

МЕТОДИЧНІ РЕКОМЕНДАЦІЇ ДО ОРГАНІЗАЦІЇ ПРАКТИЧНИХ РОБІТ З ІНФОРМАТИКИ В УМОВАХ ВПРОВАДЖЕННЯ МОДЕЛЬНО-СИМВОЛІЧНОЇ ТЕХНОЛОГІЇ

С.М. Прийма

Мелітопольський державний педагогічний університет

Реформування системи освіти України, зміна національної освітньої парадигми, що спрямовані на формування та розвиток соціально-активної, творчої особистості, вимагають впровадження нових концепцій навчання, пошуку шляхів вдосконалення педагогічного процесу. Головний стратегічний напрямок розвитку вітчизняної освіти, як шкільної так і вузівської, передбачає переосмислення не тільки змісту, але й методів, організаційних форм, всього стилю навчання.

Криза освіти, пов'язана з протиріччям між об'ємом матеріалу, часом, виділенням на його вивчення, та вимогами до рівня кваліфікації майбутнього спеціаліста (в інформатиці ця проблема постає особливо актуально), призводить до необхідності зміни традиційного навчання новим, що ґрунтувалося б на відмові від стандартно-стереотипного, шаблонного мислення та орієнтувалося б на підготовку здатної до швидкої адаптації в сучасних умовах динамічної зміни суспільства творчої особистості [1]. Саме таким і є розвивальне навчання, головна мета якого полягає в розвитку особистості як найвищої цінності суспільства, розвитку її розумових здібностей, формування громадян, здатних до свідомого суспільного вибору, збагачення на цій основі інтелектуального, творчого потенціалу українського народу.

При розвивальному навчанні діяльність викладача повинна бути спрямована на розробку та використання таких форм, змісту, прийомів та засобів навчання, що сприяли б підвищенню рівня самостійності, свідомості та творчої активності студентів в засвоєнні знань, формуванні вмінь та навичок, в їх практичному використанні, здібностей приймати самостійно рішення [2]. Нові педагогічні технології, що підтримують розвивальний тип навчання, повинні бути перш за все динамічними та гнучкими, забезпечувати режим найбільшого психолого-педагогічного, методичного та організаційного сприяння для реалізації індивідуальних інтересів, можливостей та здібностей студентів на основі використання вибору та відповідальності за нього [3].

Запропонована нами до впровадження в процес вивчення інформатики модельно-символічна технологія організації розвивального навчання [4], автором якої є П.О. Барабоха, в повній мірі дозволяє реалізувати всі завдання розвивального навчання.

Але процес впровадження основних положень проблемної символіки, на яких базується модельно-символічна технологія, вказує на нагальну необхідність переосмислення організаційних форм навчання, зокрема таких як практична робота.

Процес навчання у вищій школі включає практичні заняття, що призначені для поглибленого вивчення тієї чи іншої дисципліни. Аудиторні

практичні заняття мають виключно важливу роль у відпрацюванні у студентів навичок застосування отриманих знань для рішення практичних задач в процесі спільної діяльності з викладачем. Якщо лекція започатковує основи наукових знань в узагальненій формі, то практичні заняття призначені для поглиблення, розширення та деталізації цих знань, сприяють навичкам професійної діяльності [5].

При викладанні курсу інформатики перехід від теорії до конкретної роботи за комп'ютером на практичних заняттях для виконання вправ, завдань, написання програм та їх відлагодження супроводжується швидкою індивідуалізацією. Навіть однакова на початку, індивідуалізація ставить викладача в ситуацію «багатоверстатника», і він фізично не встигає допомогти всім. Для вирішення цього питання виконання практичної роботи за комп'ютером підтримується інструкцією [6]. Але, як показує досвід та опитування студентів, робота за інструкцією зводиться до бездумного виконання певних дій чи операцій, відпрацювання певних навичок, призводить до зниження самостійності та рівня творчої діяльності, що в подальшому негативно впливає на мислення та якість знань студентів.

Саме це й вказує на необхідність нашого пошуку з удосконалення змісту, методів та форм проведення практичних робіт з інформатики у вищій школі.

Запропонований нами підхід до організації практичних робіт в умовах впровадження модельно-символічної технології базується на реалізації таких етапів самостійності: від повного керівництва вчителя через дозовану допомогу до самоуправління пізнавальною діяльністю, переходу до творчої діяльності, під час якої самостійність реалізується повністю [6]. Звісно, з одного боку, процес проведення занять у відповідності до такого підходу для викладача стає більш цікавішим, творчим процесом, оскільки процес керування самостійною діяльністю студентів кожного разу унікальний та цікавий, але, з іншого боку, вимагає від викладача високої кваліфікації, уваги до процесу організації самостійної діяльності, до логіки побудови та організації кожного заняття та всієї спільної діяльності [7]. Реалізація основних принципів такого підходу передбачає кардинальні зміни в організації практичних робіт, зокрема, вирішенні питань диференціації завдань. Як відзначають деякі науковці [8], провідним принципом диференціації змісту освіти, зокрема в умовах впровадження розвивального навчання, повинна виступати не диференціація змісту освіти, що домінує при традиційному навчанні, наприклад, завдань (одним-простіше, іншим-складніше), а диференціація допомоги студентам з боку викладача без істотного зниження складності змісту (одні студенти потребують більшої допомоги, інші - меншої). Слід зазначити, що завжди можна виділити групу студентів, котрим можна надати повну самостійність [9]. Такий підхід дасть змогу кожному досягти максимального прояву свої можливостей та здібностей, творчого самоствердження та самовизначення.

Переваги такої форми організації практичних робіт полягають в наступному:

- можливості особистості проявити такі індивідуальні риси мислення як широта (здатність охопити все завдання повністю, не випускаючи в той час і необхідності деталей), самостійність (характерну рису вміння висловити нові завдання та знаходити шляхи їх реалізації, не звертаючись за допомогою

до інших), критичність (вміння об'єктивно оцінювати не тільки чужі, а й власні думки, ретельно та різнобічно перевіряти всі положення та висновки) [2];

- можливості переходу студентів від одного диференційованого рівня на інший в залежності від підготовленості до певного заняття (в традиційній формі ця диференціація відбувається на початку вивчення курсу і прослідковується на протязі всього його вивчення);
- можливості вільного вибору, що сприяє розвитку критичного ставлення до своїх знань та відповідальності за зроблений вибір;
- активізації самостійної навчально-пізнавальної діяльності, що проявляється в самостійній підготовці до кожного практичного заняття;
- економії навчального часу та раціонального його використанні на заняттях за рахунок самостійної підготовки вдома (для уточнення та остаточного вибору стратегії виконання рівня В достатньо ознайомитися з методичними рекомендаціями, а виконання рівня С передбачає втрату часу лише на уважне ознайомлення з детальною інструкцією);
- практичній значущості завдань, що значно підвищує рівень замотивованості їх виконання;
- обов'язковому контролю в кінці кожного заняття, що виключає можливість списування алгоритму розв'язку або результатів роботи.

Отже, на завершення хотілося б ще раз зазначити на необхідності вдосконаленні змісту, методів та форм організації навчального процесу під час викладання інформатики та звернути увагу на ті з них, що дійсно сприяють підвищенню рівня самостійності, свідомості та творчої активності студентів в засвоєнні знань, формуванні вмінь та навичок, в їх практичному використанні.

Література

1. Абрамчук В.С. Жовтяк І.В. Підхід до побудови навчальної програми з інформатики в педагогічному університеті // Тези доповідей Всеукраїнської науково-практичної конференції «Сучасний стан і перспективи шкільних курсів математики та інформатики у зв'язку з реформуванням у галузі освіти». Дрогобич, 2000.-С.212-214.
2. Столяренко Л.Д., Самыгин С.И. Психология и педагогика в вопросах и ответах. Серия «Учебники, учебные пособия». - Ростов-на-Дону:Феникс, 1999.-576 с.
3. Гохберг О.С. Проблема разработки и реализации гибких педагогических технологий обучения в вузе: Дис... канд.пед.наук:13.00.01/Славянский пед. ин.-т.-Славянск, 1995.-181 с.
4. Прийма С.М., Еремеев В.С. Використання системи проблемно-символічних сигналів при вивченні програмування // Збірник наукових праць (Матеріали науково-практичної конференції «Інформаційні технології в освіті»).- Бердянськ.:БДПШ.-2001.-С.24-29.
5. Основи педагогтики и психологии высшей школы / Под ред. А.В. Петровского.- МГУ.-1986.-304 с.
6. Бочкин А.П. Методика преподавания информатики: Учеб. пособие.- Мн.:Высш.шк., 1998.- 431 с.

Кушніренко А.Г., Лебедев Г.В.12 лекцій о том, для чего нужен школьный курс информатики и как его преподавать. Методическое пособие.- М.:ЛБЗ,2000.-464 с.

Вибанский Ю.К., Потапник М.М. Оптимизация педагогического процесса.- 2-о изд., перераб. и доп.- К.:Рад.шк., 1983.-287 с.

Фурман А.В. Психолого-педагогічна теорія навчальних проблемних ситуацій: Дис... д.-ра психол.наук:19.00.07.-К.,1993.-449 л.

МАШИННО-ЗАЛЕЖНА ОПТИМІЗАЦІЯ КОДІВ ПРОГРАМ

В.І.Салапатов

Черкаський державний університет ім. Б.Хмельницького

Для наближення програм, що створені на мовах високого рівня, до їх асемблерних еквівалентів за мінімальним розміром об'єктного коду при максимальній швидкості виконання необхідно корінним чином змінити принципи генерації коду. Саме ця фаза компіляції як раз і вносять надмірність об'єктного коду. Це перш за все обумовлено неповним використанням інформаційних ресурсів, особливо регістрів процесора та співпроцесора. В сучасних компіляторах (С, С++) здійсненна оптимізація інформації, що розміщена в стеку. Але ж найбільш ефективно команди виконуються саме у регістрах. З іншого боку, необхідно в максимальній мірі використовувати дані саме у регістрах. Це дозволить уникнути зайвих команд завантаження у регістри, також чимало різного роду перевірок. Для цього пропонується використовувати базу даних після обробки кожного оператора програми. Ця база даних повинна зберігати інформацію про вміст регістрів, стеку та змінних пам'яті. Використання цієї інформації дозволить уникнути формування зайвих команд машинних команд. З іншого боку, необхідно мати базу даних про семантику машинних команд їх формат та синтаксис асемблерних операторів. Остання інформація потрібна для синтезу програми в символічному асемблерному вигляді. Використання мови асемблера, як проміжного уявлення програм, широко застосовується багатьма сучасними компіляторами. Мова Асемблер в даному випадку дозволяє здійснити машинно-залежну оптимізацію машинного коду на символічному рівні. Формальне подання даних через певні ділянки баз даних обумовлює оптимізацію машинного коду програм. Використання такого підходу синтезу коду програм замість існуючої генерації коду дозволить створювати ефективні програми. Структурна схема фази синтезу коду має такий вигляд, як подано на малюнку.

Будемо вважати, що внутрішнє уявлення програми після її обробки на фазі синтаксичного аналізу та машинно-незалежної оптимізації вже сформована і подається до синтезатора коду. Синтезатор коду, в свою чергу, синтезує об'єктний код програми з урахуванням семантики кожного елемента внутрішнього уявлення. Після вибору групи команд потрібної семантики з бази даних про семантику та