

НАВИЧКИ РОБОТИ З КОМП'ЮТЕРНИМИ ПРОГРАМАМИ ЯК СКЛАДОВА ІНФОРМАЦІЙНОЇ КУЛЬТУРИ ОСОБИСТОСТІ

Циммерман Геннадій Анатолійович,
Циммерман Олена Володимирівна,
Запорізький національний університет, м. Запоріжжя, Україна

Інформаційна культура – якісна характеристика життєдіяльності людини в області отримання, передачі, зберігання і використання інформації, де пріоритетними є загальнолюдські духовні цінності.

На рівні особи інформаційна культура виявляється в інформаційному середовищі, яке дозволяє вирішувати проблеми доступу до знання, одержаного у будь-який момент і в будь-якому місці. При цьому користувач інформації виступає у взаємодії з інформаційним середовищем не як обмежений виконавець деякої соціальної ролі, а як особа зі своїм індивідуальним творчим вибором, здібна до активної рефлексії над власним процесом мислення. Важливою та актуальною складовою інформаційної культури особи є вектор постійної і систематичної самоосвіти, тобто «освіта не на все життя, а крізь усе життя». Самостійна пізнавальна діяльність дозволяє постійно підвищувати і удосконалювати кваліфікацію і збагачувати протягом всього життя знання, винесені з освітніх установ.

Критеріями інформаційної культури людини можна вважати його вміння адекватно формулювати свою потребу в інформації, ефективно здійснювати пошук потрібної інформації у всій сукупності інформаційних ресурсів, переробляти інформацію і створювати якісно нову, вести індивідуальний інформаційний пошук, адекватно відбирати і оцінювати інформацію, а також здібність до інформаційного спілкування і комп'ютерну грамотність [1].

На підтримку важливості у зазначених напрямках питання інформаційної культури особистості для забезпечення потреб держави та суспільства у становленні конкурентоспроможної, мислячої особистості, готової протягом життя самовдосконалюватись в освіті впроваджується так звана компетентнісна парадигма [2,3,4,5]. У цьому аспекті шкільна та вузівська інформатика виступають важливою ланкою для формування навичок кваліфікованого використання сучасних інформаційних технологій. Тобто компетентне, ефективне володіння учнями комп'ютерними програмами – це одночасно і кінцева мета навчання, і складова певної інформаційної технології, яка застосовується для розв'язання реальних життєвих задач, і звичайно ж необхідний елемент інформаційної культури сучасної людини.

Потужність сучасних обчислювальних систем сконцентрована в їх апаратному забезпеченні (АЗ) і вона може бути реалізована лише відповідним програмним забезпеченням (ПЗ). Тому навички освоєння нового та використання вже знайомого ПЗ знаходяться в центрі уваги й під час навчання в сучасному навчальному закладі, й під час виконання завдань на сучасному виробництві. Чим багатіший арсенал комп'ютерних програм, якими людина вільно володіє, тим швидше, точніше будуть отримані розв'язки багатьох навчальних та виробничих завдань.

Власний досвід викладання інформатики (Запорізький ліцей №105, Економіко-правничий коледж, Запорізький національний університет) дав змогу авторам звернути увагу на деякі методичні питання, що виникають на етапах знайомства та вдосконалення навичок роботи з ПЗ.

1. Навички користування комп'ютером, їх розвиток і вдосконалення

Можна зазначити, що останніми роками змінився зміст поняття «вміння працювати з комп'ютером». Беручи до уваги міркування, що кількість корисних програм у світі швидко збільшується та, що професіоналом (експертом) можна стати, як правило, з вузького кола питань, сьогодні можна «вміти працювати» лише з конкретними програмами або з певним класом програм.

Виходячи з того, що робота з комп'ютером полягає у використанні ПЗ, **рівні навичок користування комп'ютером** можна визначати певним орієнтовним переліком програм і ефективністю роботи з ними:

Базові навички роботи з комп'ютером передбачають:

- уміння працювати з операційною системою (ОС) – налаштування елементів інтерфейсу та володіння прийомами роботи з ним, використання файлової системи (створення, видалення, перейменування, копіювання, перенесення файлів), управління певними функціями ОС (пошук файлів);
- використання стандартних додатків;
- виконання установки та налаштування нового АЗ та ПЗ;
- розуміння можливостей архіваторів, антивірусних програм, браузерів Internet та володіння прийомами роботи з ними.

Розвинені навички передбачають при розв'язанні завдань впевнене застосування прикладного ПЗ для роботи з: сучасними текстами складної структури; комп'ютерною графікою; відцифрованим звуком і відео; комп'ютерними мережами; базами даних; електронними таблицями.

Професійні навички передбачають уміння ефективно працювати з певними класами ПЗ. Цей рівень вимагає кваліфікованого (експертного) знання конкретних програмних продуктів, а також наявності практичного досвіду. Ефективну роботу з програмами відрізняє висока продуктивність праці, використання прийомів автоматизації. Розвиток таких навичок відбувається тільки в практичній діяльності.

Цілком зрозуміло що наведена нами класифікація досить умовна та суб'єктивна (як і багато інших). Але безсумнівним є те, що на вищій щаблі можна потрапити лише поступово підкоряючи нижчі. Задача викладача донести цю думку до учнів, укріпити впевненість у собі, захопити справою вивчення визначених програм. У своїй педагогічній діяльності ми, наприклад, для розв'язання цієї задачі практикуємо метод експертів (вони готують повідомлення щодо закріплених за ними програм, консультують своїх співучнів), метод проектів (тематичні конкурси сайтів, електронних газет, програмних розробок та ін.). Такий підхід дає можливість зробити навчання цікавим, підсилити його мотивацію, розвивати й охопити корисною діяльністю учнів з різною підготовкою, зорієнтувати їх на колективні методи праці з розподілом обов'язків, підвищує загальну інформаційну культуру.

Важливим є питання як саме оцінювати рівень навичок користувача. У цьому питанні корисною буде знайома з підручників та скоригована практикою схема визначення рівнів навичок користування ПЗ:

Рівень навичок	Конкретні дії, що відповідають рівню
Розуміння	Обговорювати, складати опис, пояснювати, перекладати
Запам'ятовування	Давати визначення, називати, складати список
Застосування	Оперувати, ілюструвати, використовувати, малювати
Аналіз	Порівнювати, протиставляти, обчислювати, розв'язувати
Синтез	Складати, проектувати, створювати
Оцінка	Оцінювати, вимірювати

Розуміння – початковий рівень засвоєння. Дає можливість обговорювати концептуальні питання (умови використання програми, межі її застосування).

Запам'ятовування – наступний рівень засвоєння навчального матеріалу. Наприклад, щоб перевірити запам'ятовування, можна запропонувати перелічити кроки, необхідні для запуску певної програми, завантаження вже збереженого об'єкту або створення нового.

Застосування – можливість користування програмою в нових умовах.

Аналіз – потребує від користувача вміння порівнювати та протиставляти ідеї даної програми (як концепції) та інших.

Синтез – можливість використати програму для розробки нових ідей.

Оцінка – генеральна оцінка певної програми, результатів її роботи, результатів користування нею певним користувачем.

Для більш детальної оцінки рівнів користування ПЗ ми звертаємось по допомогу до критеріїв 12-бальної системи шкільного оцінювання та вимог державних стандартів освіти.

Прийоми освоєння нового ПЗ

Робота з комп'ютером – це завжди освоєння нового. Технології виробництва комп'ютерної техніки швидко прогресують. Як наслідок активно створюються програми, що дозволяють реалізувати нові технічні можливості.

Коли робота з новою програмою тільки починається, користувач часто не знає, що з'явиться на екрані через декілька секунд. Звичайно потрібний час, щоб навчитися застосовувати програму на початковому рівні. Ефективне використання багатьох програм досягається тривалою практикою.

Освоєння складних професійних програм – нелегка задача, але зрозуміло, що автори програм не намагаються спеціально створити труднощі для користувача і складність сучасних програм пояснюється тільки великою кількістю включених до них можливостей. Не всі ці можливості потрібні кожному користувачу, тому у багатьох випадках освоєння програм можна спростити, зосередившись на вивченні первинних, базових прийомів роботи.

Усі існуючі різноманітні програми мають чимало спільного. Це, перш за все, однотипний інтерфейс, що як правило має три компоненти:

- робоче поле (область, в якій виконується робота з даними);
- інструменти (спеціальні засоби для виконання простих або комплексних дій);
- елементи управління (засоби для налаштування інструментів програми).

При знайомстві з програмами вивчати в робочій області фактично нема чого, тому первинне вивчення незнайомої програми зводиться до двох питань: 1) вивчення дії інструментів програми; 2) вивчення елементів управління програмою (вони стандартизовані та в більшості програм повторюються).

Робота з довідковими засобами

Існуюча на сьогодні стандартизація інтерфейсу дозволяє звести освоєння незнайомої програми до двох прийомів.

- 1) Відшукати інструменти програми (як правило вони розміщені на панелях інструментів), визначити їх призначення та прийоми користування.
- 2) Проглянути елементи управління програмою (меню, командні кнопки), виділити серед них знайомі і незнайомі. Перевірити дію знайомих елементів і встановити призначення незнайомих.

З цього витікає, що при вивченні незнайомої програми може виникнути потреба в додатковій інформації. В більшості випадків така інформація вже вбудована в програму – це довідкова система. Вона містить повний набір інформації та рекомендації щодо роботи з програмою і дозволяє одержати необхідну довідку під час роботи.

Кожна стаття довідкової системи відкривається в окремому вікні і є, як правило, гіпертекстом. Терміни-посилання, виділяються в тексті статті певним кольором. Посилання можуть бути оформлені у вигляді кнопок переходу. Вікна довідкової системи зазвичай не мають рядка меню. Замість нього у верхній частині вікна розташовуються кнопки переходу.

2. Збирання інформації про програму

Документальний супровід

Практично до всіх програм додається електронна або друкована документація, які є джерелом корисної інформації про програму.

Знайомство з програмою починається з **інформаційних екранів** установки. З них можна дізнатися про призначення програми, її можливості.

Друкowana документація це, як правило, досить містке керівництво. Його об'єм часто зводить нанівець бажання уважно його прочитати. Проте, у разі труднощів, – це одне з вичерпніших джерел необхідної інформації.

У багатьох випадках додаткова довідкова інформація до програми надається у вигляді текстових файлів, що входять до складу дистрибутивного комплексу. Традиційно такі файли мають ім'я README, яке походить від англійського «прочитай мене». Файл README містить інформацію про установку програми, доповнення і уточнення до друкованої керівництва, а також іншу інформацію, яку автори програми визнали потрібним опублікувати.

Довідкова і навчальна література

Робота з ПЗ може викликати труднощі в початківців і навіть у досвідчених користувачів. Майже за будь-якою відомою програмою можна знайти декілька книг, розрахованих на різний рівень підготовки читача. Автори цих книг звичайно є експертами у відповідній області, вміють провести малодосвідченого користувача через всі етапи навчання. Значна частина комп'ютерної літератури – перекладна, але якщо є можливість вибору, то слід віддавати перевагу книгам вітчизняних авторів. Ці книги краще адаптовані для вітчизняного читача, їх автори не обмежуються переказом офіційної інструкції до програми і діляться корисними порадами і практичним досвідом.

Інформація з Internet

У перелічених «традиційних» інформаційних джерелах не завжди можна одержати потрібні відомості про програму. Тоді користуються Internet. За допомогою пошукової системи (під час навчання ми звертаємо увагу учнів, як правило, на Google, Yandex, Rambler) неважко знайти Web-сторінку, присвячену потрібній програмі (серії програм). На першому місці серед них **сайти компаній-виробників і окремих авторів**. Вони містять відомості про версії програми; «заплатки» (patch), пов'язані з доробкою програми та виправленням помилок; посилання на інші ресурси обраної тематики; можна знайти безкоштовні, умовно-безкоштовні, демонстраційні, пробні версії програм.

Всесвітнє павутиння зберігає чимало архівів, в яких накопичуються відповіді на питання, що часто виникають під час користування різними поширеними комп'ютерними програмами. Їх позначають англійським скороченням **FAQ** (Frequently Asked Questions – питання, що часто ставляться). В них можна знайти відомості про прийоми виконання типових операцій, та вирішення характерних проблем.

Число існуючих Web-сторінок у світі досягає мільйонів, практично з будь-якого питання знаходяться ентузіасти, що створили **особисті Web-сторінки** для їх освітлення. Автори сторінок, присвячених одній програмі, зазвичай знайомі із сторінками інших авторів, що займаються тією ж темою, і, як правило, встановлюють на своїх сторінках посилання, дозволяючи продовжити пошук.

Щодо розміщення інформації на Web-сторінках, як правило, автор сам вирішує, про що він хоче розказати своїм читачам. В той же час, інколи вкрай необхідно власноруч поставити питання. В Internet є служби, що дозволяють звернутися до всього світу і одержати відповідь. 1) **Дискусійні форуми** (Chat-конференції) – надають відвідувачу можливість поставити питання або відповісти на повідомлення, залишене іншим відвідувачем, прочитати хроніку обговорення певного питання; 2) **Телеконференції** (групи новин) спеціально призначені для збору повідомлень, пов'язаних з обговоренням якої-небудь теми.

Спеціалізованою послугою електронної пошти є **списки розсилки**. Підписавшись на одному з Web-вузлів на список розсилки, присвячений тій або іншій програмі, можна щодня одержувати електронною поштою всі питання та відповіді, які надходять від усіх учасників цього списку розсилки.

3. Методичні прийоми ознайомлення з новими програмами

Як навчитися працювати з новою програмою? Більшість педагогів віддає перевагу інтерактивному методу ознайомлення. Він витікає з принципу: **щоб навчитися працювати, слід працювати.**

Перший прийом – Запуск програми. При установці програми створюються нові пункти Головного меню. Тоді для запуску слід скористатися одним з таких пунктів. Якщо програма не створила спеціального пункту в Головному меню, то до неї слід віднести з підозрою, бо вона може некоректно поводитися з ОС.

Якщо запуск такої програми все ж таки потрібен, слід знайти і відкрити папку, що містить файли програми. Потім треба запустити виконуваний файл програми, який зазвичай має розширення імені EXE та найбільш відповідає назві програми. Після того, як програму вдалося успішно запустити, слід створити для неї ярлик. У разі необхідності можна налаштувати властивості ярлика так, щоб надалі запуск програми не викликав проблем (наші учні тренуються за цією схемою встановлювати середовища програмування).

Другий прийом – Аналіз інтерфейсу програми. Отже, запущена незнайома програма. Перш ніж що-небудь робити, слід уважно дослідити вікно програми, його основні елементи, виділяючи знайомі й незнайомі. Всі програми перед розповсюдженням налаштовуються так, щоб з ними можна було розпочати роботу відразу. Звичайно, надалі кожен має можливість змінити параметри налаштування. Тому елементи вікна програми, які з'являються в перший момент, найбільш важливі і, скоріш за все, саме вони найбільш часто використовуватимуться в роботі з програмою.

Рядок заголовку містить назву програми та ім'я відкритого документа. В деяких випадках тут може з'явитися ім'я файлу документа, що дозволить визначити тип документів, з якими працює програма.

Під рядком заголовка як правило розташовується рядок меню. В більшості програм він містить стандартні пункти (Файл, Правка, Вид). Допоміжні меню, що відкриваються при виборі цих пунктів, дозволяють швидко визначити як стандартні межі, так і особливості запущеної програми.

Нижче рядка меню розташовуються панелі інструментів, які зазвичай схожі на аналогічні панелі програм того ж класу. Панелі інструментів часто містять кнопки стандартного призначення, що використовують в багатьох програмах.

Далі слід звернути увагу на робочу область вікна. Її вид та розміри можуть багато розповісти про тип документів, для роботи з якими призначена програма. Наприклад, наявність у вікні документа окремих кнопок управління розміром вікна зазвичай говорить про те, що програма здатна одночасно працювати з декількома вікнами. Якщо документ не візуалізується, це ознака того, що така дія не потрібна (наприклад, програма призначена для роботи зі звуком).

Третій прийом – Визначення напряму використання програми. Деяке уявлення про це ми отримуємо ще під час установки, але зустрічаються випадки, коли такої попередньої інформації немає.

Якщо зробити висновки про призначення програми в результаті аналізу її вікна не можливо, краще звернутися до документації. Довідкова система, як правило, містить спеціальний розділ, що описує призначення програми.

Якщо документації немає, слід уважно вивчити вікно програми. Вид і зміст пунктів рядка меню, кнопки панелей інструментів та інші елементи управління зазвичай дозволяють зробити висновок про призначення програми.

Наприклад, якщо рядок меню містить пункт Формат, значить це – редактор невідомого призначення і треба з'ясувати, з якими типами документів він дозволяє працювати. На це ж указує панель інструментів, що містить кнопки для вставки об'єктів. Накопичення досвіду роботи і знайомства з програмами різних класів дає змогу навчитись швидко визначати призначення програм.

Якщо спроби зрозуміти призначення програми не дали результату, слід поставити собі питання: «Чи варта ця програма того, щоб її освоювати?». Якщо автори програми не визнали потрібним або не зуміли створити дружній інтерфейс і зрозумілу довідкову систему, то чи можна розраховувати на те, що інші елементи програми зроблені краще? (Це правило обов'язково враховують наші учні - учасники конкурсів власних програмних розробок).

Четвертий прийом – Дослідження рядка меню. У багатьох додатках ОС більшість операцій можна виконати декількома способами. Для цієї мети можна використовувати команди рядка меню, кнопки панелі інструментів, контекстні меню, комбінації клавіш клавіатури.

Особливий інтерес представляє рядок меню. Ним не завжди зручно користуватися в реальній роботі, але при дослідженні незнайомої програми цей елемент управління корисний (наприклад, у додатках Windows діє принцип: все, що можна зробити в програмі, можна зробити засобами рядка меню).

П'ятий прийом – Аналіз форматів документів, що використовуються програмою. Наступний крок дослідження програми полягає в тому, щоб визначити формати документів, з якими вона може працювати. Програма, що заслуговує на увагу, повинна мати можливість роботи хоча б з декількома із загальнопоширених типів документів. Це важливо, щоб передавати готові документи іншим додаткам або імпортувати в програму документи, підготовлені іншими програмами.

Що вважати загальнопоширеним, залежить від авторів програми, тому деякі програми цілком можуть використовувати унікальні формати даних. Але, наприклад, текстовий редактор, що записує текст у своєму власному форматі і не розуміє жодного з широко поширених форматів, є безсумнівно незручним.

Визначити формати документів, з якими програма може працювати досить легко (команда Файл/Відкрити). Діалогове вікно відкриття файлу містить список типів файлів, формати яких програма «розуміє» й тому обробляє. Треба розкрити і проглянути цей список, знайти в ньому знайомі типи файлів. Це дозволяє визначити сумісність нової програми з тими, які вже знайомі.

4.Методичні прийоми освоєння нових програм

На етапі ознайомлення з програмою з'ясовується структура її вікна, склад інструментів і елементів управління, у загальних рисах встановлюється її призначення. Наступний етап полягає в освоєнні програми на рівні користувача. За наслідками цього етапу програма може стати постійним інструментом.

Перший прийом – Знайомство з прикладами, шаблонами, зразками документів. Для того, щоб оцінити можливості нової програми, краще за все скористатися наявними зразками документів, створених за її допомогою. Більшість програм при установці копіюють на диск файли зразків або шаблонів, призначених для швидкого створення певних документів.

Починати краще зі створення нової папки, в якій зберігатимуться документи, створені або відредаговані за допомогою програми, що вивчається. Це захистить комп'ютерну систему від руйнації після експериментів.

Редагування готового документа новими засобами – корисний, але не єдиний експеримент. Можна спробувати визначити, яким чином сформовані окремі деталі документа. Якщо є документ, створений за допомогою іншої програми, можна спробувати відтворити його новими засобами.

Шаблони документів (якщо вони є в новій програмі), відіграють особливу роль. Їх можна прийняти за основу майбутніх документів, а застосовані елементи оформлення можуть дати добре уявлення про можливості програми.

Другий прийом – Експерименти з пробними документами. Після знайомства з наявними документами, можна створити документ самостійно. Починати роботу слід із

запису порожнього документа в спеціально відведену для експериментів папку. Мета експерименту – перевірка дії команд редагування.

Під час експериментів не слід безладно переходити від однієї кнопки до іншої і від одного пункту меню до іншого. Під час дослідження не потрібне створення реального документа. Досить послідовно перевіряти доступні команди, стежачи за тим, до чого приводить їх вживання.

Одна з перших задач дослідження – навчитися виділяти об'єкти, з якими працює програма. Можливо, це робиться одинарним або подвійним натисненням кнопки мишки. Можливо, навколо об'єкту треба обвести контур за допомогою мишки. Слід також перевірити дію правої кнопки мишки, склад контекстного меню, що відкривається при цьому.

Якомога раніше слід відшукати команду, що скасовує дію попередньої команди, або команду, що скасовує всі попередні дії. Команда відміни операцій дозволяє не втрачати логіку дослідження і просуватися вперед.

Експерименти дозволяють визначити питання, з якими можна звернутися до довідкової системи або до засобів пошуку.

Третій прийом – Порівняння програм з аналогами. Найшвидший спосіб освоєння нових програм полягає в порівнянні їх з раніше вивченими аналогами. Як правило, всі програми одного класу мають більше спільних рис, ніж відмінностей. Знаходячи їх, можна або швидко освоїти нову програму, або зрозуміти, що користуватися нею не варто.

Наприклад, будь-які текстові процесори повинні мати можливість здійснювати одні й ті ж операції форматування тексту. Способи їх здійснення теж повинні бути близькими. Задача зводиться в першу чергу до того, щоб з'ясувати, де в новій програмі розміщуються давно знайомі команди.

Так, знайомство з програмою Internet Explorer компанії Microsoft дозволить швидко розібратися з програмою Firefox, розробленої в рамках проекту Mozilla, яка також призначена для організації роботи з Internet-ресурсами.

Четвертий прийом – Налаштування програми як засіб аналізу. Цікавий підхід до освоєння програми полягає в зміні її параметрів. Річ у тому, що варіанти налаштування програми відображають її можливості. А з'ясування того, які параметри можна змінити, дозволяє з'ясувати, на що здатна програма. При цьому змінювати параметри не обов'язково. Досить відкрити відповідне діалогове вікно і подивитися, які параметри можна регулювати.

Налаштування програми зазвичай здійснюється через пункти меню Вид, Сервіс, Параметри. Діалогові вікна налаштування параметрів програми зазвичай містять вкладки, призначені для різних груп параметрів. Оцінити їх призначення та важливість можна лише проглянувши їх всі.

Питання безпеки використання нових програм

Новостворених програм чимало. Певну їх частину відносять до програм сумнівного походження і спроба їх використання може надовго запам'ятися тим, скільки часу піде на усунення наслідків.

Надійними (відносно) можна вважати програми, створені крупними або відомими фірмами-виробниками і придбані в офіційному порядку. Установка такої програми на комп'ютер і спроба зайнятися її вивченням навряд чи приведе до яких-небудь неприємних наслідків. Видалення такої програми теж, як правило, відбувається коректно.

Умовно-безкоштовні програми, що додаються на CD до періодичних видань, представляють більш небезпечну категорію. В ході установки, перегляду і подальшого видалення цих програм може порушуватися працездатність інших програм або всієї ОС. Після їх видалення на жорсткому диску нерідко залишаються «загублені» невидані файли, які займають місце і знижують продуктивність роботи комп'ютерної системи. Тобто користування ними вимагає регулярної чистки жорсткого диску і періодичного перевстановлення ОС.

Особливу небезпеку представляють програми, одержані з невідомих і сумнівних джерел (від друзів, знайомих, з безкоштовних Internet-ресурсів). Підходити до таких програм слід з обережністю, утримуючись від їх використання з цікавості, обов'язково перевіряючи їх на відсутність вірусів. А файли, одержувані електронною поштою від невідомих доброзичливців, повинні знищуватися негайно без перегляду і оцінки їх корисності.

Висновки

Розглянуті питання освоєння програмного забезпечення ПК є актуальними для викладання інформатики як з точки зору підготовки інформаційно-культурних компетентних працівників, так і внаслідок державної політики використання програмного забезпечення без порушень прав виробника цього ПЗ. При цьому організація контролю за освітніми результатами ведеться нами в таких напрямках:

Експрес-перевірка засвоєння матеріалу на протязі навчального заняття. Застосовуються такі типи завдань (залежно від характеру розумової діяльності, що необхідна для отримання відповіді):

1. Завдання на розуміння: переклади (переклад пояснення в більш значущу форму або переклад з наочної форми у вербальну), інтерпретації (вміння інтерпретувати пояснення є свідомством розуміння), порівняння (виокремлення загального та часткового є свідомством розуміння концепції), екстраполяція (перевірка правильності сприйняття інформації та можливості робити висновки на основі отриманих знань за допомогою конструкцій “що буде якщо ... спрогнозуйте наслідки”).
2. Завдання на запам'ятовування (це закриті запитання, що вимагають від учня повторити певні факти).
3. Завдання на застосування (відкриті запитання на відбір необхідної інформації для розв'язання практичної задачі).
4. Ситуативні завдання (в них може бути присутня надлишкова інформація, необхідно проаналізувати та запропонувати можливі розв'язки).

Тестування. Спираючись на основні положення теорії тестування (В. Аванесов) нами розробляються комплекти тестових завдань та виконується відповідна автоматизована процедура перевірки рівня засвоєння навчального матеріалу. При цьому ми широко застосовуємо як вже відомі програми (тест Шестопалова, HyperTest), так і власні учнівські розробки.

Реферати та усні доповіді. Для постійного тренування загально-навчальних навичок пошуку, реферування та усного повідомлення інформації, учні згідно плану та власних уподобань обирають тему доповіді, таку що стосується питання застосування певної програми до розв'язування конкретної прикладної задачі. При цьому якість підготовки матеріалів та самого виступу оцінюється за критеріями, визначеними разом вчителем та учнями. Окрім стандартних вимог до такої форми роботи (план, обмежений обсяг, посилання на літературу, оформлення згідно сучасних правил), учні мають надати додатково глосарій (ключові слова з поясненням їх змісту) та перелік контрольних запитань, якими можна скористатися іншим для перевірки рівня ознайомлення з цим матеріалом.

Останнє, на що хотілося б звернути увагу, це обов'язкове ознайомлення учнів з цілями тієї чи іншої роботи з певною програмою, як одне з найважливіших чинників отримання якісних знань та навичок. Вважаємо, що з практичної точки зору слова, що найбільш підходять для опису таких цілей (дії, що можна перевірити) це: писати, говорити, створювати, визначати, конструювати, додавати, вирішувати, обчислювати, застосовувати, вибирати. А слова: розуміти, оцінювати, знати, розвивати, вивчати – як правило не підходять для опису цілей, тому що ці дії важко перевірити. Коли учень знає навіщо йому освоювати роботу з певною програмою та й ще згоден, що вона йому дійсно

знадобиться, процес отримання та закріплення навичок роботи з нею буде значно результативніший.

Таким чином, учень або студент, підготований з урахуванням зазначених нами методичних прийомів навчання, стає адаптованим до сучасних вимог життя та, звісно, активним суб'єктом будь-якого сучасного локального або єдиного освітнього середовища.

Література

1. *Винарик Л.С., Береуцкий Я.Г., Щедрин А.Н.* Информационная культура в современном обществе: Учебное пособие. – Донецк: Институт экономики промышленности; ДИЭХП, 2003. – 322 с.

2. *Зимняя И. А.* Ключевые компетенции – новая парадигма результата образования // Высшее образование сегодня, 2003. - №5. – с.34-42.

3. *Иванова Т. В.* Компетентностный подход к разработке стандартов для 11-летней школы: анализ, проблемы, выводы // Стандарты и мониторинг в образовании. – 2004. - №1. – с.16-20.

4. *Мозолин В. В.* Компетентностный подход в обучении информатике: возможности и проблемы // Информатика и образование, 2005. – №2. – с.115-116.

5. Компетентнісний підхід у сучасній освіті: світовий досвід та українські перспективи: Бібліотека з освітньої політики/ Під заг. ред. О.В. Овчарук. – К.: «К.І.С.», 2004. – 112 с.