

## НА ДОПОМОГУ ПЕДАГОГУ

Проведене нами дослідження дозволяє зробити висновок про те, що формування елементів художнього дизайнерського мислення не може бути короточасним процесом чи будуватися фрагментарно. Для цього нами розроблена багаторівнева програма, розрахована на весь період навчання дитини в школі. Вона успішно застосовується у Красноперекопській ЗОШ №5 (Автономна Республіка Крим).

Постановка та розв'язання дизайнерських проблемних ситуацій на основі спеціальних знань, розкриття в процесі навчання дизайнерської якості начала речей, залучення дітей до активних самостійних спостережень природи, оточуючого предметного середовища сприяли активізації творчого потенціалу особистості та переходу від алгоритмічних до евристичних способів розв'язання задач.

Проведені спостереження в цілому підтвердили правильність обраного шляху щодо формування елементів дизайнерського мислення в учнів шкільного віку. Серед перспективних питань, які потребують подальшого дослідження даної проблеми, найбільш актуальною ми вважаємо розробку системи підготовки вчителів для цілеспрямованої роботи щодо формування елементів дизайнерського мислення учнів.

1. Каган М.С. Человеческая деятельность (опыт систематического анализа) – М.: Политическая литература, 1984. – С.23-33.
2. Калинина Г.П. Использование элементов дизайна на уроках труда. //Начальная школа. – 1989. – №4 – С.9-14.
3. Развитие у школьников конструкторского творчества на уроках труда. Метод. рекомендации под ред. Качнева В.И. – М.: Просвещение, 1997. – С.14 -25.
4. Игнатьев Е.У. Воображение и его развитие в творческой деятельности. – М.: Просвещение, 1985. – С.67-91.

***Зайцева Т.В.***

### **МЕТОДИЧНІ РЕКОМЕНДАЦІЇ ДО ВИКЛАДАННЯ РОЗДІЛУ "ВВЕДЕННЯ ДО ПРОГРАМУВАННЯ"**

Програмування – одна з традиційних тем в курсі інформатики, місце і вага якої в програмі даної дисципліни постійно змінюється. Найбільш істотний перегляд цього питання відбувся у зв'язку з перенесенням інформатики із старших класів у базову школу і з розвитком комп'ютерних технологій. Останні видання програми шкільного курсу інформатики і обчислювальної техніки наголошують на вивченні курсу користувача, хоча програмування займає ще значне місце.

НА ДОПОМОГУ ПЕДАГОГУ

Новим напрямком у розвитку розділу програмування є введення до курсу інформатики вивчення об'єктно-орієнтованих мов програмування. Вчитель інформатики може в залежності від наявності комп'ютерної техніки і профільного напрямку школи викладати структурну чи об'єктно-орієнтовану мову програмування. Як об'єктно-орієнтовану мову в шкільній практиці можна використовувати розширення мови Паскаль – середовище візуального програмування Delphi-Pascal і Visual Basic [1].

**Мета** даної статті – характеристика недоліків, що зустрічаються в шкільній практиці викладання програмування та методичні рекомендації щодо поліпшення процесу навчання основам програмування.

Розглянемо питання – з якою метою потрібно вивчати програмування в загальноосвітній школі в базовому курсі інформатики?

1. З метою фундаменталізації курсу інформатики. Одним з фундаментальних принципів є принцип програмного керування роботою комп'ютера. Зрозуміти його неможливо, не знаючи, що таке програма для ЕОМ. Таким чином, знайомство з програмуванням є елементом загальноосвітнього змісту інформатики.

2. З погляду профорієнтаційної функції предмета. Будь-який шкільний предмет повинен поряд з освітньою, виховною і розвиваючою функціями виконувати ще і профорієнтаційну. Програмування є професійною галуззю діяльності, дуже важливою, сучасною і престижною.

Залежно від мети й обсягу курсу наповнення методичної системи вивчення цього розділу може бути різним. Класичним, універсальним і найбільш розповсюдженим у шкільній практиці є процедурний напрямок. Процедурні мови найбільше підходять для навчальної мети.

Процес вивчення і практичного засвоєння програмування можна розділити на 3 етапи:

1. Вивчення методів побудови обчислювальних алгоритмів.
2. Вивчення мови програмування.
3. Вивчення і практичне засвоєння конкретної системи програмування.

Питання першої частини можна розглядати при вивченні розділу "Алгоритмізація". У розділі базового курсу "Введення до програмування" необхідно продовжувати ту ж структурну лінію, що була закладена в алгоритмічному розділі, а саме

## НА ДОПОМОГУ ПЕДАГОГУ

послідовність вивчення матеріалу повинна визначатися наступними принципами:

- *від простого до складного*: поступове ускладнення задач;
- *новизна*: кожна задача вносить деякий новий елемент знань (нова команда, новий прийом);
- *наслідування*: наступна задача потребує використання знань, які були отриманні при розв'язуванні наступних задач [2].

Знайомство з основами програмування йде двома паралельними напрямками: знайомство з конструкціями мови програмування та з типами даних мови програмування.

Зупинимося на введенні поняття величини та характеристики величин. У кожній мові програмування існує своя концепція типів даних, своя система типів. Однак у будь-яку мову входить мінімально-необхідний набір основних типів даних, до якого відносяться: цілий, дійсний, символний і логічний. З типом величини пов'язані чотири його характеристики:

- 1) множина припустимих значень;
- 2) множина операцій над даними;
- 3) множина функцій, визначених над даними цього типу;
- 4) множина визначених відношень.

Дану схему слід використовувати при знайомстві учнів з характеристиками будь-якого типу даних.

З методичної точки зору знайомство учнів з типами величин краще розподілити на окремі уроки і вводити нові типи поступово, по мірі необхідності, в наступній послідовності: Integer, Real, Char, Boolean. Наприклад, після знайомства з цілим типом даних і розв'язування деякої кількості задач, вчитель змінює умову задачі таким чином, щоб учні зіштовхнулися з проблемною ситуацією: розв'язати дану задачу для діапазону цілих чисел не можливо. Вчитель вводить поняття типу даних Real. Не слід поспішати з знайомством характеристик логічного типу. Введення нового типу повинно викликатися практичною необхідністю, тобто алгоритмом розв'язування конкретної задачі.

Не слід забувати, що існує так зване розширення типів даних. Наприклад, знайомство з групою цілих типів Borland Pascal (LongInt, ShortInt, Byte) повинно йти на інтуїтивному рівні під час розв'язування задач. При розв'язуванні задач на знаходження факторіалу натурального числа, учні зіштовхнуться з проблемою, коли задане натуральне число більше за 7, то діапазону цілих чисел недостатньо – вводимо розширення цілого типу даних.

## НА ДОПОМОГУ ПЕДАГОГУ

Протягом цих уроків учні можуть складати таблицю, в яку заносяться характеристики різних типів даних (див. табл.1). Спочатку учні користуються таблицею при розв'язуванні задач, поступово інформація відкладається в їх пам'яті і необхідність в таблиці відпадає.

Таблиця 1

<i>Тип</i>	<i>Множина значень</i>	<i>Операції</i>	<i>Функції</i>	<i>Відношення</i>
цілий				
дійсний				
логічний				
символьний				
масиви				
рядки				

При знайомстві з функціями, які можна використовувати для конкретного типу даних слід звернути увагу учнів на різниці між поняттями:

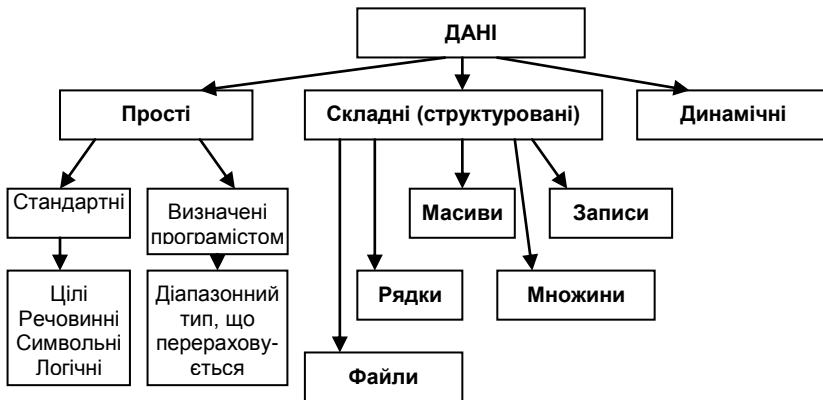
- функції, що визначенні над даним типом (в цьому випадку аргумент і результат виконання функції належать даному типу);
- функції, результат виконання яких належить даному типу (тип аргументу немає значення);

Останні функції зручно називати функціями перетворення типів, до них відносяться такі функції, як відкидання дробової частини (Trunc) та округлення числа (Round). Для них аргументи – дійсні числа, а результат – цілі числа. До цієї ж групи зручно віднести і функції Char (знаходження порядкового номеру символу) та Ord (по номеру визначається символ).

На останньому уроці, після знайомства з усіма основними типами даних, для узагальнення і систематизації знань учнів слід провести класифікацію типів даних. Для цього можна запропонувати наступну схему, де з деякими типами даних учні ще не знайомилися, але вони отримують загальну концепцію типів даних мови програмування Паскаль [3].

В даній схемі використовувався варіант класифікації даних за структурою. Дані поділяються на прості і структуровані. Для простих величин (їх ще називають скалярними) справедливе твердження: одна величина – одне значення. Для структурованого: одна величина – декілька значень.

Дії над величинами, обумовлені алгоритмом, ґрунтуються на наступній ієрархії понять: операція – вираз – команда (оператор) – система команд.



**Операція** – найпростіша закінчена дія над даними.

**Вираз** – запис в алгоритмі, що визначає послідовність операцій для обчислення деякої величини.

**Команда** – розпорядження виконавцю виконати деяку закінчену дію.

**Система команд** – повний список команд конкретного виконавця.

Далі зупинимося на двох вузлових поняттях програмування: змінна й оператор присвоювання.

Процес розв'язування задачі – це процес послідовної зміни значень змінних. Змінна одержує шуканий результат у процесі присвоювання. На рівні машинних команд присвоювання – це занесення в чарунку пам'яті, відведеної під змінну, значення, що визначенні в результаті виконання команди. До таких команд відносяться: присвоювання, процедура введення даних і передача параметрів підпрограмам.

Досвід показує, що в багатьох випадках нерозуміння програмування деякими учнями походить від нерозуміння змісту та механізму дії операції присвоювання. Тому особливу увагу на перших порах треба звертати на це питання.

<Змінна> := <вираз>

Механізм дії:

- 1) обчислити значення виразу;
- 2) присвоїти (покласти) отримане значення змінній.

Звернути увагу учнів, що дана команда виконується праворуч ліворуч. Далі обов'язково слід пояснити зміст команди:

$X := X + 1$

Пояснити задачу обміну значеннями двох змінних.

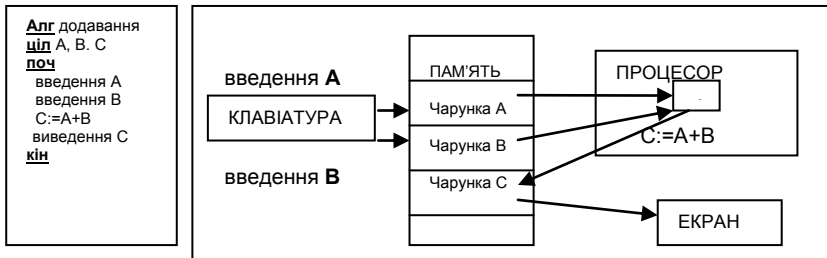
## НА ДОПОМОГУ ПЕДАГОГУ

$x:=y; y:=z; z:=x$

Під введенням у програмуванні розуміється процес передачі даних із зовнішнього пристрою до оперативної пам'яті. По команді введення робота процесора переривається і відбувається чекання дій користувача; користувач набирає на клавіатурі дані і натискає <Enter>; значення, які були введені присвоюються змінним.

Для візуалізації процесів, що відбуваються усередині комп'ютера при виконанні даної команди, можна використати схему 1 [2].

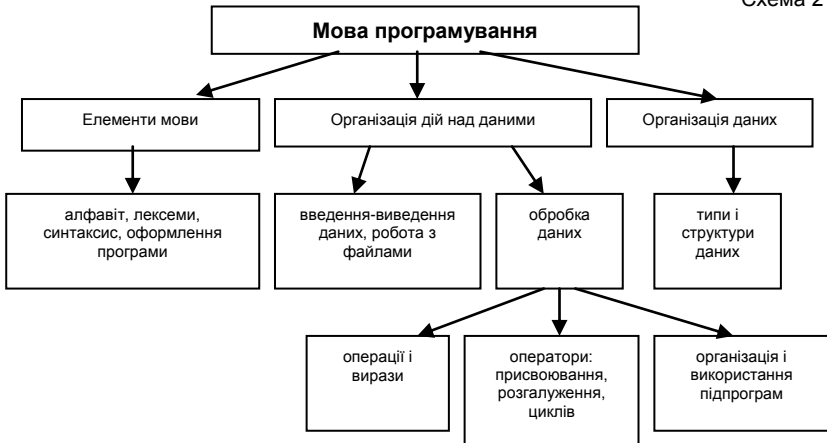
Схема 1



Вчителям не слід забувати, що ефективним методичним засобом, що дозволяє досягти розуміння програмування, є ручне трасування алгоритмів.

Методика вивчення мови програмування спирається на структуру самого об'єкта вивчення – мови програмування, яку можна представити на схемі 2:

Схема 2



## НА ДОПОМОГУ ПЕДАГОГУ

Основний метод вивчення мови програмування – демонстрація використання конструкцій мови на прикладах простих програм з короткими коментарями. Деякі поняття досить сприймати учням на інтуїтивному рівні. Наочність такої мови як Паскаль полегшує це сприйняття. Розумінню матеріалу допомагає аналогія між Паскалем і алгоритмічною мовою. Для виконання учнями нескладних самостійних завдань на першому етапі досить діяти методом "за зразком".

Проблему як зв'язати вивчення методів побудови алгоритмів роботи з величинами і мови програмування можна вирішити 2 варіантами:

1) спочатку розглядаються різноманітні алгоритми, для опису яких використовуються блок-схеми й алгоритмічна мова, а потім – правила мови програмування, способи перекладу вже побудованих алгоритмів у програму на цій мові;

2) алгоритмізація і програмування освоюються паралельно.

Бажано, щоб учні як раніше одержали можливість перевіряти правильність своїх алгоритмів, працюючи на комп'ютері. Але навіть при використанні комп'ютера, спочатку, рекомендується не відмовлятися від ручного трасування.

Отже, протягом вивчення розділу "Введення до програмування" вчитель повинен загалом характеризувати існуючі напрямки в програмуванні, розглядати сучасні підходи до розробки програм, давати класифікацію мов програмування й описувати для розв'язування якого типу задач підходять різні мови програмування.

1. Міністерська програма з шкільного курсу інформатики.
2. Семакіна І.Г., Шеїна Т.Ю. Преподавание базового курса информатики в средней школе. Методическое пособие. – М.: Лаборатория Базовых Знаний, 2000. – 496с.: ил.
3. Львов М.С., Співаковський О.В. Основи алгоритмізації та програмування. Навчальний посібник – Херсон, 1997. – 122с.
4. Верлань А.Ф., Алатова Н.В. Інформатика: Підручник для учнів 10-11 кл. серед. загальноосв. шк. – К.:Форум, 2000. – 233 с.

**Коваленко А.Г.**

## ФОРМУВАННЯ ПОЧУТТЯ ГУМОРУ У ШКОЛЯРІВ НА УРОКАХ МОВИ ТА ЛІТЕРАТУРИ

Емоції і почуття мають дуже велике значення в житті людини. Вони збагачують відображення людиною об'єктивної дійсності і стають у зв'язку з її потребами та інтересами важливими спонуканнями до діяльності.