика средней школы / Под ред. М.А.Данилова, М.Н.Скаткина. – М., 1975. – С. 162.
2. Пойя Д. Как решать задачу / Пойя Д. – М.: Либроком, 2010. – 208 с.
3. Гончаренко С.У. Український педагогічний словник / С.У. Гончаренко. – К.: Либідь, 1997. – 376 с.

Кулаков М., Сперкач М.

### СИСТЕМА АНАЛІЗУ ВПЛИВУ СКЛАДУ ТА РЕЖИМІВ ВИКОРИСТАННЯ ЕЛЕКТРИЧНИХ АКУМУЛЯТОРІВ НА ЕФЕКТИВНІСТЬ ЇХ РОБОТИ

В час стрімкого розвитку техніки та науки велику роль відіграє використання електроенергії. Існує чимало оригінальних способів добути енергію із навколишнього простору: від Сонця, вітру, припливів, теплової енергії Землі тощо. Але проблемою залишається накопичення та збереження енергії. Існує ряд способів накопичення енергії з точки зору типу енергії та ємності запасу, потужності, габаритів та обслуговування.

Найбільш звичний спосіб накопичення електричної енергії реалізується з використанням готових блоків, які підключаються до джерела, тобто акумуляторів. Акумулятори практично не вимагають обслуговування і зручні з точки зору споживання накопиченої енергії[1].

Існують різні сфери людської діяльності, в яких мобільність вимагає наявності бортового джерела струму. Рішення сучасної науки – використання акумуляторів. Акумулятори використовуються майже всюди, починаючи з автомобілів, морського, повітряного і залізничного транспорту, додамо сюди радіотехніку, телефонні і електричні станції, електрокари, для освітлення і сигналізації на штучних супутниках Землі, космічних ракетах тощо.

Розповсюдженими на даний час є акумулятори з використанням літія, основними перевагами котрих є: дуже малий рівень саморозряду (менш ніж 1% в рік); висока напруга заряду (до 3.5В); дуже широкий діапазон робочих температур; стабільність напруги; більш висока екологічна безпечність у порівнянні з іншими видами хімічних джерел струму [2].

Існує брак інформації щодо використання різних типів акумуляторів та недостатній рівень вивченості електричних властивостей різних матеріалів, з яких виробляють ці батареї та їх впливу на роботу акумулятора. У зв'язку з цим виникла ідея створення системи аналізу впливу специфіки використання батареї, кількості та якості підзарядок тощо на її життєвий цикл. Результати роботи цієї системи допоможуть збільшити ефективність використання матеріалів, що дозволить зменшити розмір акумуляторів та знизити їх ціну.

Для реалізації цієї ідеї було залучено сервіс-компанію, яка надає послуги оренди електричних акумуляторів.

Вивчаючи властивості матеріалів різних акумуляторів отримуємо великі обсяги інформації, які надходить із спеціального пристрою в процесі дослідження. Виникає необхідність обробити цей масив інформації та саме для її обробки і розробляється система, про яку далі піде мова.

В процесі роботи було визначено основні компоненти та розроблено архітектуру майбутньої системи; проведено аналіз технічної літератури щодо методів побудови розподілених сховищ даних, алгоритмів стиснення та відновлення даних; виявлено