

Міністерство освіти і науки України  
Полтавський державний педагогічний університет  
імені В.Г. Короленка  
Кафедра математичного аналізу та інформатики

Барболіна Т.М.

**ШКІЛЬНИЙ КУРС  
ІНФОРМАТИКИ ТА  
МЕТОДИКА ЙОГО  
ВИКЛАДАННЯ**

*Частина 2  
Часткова методика*

*Навчальний посібник*

Полтава–2008

Барболіна Т.М. Шкільний курс інформатики та методика його викладання: навчальний посіб. / Барболіна Т.М. – Полтава, 2008. – Ч.2. Часткова методика. – 116 с.

Рецензенти: *А.А. Роскладка* — кандидат фізико-математичних наук, доцент кафедри математичного моделювання та соціальної інформатики Полтавського університету споживчої кооперації України  
*О.П. Губачов* — кандидат фізико-математичних наук, доцент кафедри математичного аналізу та інформатики Полтавського державного педагогічного університету імені В.Г. Короленка

Ухвалено до друку вченою радою Полтавського державного педагогічного університету імені В.Г.Короленка

протокол № 4 від 31.10.2008

У посібнику висвітлюються особливості викладання основних тем шкільного курсу інформатики. Розглядаються питання формування основних понять шкільного курсу інформатики, специфіка ознайомлення з апаратною і програмною складовими інформаційної системи, методика викладання основ алгоритмізації та програмування.

## *Лекція № 9*

# **МЕТОДИКА ОЗНАЙОМЛЕННЯ З БАЗОВИМИ ПОНЯТТЯМИ КУРСУ ІНФОРМАТИКИ**

### **Методика ознайомлення з поняттям інформації**

Поняття інформації є ключовим поняттям, яке пов'язує інші теми шкільного курсу інформатики. Більше того, поняття інформації використовується в усіх виробничих та наукових сферах, тому конкретне тлумачення елементів, пов'язаних з поняттям «інформація», залежить від методів конкретної науки, мети дослідження, навіть досвіду чи психічного стану людини. Разом з тим побудова уроків цієї теми є методичною проблемою, до цієї теми найменша кількість літератури, до того ж викладення окремих питань у різних джерелах є досить суперечливим. Поняття «інформація» абстрактне, досить складне та багатогранне, строго кажучи, є неозначуваним у рамках інформатики. Тому його можна лише охарактеризувати, описуючи взаємодію з іншими поняттями інформатики, зміст даного поняття розкривається та уточнюється протягом досить тривалого часу вивчення курсу. Однак це не означає, що таке поняття не може бути предметом розгляду уже з перших уроків. Наприклад, таким самим абстрактним поняттям є поняття «натуральне число», яке не визначається протягом вивчення математики, але освоюється школярами з першого класу.

Ознайомлення учнів з поняттям інформації на перших уроках курсу (як це передбачено і навчальною програмою) дозволяє аргументовано розкрити зміст навчального предмета інформатики, ознайомити з його завданнями. Основні методи вивчення матеріалу: за логікою — індуктивний, за джерелами подання інформації — пояснювально-ілюстративний, за ступенем самостійності учнів — репродуктивний.

Вважається доцільним виділення таких етапів формування поняття інформації:

- 1) введення поняття інформації та повідомлення;
- 2) з'ясування взаємозв'язку між поняттями інформації та повідомлення;
- 3) формування уявлень про носії інформації;
- 4) ознайомлення із способами подання інформації;
- 5) розгляд питань про види інформації;
- 6) висвітлення оцінювання й вимірювання інформації, поняття шуму, взаємоперетворення інформації і шуму;
- 7) формування уявлень про кодування повідомлень, за допомогою яких передається інформація;
- 8) з'ясування властивостей інформації;
- 9) ознайомлення з інформаційними процесами.

Поняття інформації не визначається через простіші, вводиться конкретно-індуктивним способом за допомогою наочних, добре знайомих прикладів. Як суттєві ознаки поняття слід відзначити, що розглядаються принаймні дві системи, одна з яких породжує деяку сукупність сигналів, подразників, а інша — сприймає. При цьому несуттєвим є спосіб подання сукупностей повідомлень, якісний і кількісний склад систем.

Інформацію передають за допомогою повідомлень, які передаються за допомогою послідовності сигналів від джерела до приймача повідомлення. Слід наголосити на відсутності взаємно однозначної відповідності між інформацією і повідомленням. Особливої уваги також потребує поняття шуму, зокрема такі положення:

- 1) якщо повідомлення не несе корисної інформації, то воно несе шум;
- 2) можливість взаємного перетворення інформації та шуму.

З'ясування сутності зазначених понять вимагає значної кількості прикладів. Для підвищення інтересу учнів можна підготувати комп'ютерну презентацію, яка містила б необхідні малюнки, та запропонувати учням зробити висновки після її перегляду.

З поняттями носіїв повідомлень учні зустрічаються протягом освоєння всієї практичної частини шкільного курсу інформатики. Поняття носія повідомлення доцільно вводити

Приклад змісту навчальної комп'ютерної презентації з «Текстовий редактор» .....	94
Приклад план-конспекту уроку з теми «Табличний процесор» .....	103
Приклад завдань і критеріїв оцінювання практичної роботи з теми «Глобальна мережа Інтернет та її можливості» .....	110
Приклад практичних завдань для повторення курсу інформатики .....	112

## ЗМІСТ

<i>Лекція № 9</i> Методика ознайомлення з базовими поняттями курсу інформатики .....	3
Методика ознайомлення з поняттям інформації .....	3
Ознайомлення з поняттям моделі .....	7
Особливості формування поняття інформаційної системи ..	10
Самостійна робота студентів .....	14
<i>Лекція № 10</i> Методика ознайомлення з окремими видами програмного забезпечення .....	17
Особливості вивчення операційної системи .....	17
Особливості ознайомлення з операційною системою Windows .....	20
Ознайомлення з можливостями глобальної мережі Інтернет ..	22
Самостійна робота студентів .....	26
<i>Лекція № 11–12</i> Методика навчання прикладного програмного забезпечення загального призначення .....	32
Загальні особливості навчання прикладного програмного забезпечення загального призначення .....	32
Особливості вивчення графічного редактора .....	37
Особливості вивчення текстового редактора .....	40
Особливості вивчення табличного процесора .....	44
Особливості вивчення баз даних та СУБД .....	49
Самостійна робота студентів .....	53
<i>Лекція № 13–14</i> Методика навчання основ алгоритмізації та програмування .....	59
Методичні особливості навчання основ алгоритмізації .....	59
Методика вивчення систем візуального програмування .....	66
Навчальна алгоритмічна мова .....	67
Особливості вивчення окремих тем .....	69
Самостійна робота студентів .....	78
Список літератури .....	84
Додатки Приклади методичних матеріалів з різних тем шкільного курсу інформатики .....	87
Приклад інформатичного диктанту з теми «Інформаційна система» .....	87
Приклад тестового завдання з теми «Операційна система» ..	88
Приклад завдань практичної роботи з теми «Графічний редактор» .....	91

описово, звернувши увагу на поділ носіїв на довгоіснуючі та недовгоіснуючі. Слід окремо виділити носії, які використовуються при роботі з комп'ютером. Особливе значення має подання на довгоіснуючих носіях — письмо. Поняття носія повідомлення важке, особливо, якщо мається на увазі не матеріальний об'єкт, а, наприклад, хвиля, стан речовини тощо. Для його засвоєння доцільно запропонувати вправи на визначення носія повідомлення при різних способах їх подання, передавання і зберігання.

На наступному етапі доцільно розглянути способи та засоби подання і передавання повідомлень, одним із яких є мова. Існує досить багато різних мов — мови різних народів, мова глухонімих, мова сигнальників на кораблях, мова спілкування двох людей, які не знають рідної мови один одного. Поняття мови не обмежується випадком спілкування між людьми, воно використовується й у випадку порівняно високо розвинених форм спілкування між іншими живими істотами. Так, можна говорити про мову орієнтації бджіл, мову спілкування птахів, звірів (крики тривоги, заклик і загрози, різноманітні рухи, пози тощо). Залежно від інформаційних можливостей виділяють три типи мов. До першого типу відносяться природні (по-іншому — комунікативні) мови людського спілкування. Другий тип складають формалізовані мови, які характеризуються однозначністю змісту слів, які складають ці мови. До таких мов перш за все відносяться мови наукових дисциплін. Нарешті, третій тип — це формальні мови, для яких жорстко визначені правила побудови слів та речень мови, а також їх змістова інтерпретація. Найпростішою та найбільш широко вживаною формальною мовою є мова запису натуральних чисел за допомогою цифр. На її основі в початковій школі будується наступна формальна мова — мова арифметичних виразів. Потім будується мова алгебраїчних виразів і т.д. Формальною мовою є також мова програмування, яка вивчається в школі як елемент курсу інформатики. Проте ця мова не є першою формальною мовою, з якою доводиться мати справу школярам. І розуміння цієї обставини учнями дуже важливо з точки зору формування у них загальної інформаційної картини світу — адже нерідко само існування формальних мов пов'язуються тільки з програмуванням, тобто знов-таки з комп'ютерами.

При ознайомленні учнів зі способами подання повідомлень важливо також досягти розуміння ними того, від чого залежить подання повідомлень саме обраною мовою (швидке і надійне кодування повідомлення, зручність для адресата).

Розглядаючи види інформації, доцільно розглянути кілька класифікацій за різними основами: за способами сприйняття (візуальна, аудіальна, тактильна, нюхова, смакова), за формою подання (текстова, числова, графічна, музична, комбінована), за суспільним значенням (масова і спеціальна).

При висвітленні оцінювання й вимірювання інформації провідні українські методисти вважають доцільним говорити не про кількість інформації, а про величину повідомлень та відповідні одиниці вимірювання. Важливо розуміти, що інформація може оцінюватися людьми з різних точок зору – за змістом, за важливістю. Довжина повідомлення, яку можна визначити тривалістю передавання чи приймання, нічого не говорить про те, багато чи мало корисної інформації несе повідомлення, чи воно несе лише шум.

Ознайомлюючи учнів із кодуванням повідомлень, слід звернути увагу на те, що для подання й опрацювання повідомлень у комп'ютері використовують двійкові коди.

Під час розгляду властивостей інформації не слід намагатися дати найбільш повні та загальні відомості. Необхідно виділити лише ті сторони цього поняття, на які спиратиметься подальше вивчення матеріалу, і які будуть використовуватися учнями в їх практичній діяльності.

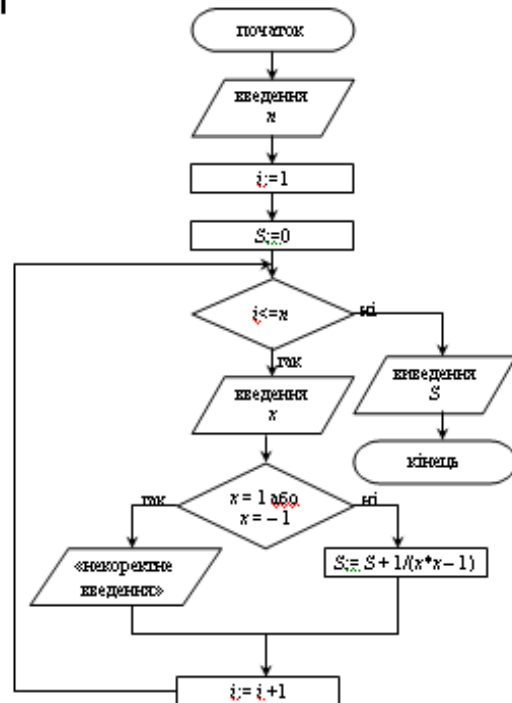
Особливе місце у вивчення теми відіграють питання про інформаційні процеси, адже інформатику нерідко визначають як науку про методи накопичення, передавання, зберігання інформації. Основними інформаційними процесами є: пошук — збирання — зберігання — передавання — опрацювання — використання — захист інформації. Специфіка інформаційних процесів вводиться на прикладах, чіткі означення не даються. Головною причиною є те, що в реальному інформаційному процесі нерідко є переплетеними одержання, перетворення, передача, використання інформації. Наприклад, при читанні книги має місце одержання інформації, але водночас відбувається перетворення інформації, її передача, організація зберігання.

```
program Calculate;
var x,s:real;
    i,n:integer;
begin
  writeln('Введіть кількість доданків');
  readln(n);
  for i:=1 to n do
  begin
    writeln('Введіть xi');
    readln(x);
    if (x=1) or (x=-1)
    then writeln('некоректне введення')
    else s:=s+1/(x*x-1)
  end;
  writeln('S=',s:1:3);
end.
```

## Можливий вигляд документу

Задача Обчислити значення виразу  $y = \sum_{i=1}^n \frac{1}{x_i^2 - 1}$ , якщо

кількість  $n$  доданків і самі числа  $x_i$  вводяться з клавіатури (якщо при деякому  $x_i$  вираз не має смислу, то повинно виводитися відповідне повідомлення і програма продовжує свою роботу).



Одним із основних питань є питання опрацювання інформації. Важливо, щоб учні розуміли, що люди у своїй діяльності постійно зустрічаються з необхідністю опрацювання інформації. Слід також підвести учнів до висновку, що комп'ютер, хоча і є одним із найефективніших засобів опрацювання повідомлень, мислити не може.

## Ознайомлення з поняттям моделі

Який би варіант отримання інформації не мав місце, інформація про об'єкт, явище або процес, що розглядається, буде очевидно неповною. Справа в тому, що будь-який приймач інформації, чи жива істота чи автоматичний пристрій, внаслідок своєї скінченності (скінченна кількість клітин мозку чи комірок оперативної пам'яті) не може відобразити *всю* інформацію про об'єкт, явище чи процес внаслідок нескінченності цієї інформації — адже будь-який реальний об'єкт, явище, процес має нескінченну множину властивостей чи проявів. Таке за необхідністю обмежене подання об'єкта, процесу чи явища називають його **інформаційною моделлю**. Структура інформаційної моделі — це набір істотних чинників та зв'язки між ними. **Одним** методисти вважають, що головна мета вивчення моделі пов'язана з розглядом етапів розв'язування задач на комп'ютері. Розуміння змісту цих етапів — одна з важливих цілей курсу інформатики, яка досягається поступово, у процесі вивчення учнями всього навчального матеріалу. У той же час моделювання як метод пізнання має значно ширше застосування, ніж розв'язування задач за допомогою комп'ютера, і модель може розглядатися як одне із базових понять інформатики, тому поняття моделі (як і формалізації) є чи не найголовніші змістові лінії курсу інформатики. З метою формування навичок формалізації доцільно пропонувати учням завдання, що вимагають побудови інформаційних моделей, не тільки при розгляді етапів розв'язування задач на комп'ютері, але й при вивченні інших розділів курсу (зокрема, прикладного програмного забезпечення загального і навчального призначення). При цьому необов'язково вводити означення моделі, розглядати класифікації тощо.

Поняття моделі пов'язано з поняттям об'єкта. Це поняття яке є не означуваним і розуміється як все те, на що спрямована увага людини, з чим людина працює фізично чи розумово. Для розрізнення об'єктів їм надають імена. Властивості об'єкта — це сукупність ознак, за якими його можна відрізнити від інших; кожна з властивостей характеризують параметри, які можуть набувати різних значень (слід розрізнити параметр та його значення).

Важливим є розуміння ланцюга «об'єкт моделювання — мета моделювання — модель».

### Основні цілі моделювання

1. Модель потрібна для розуміння будови об'єкта (протікання процесу, явища), яка його структура, властивості, закони розвитку та функціонування, взаємодія із оточуючим світом (понятійна модель).
2. Модель потрібна для того, щоб визначити найкращі способи керування об'єктом чи процесом (управлінська модель).
3. Модель потрібна для того, щоб прогнозувати прями та опосередковані наслідки функціонування чи розвитку об'єкта, процесу чи явища (прогностична модель).
4. Модель потрібна для організації процесу навчання чи самонавчання (навчальні моделі).

У літературі немає однаковості щодо означення моделі. Наведемо як приклад кілька означень із різних навчальних і методичних посібників.

«Модель — це прообраз, опис чи зображення якого-небудь об'єкта»[1].

«Модель — це опис істотних для поставленої задачі властивостей і закономірностей поведінки об'єктів, що забезпечують її розв'язання»[12]

«Модель — штучно створений об'єкт у вигляді схем, рисунків, логіко-математичних, знакових формул, фізичних конструкцій та ін., який деякою мірою характеризує досліджуваний об'єкт, безпосереднє вивчення якого неможливо або утруднено» [23].

«Модель — це відповідне цілям моделювання подання деякого об'єкта (процесу або явища) іншим об'єктом (процесом

#### 1. Внутрішня

- a. оперативна
- b. постійна
- c. кеш-пам'ять
- d. відеопам'ять
- e. реєстри процесора

#### 2. Зовнішня

- a. накопичувачі на магнітних дисках
  - i. НГМД
  - ii. НЖМД
- b. накопичувачі на оптичних дисках
  - i. CD
  - ii. DVD

### Завдання 2

*повторення роботи з об'єктами у середовищі текстового процесора, побудова блок-схем алгоритмів, складання описів програм мовою програмування високого рівня з використанням операторів циклу і операторів розгалуження*

Оформити у текстовому процесорі розв'язування поданої нижче задачі. Документ повинен містити:

- завдання
- блок-схему
- Паскаль-програму

Задача Обчислити значення виразу  $y = \sum_{i=1}^n \frac{1}{x_i^2 - 1}$ , якщо

кількість  $n$  доданків і самі числа  $x_i$  вводяться з клавіатури (якщо при деякому  $x_i$  вираз не має смислу, то повинно виводитися відповідне повідомлення і програма продовжує свою роботу введенням наступного числа).



## Приклад практичних завдань для повторення курсу інформатики

### Завдання 1

повторення основних понять з теми «Інформаційна система», основних дій з файлами і папками, створення архівів, роботи в текстовому процесорі: введення тексту, форматування тексту за допомогою стилів, створення списків, форматування розділів, додавання об'єктів

1. Створити на робочому столі папку «ІС».
2. У створеній папці створити документ Word з іменем Hardware.
3. Відкрити створений документ.
4. Встановити розмір сторінки А5.
5. Додати об'єкт WordArt з текстом «Апаратна складова».
6. Додати текст «Основні принципи функціонування».
7. Застосувати до цього тексту стиль **Заголовок 1**.
8. Перелічити основні принципи функціонування комп'ютера (оформити як нумерований список).
9. Вставити розрив сторінки.
10. На другій сторінці як багаторівневий список відобразити структуру пам'яті.
11. Закрити текстовий процесор, зберігши зміни у файлі.
12. Створити архівний файл, додавши у нього створену Вами папку і знищивши при цьому вихідні файли.
13. Одержаний архів перемістити в папку **Мои документи**.
14. Перевірити наявність архіву у папці і продемонструвати учителю.
15. Знищити архів.

Можливий вигляд документу

Апаратна складова

#### Основні принципи функціонування

1. Принцип двійкового кодування.
2. Принцип програмного управління.
3. Принцип адресності.
4. Магістрально-модульний принцип.

або явищем), яке зберігає істотні властивості і може бути вивчене підходящим інструментарієм тієї чи іншої науки» [4].

В усіх наведених означеннях підкреслюється, що модель є певним заміщенням об'єкту-оригіналу іншим об'єктом. Проте щодо врахування інших суттєвих ознак поняття модель означення суттєво відрізняються. Перше означення найбільш коротке і найлегше запам'ятовується, проте у ньому вказують форми подання моделей, але не підкреслюється зв'язок моделі й об'єкта-оригіналу. Зв'язок моделі із розв'язуванням деякої задачі найбільш явно відображено у другому означенні. Третє означення підкреслює той факт, що модель частково відображає властивості оригіналу (проте не вказується, яким чином слід добирати властивості для побудови моделі). У четвертому означенні вказується можливість моделювання не тільки об'єктів, але і явищ, процесів.

Зрозуміло, що детальне означення недоцільно одразу повідомляти школярам, адже воно містить багато складних для розуміння нюансів. Введенню поняття повинна передувати мотивація, під час якої слід продемонструвати важливість моделювання не тільки для науки, але й для повсякденного життя, присутність моделей у будь-якому процесі одержання інформації, звернути увагу на істотні ознаки поняття.

Серед усього розмаїття моделей в курсі інформатики розглядаються у першу чергу інформаційні. Утім, у авторів різних підручників немає однотайності ні у визначенні інформаційних моделей, ні у з'ясуванні місця інформаційних моделей серед інших видів моделей. У посібнику О.Ю. Гаєвського інформаційні моделі разом з математичними є видами знакових моделей. Н.В. Морзе вважає, що поняття інформаційної моделі є родовим щодо поняття математичної моделі, іншими видами інформаційної моделі є словесна, графічна, таблична, тобто однозначність трактування відображається і в означеннях інформаційної моделі.

Означення на зразок «Інформаційна модель — це сукупність інформації про об'єкт або процес» втрачає цілепокладання моделі, істотність інформації, на основі якої будується модель, крім того втрачається різниця між моделлю та інформаційною моделлю, адже будь-яка модель містить у собі деяку інформацію про об'єкт.

Н.В. Морзе пропонує таке означення: «Інформаційна модель — опис об'єкта чи процесу, у якому вказано деякі типові властивості і характеристики об'єкта, важливі для конкретної задачі».

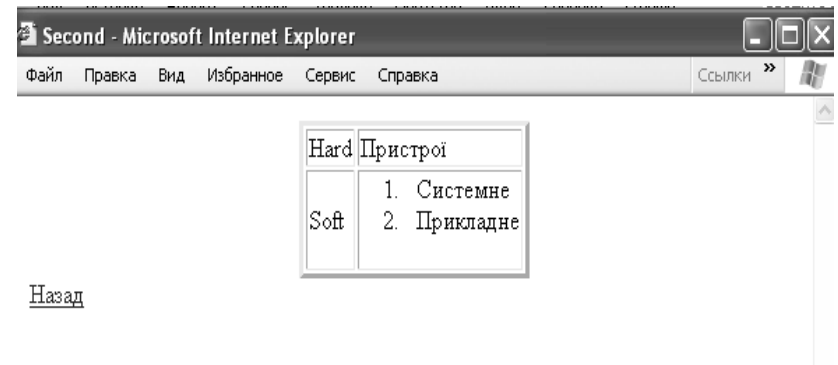
Наведемо ще одне означення інформаційної моделі (за посібником О.Г.Гейна): «Інформаційна модель — модель, яка представляє об'єкт, явище або процес набором параметрів і зв'язків між ними». У цьому означенні підкреслюється спосіб подання істотної інформації про об'єкт при побудові інформаційної моделі: значеннями тих чи інших параметрів. При цьому параметри можуть бути як числовими, так і інших типів, зв'язки між ними можуть задаватися формулами чи іншими способами, наприклад, у вигляді таблиць. З'ясування зв'язків між параметрами інформаційної моделі часто чи не найскладніша частина у побудові моделі, яка виникає після опису параметрів. У статті М.В. А.П. Єршова, в інформатиці розглядаються методологічні принципи побудови інформаційних моделей і маніпулювання з ними, а побудова й обґрунтування таких моделей є задачею окремих наук. Таким чином, інформаційне моделювання як прикладний розділ інформатики пов'язаний з різними предметними галузями: технікою, економікою, природничими, соціальними науками та ін.

#### Типи задач у галузі інформаційного моделювання (за зростанням складності)

1. Задано інформаційну модель, навчитися її аналізувати, робити висновки, використовувати для розв'язування задачі.
2. Систематизувати набір даних і таким чином створити інформаційну модель.
3. Розробити інформаційну модель реального об'єкту (процесу, системи).

#### Особливості формування поняття інформаційної системи

У процесі вивчення курсу інформатики учні повинні бути орієнтовані і як майбутні користувачі комп'ютера, і як люди, які в майбутньому можуть обрати професію, пов'язану зі створенням, виробництвом та обслуговуванням перетворювачів



Мал.7

#### Критерії оцінювання

	<i>Кількість балів</i>
Правильність збереження файлів	0,5
Заголовки сторінок	1
Задання фону сторінки	1
Вставка зображення	1
Додавання заголовку відповідного рівня	0,5
Вирівнювання заголовку	0,5
Зміна накреслення символів (курсив, напівжирний)	1
Зміна кольору тексту	1
Вирівнювання абзацу	0,5
Ненумерований список	1
Нумерований список	1
Вставка таблиці	1
Коректність гіперпосилань	1
Якість HTML-коду	1
<b>Разом</b>	<b>12</b>

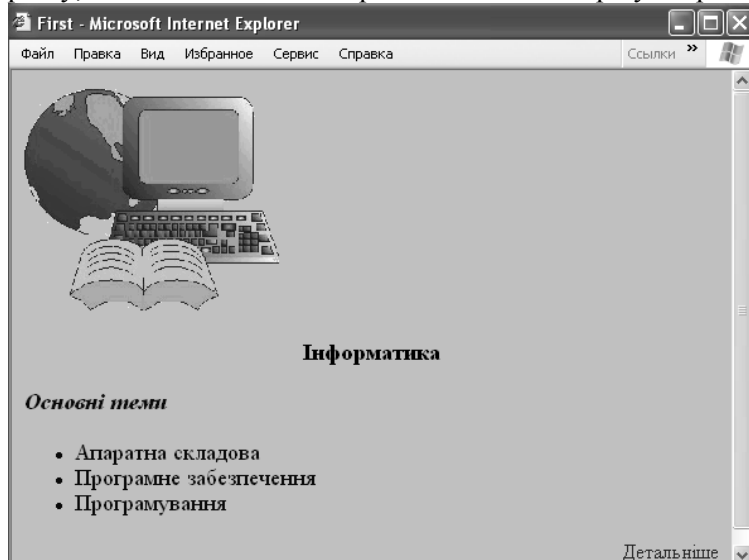
**Домашнє завдання.** Вивчити теоретичний матеріал конспекту. Записати в зошиті таблицю з формулами для розрахунку суми сплати за спожиту електроенергії протягом кварталу при наявності лічильника. Знайти середню суму за місяць.

Для того, щоб обчислити суму за місяць необхідно від поточних показників лічильника відняти попередні показники і одержану різницю помножити на тариф.

## Приклад завдань і критеріїв оцінювання практичної роботи з теми «Глобальна мережа Інтернет та її можливості»

**Тема:** Створення Web-документів

**Завдання.** Створити дві веб-сторінки з іменами z.html (заголовок сторінки *First*) і z1.html (заголовок сторінки *Second*), відповідно до зразку (мал. 6, 7). Колір фону першої сторінки *silver*, колір списку — *navy*. Текст «Інформатика» оформити як заголовок. Текст «Детальніше» є гіперпосиланням на другу сторінку, текст «Назад» — гіперпосиланням на першу сторінку.



Мал. 6

інформації. Їм потрібно знати, як користуватися комп'ютером, як його організовано і які принципи його конструювання.

### Основні цілі вивчення розділу

- сформувати поняття комп'ютер, інформаційна система, суттєві ознаки комп'ютера, програмна й апаратна складові інформаційної системи, архітектура комп'ютера, принципи функціонування, комп'ютерна мережа, канал зв'язку, сервер, робоча станція;
- познайомити з архітектурою конкретного комп'ютера й основними принципами роботи ЕОМ при опрацюванні повідомлень.

Опанування змістової лінії «Комп'ютер» відбувається за двома напрямками:

- теоретичне вивчення будови, принципів функціонування і зберігання даних (повідомлень) у комп'ютері;
- практичне опанування комп'ютера, одержання навичок його застосування для виконання різних видів роботи з інформацією.

Будова комп'ютера вивчається на рівні архітектури, тобто на рівні опису будови і принципів роботи комп'ютера без подробиць технічного характеру.

### Етапи формування поняття комп'ютер

1. Початкове ознайомлення з поняттям комп'ютера, систематизація та узагальнення знань учнів про комп'ютер.
2. Розкриття функціональної структури комп'ютера, призначення основних пристроїв, формування уявлень про принципи їх дії, принципи збереження, введення і виведення інформації.
3. Робота на персональному комп'ютері. Техніка безпеки при роботі з комп'ютером.
4. З'ясування основних принципів функціонування комп'ютера.
5. Вивчення основних галузей застосування комп'ютера.

При формуванні уявлень про комп'ютер слід виходити з того, що вони повинні бути пов'язаними з поняттями інформації й алгоритму. Як основний метод рекомендується бесіда на частково-пошуковій основі з опорою на асоціативне мислення (порівняння з пристроєм для опрацювання деякої сировини), що

дозволяє учням самостійно дати описове означення комп'ютера та визначити суттєві ознаки комп'ютера.

### Суттєві ознаки комп'ютера

- досить складний електричний пристрій, що складається з деякої сукупності взаємозв'язаних і взаємодіючих пристроїв, тобто система пристроїв;
- призначений для автоматичного швидкого опрацювання повідомлень за програмою (сукупність правил опрацювання);
- «може» опрацьовувати різні (але не будь-які) повідомлення, які надходять через відповідні пристрої введення, що перетворюють повідомлення на сукупність електричних сигналів, які фіксуються у пам'яті комп'ютера;
- «може» запам'ятовувати вхідні повідомлення та одержані в результаті опрацювання;
- «може» видавати опрацьовані повідомлення у різних формах через пристрої виведення;
- «може» працювати лише за правилами (програмами), заздалегідь введеними людиною (при цьому поняття програми означувати не рекомендується).

Учні повинні засвоїти, що:

- 1) за допомогою комп'ютера можна опрацьовувати такі види повідомлень: символні, числові, графічні, звукові;
- 2) будь-яка інформація в пам'яті комп'ютера подається у вигляді двійкового коду (правила опрацювання двійкових кодів вводити вважається недоцільним, оскільки цей матеріал не є практично значущим та важко розуміти).

Щоб свідомі розуміти, що комп'ютер використовується не тільки для виконання різного роду обчислень, слід крім поняття комп'ютер вводити поняття інформаційної системи. Внаслідок удосконалення технічних характеристик розширюються можливості використання комп'ютера при опрацюванні різних повідомлень. Тому такий пристрій разом з програмним забезпеченням і даними правильно називати не обчислювальною системою, а системою для опрацювання інформації, тобто **інформаційною системою**.

Слід відзначити, що в літературі зустрічається й інше тлумачення терміну «інформаційна система»: наприклад, у [33]

Після виконання практичної роботи таблиця повинна мати вигляд:

Назва річки	Загальна довжина	Довжина в Україні	Частка довжини в Україні
Дністер	1362	705	0,517621145
Прут	967	272	0,281282316
Тиса	977	272	0,278403275
Дніпро	2201	981	0,445706497
Сейм	748	250	0,334224599
Дунай	2850	164	0,05754386
	9105		0,319130282
	9105		0,319130282

У режимі відображення формул:

Назва річки	Загальна довжина	Довжина в Україні	Частка довжини в Україні
Дністер	1362	705	0,517621145
Прут	967	272	0,281282316
Тиса	977	272	0,278403275
Дніпро	2201	981	0,445706497
Сейм	748	250	0,334224599
Дунай	2850	164	0,05754386
	9105		0,319130282
	9105		0,319130282

### 6. Підведення підсумків. Повідомлення домашнього завдання

Сьогодні на уроці ми познайомилися, як використовувати формули для обчислень в середовищі табличного процесора, навчилися вводити формули в комірки.

Згадаємо, з якого символу повинна починатися формула?

*(Дорівнює)*

За допомогою якого інструменту можна поширити формулу на певний діапазон?

*(За допомогою автозаповнення)*

Як швидко обчислити суму значень комірок діапазону?

*(За допомогою функції СУММ)*

Як викликати функції, що найчастіше використовуються?

*(За допомогою кнопки Автосума, вибравши потрібну функцію натисненням на стрілочку поряд із кнопкою)*

Назва річки	Загальна довжина	Довжина в Україні	Частка довжини в Україні
Дністер	1362	705	0,517621145
Прут	967	272	0,281282316
Тиса	977	272	0,278403275
Дніпро	2201	981	0,445706497
Сейм	748	250	0,334224599
Дунай	2850	164	0,05754386
	9105		0,319130282

7. У комірці B9 обчислити сумарну довжину всіх перелічених річок за допомогою функції СУММ. Для цього:
- встановити курсор у комірку B9;
  - натиснути кнопку Автосумма  $\Sigma$  на панелі інструментів;
  - змінити діапазон сумування, виділений мерехтливою рамкою;
  - натиснути [Enter].
8. У комірці D9 обчислити середньої частки довжини річок в Україні за допомогою функції. Для виклику функції клацнути на стрілці біля кнопки  $\Sigma$  і вибрати необхідну.
9. У комірці C9 обчислити мінімальну довжину перелічених річок в Україні.

**Вихідний документ Rivers.xls** містить таблицю

Назва річки	Загальна довжина	Довжина в Україні	Частка довжини в Україні
Дністер	1362	705	
Прут	967	272	
Тиса	977	272	
Дніпро	2201	981	
Сейм	748	250	
Дунай	2850	164	

під інформаційною системою розуміються всеможливі засоби і методи організації і зберігання великих інформаційних масивів (сховищ інформації) і на їх основі надання інформаційних послуг населенню. У цьому розумінні інформаційна система близька до баз даних.

Важливим є розуміння учнями того, що комп'ютер є двоєдиною системою, яка складається з апаратної та програмної частини. Н.В. Морзе вважає більш доцільним виділяти не програмну, а інформаційну складову (до неї входять програми, з якими працює комп'ютер, та різні набори даних).

Після ознайомлення зі структурою інформаційної системи цілком логічно переходити до ознайомлення з апаратною складовою. Паралельно з вивченням основного матеріалу відбувається вивчення правил роботи на комп'ютері.

#### Етапи навчання роботи на комп'ютері

1. Правила техніки безпеки. На перших етапах рекомендується дозволити учням доступ лише до клавіатури та миші.
2. Клавіатурний тренажер для ознайомлення з призначенням клавіш.
3. Навчання розуміти повідомлення операційної системи, правильно на них реагувати, запускати програми, аналізувати результати.
4. Закріплення навичок роботи з комп'ютером, його периферійними пристроями та практичних та лабораторних роботах.

На заключному етапі вивчення теми слід значно розширити уявлення учнів про різні типи комп'ютерів, можливості їх використання, сфери застосування, сприяти продовженню процесу формування поняття комп'ютера. Урок з теми «Основні галузі застосування комп'ютерів» доцільно проводити у вигляді семінару, конкурсу на краще повідомлення тощо. Такий підхід дає можливість підвищити пізнавальну активність та інтерес учнів до інформатики, що, у свою чергу, впливає на ефективність засвоєння основних понять інформатики.

Вважається недоцільним вимагати від учнів глибоких знань відносно історії створення комп'ютера, дат і прізвищ, головним завданням є формування уявлень та знань учнів про те, що при

зміні апаратної складової основні принципи функціонування комп'ютера залишаються незмінними. Важливо також, щоб учні розуміли вплив науково-технічного прогресу на діяльність людини, роль учених України у розвитку обчислювальної техніки в світі.

## Самостійна робота студентів

### Матеріал для самостійного опрацювання

#### Вивчення основних пристроїв та принципів дії комп'ютера

Серед питань, пов'язаних із вивченням апаратури та принципів дії комп'ютера, можна виділити чотири основні групи:

- структура комп'ютера (на рівні структурної функціональної схеми), функції та призначення основних пристроїв взаємодії апаратної і програмної складових;
- фізичні основи функціонування основних елементів (передбачається, що ці питання повинні вивчатися на уроках фізики);
- арифметико-логічні основи будови і дії комп'ютера.

Вивчення основних понять інформаційної системи доцільно будувати циклічно, коли спочатку формуються лише найзагальніші уявлення про об'єкт вивчення, з часом дещо уточнюються його характеристики (чи ознаки поняття, якщо йдеться про його формування), далі (можливо в наступних класах чи при вивченні інших навчальних предметів) ці уточнення стають ще детальнішими і нарешті у кожного учня буде сформоване своє власне бачення об'єкта вивчення. При ознайомленні з основними пристроями комп'ютера доцільно разом з учнями побудувати відповідні таблиці для відображення їх основних характеристик, схематично відображати структуру, класифікацію сучасних комп'ютерів обмежених, пристрої комп'ютера рекомендується вивчати за допомогою навчальних фільмів, програм, комп'ютерних презентацій.

Тема насичена великою кількістю нових понять, які до того ж не використовуються у практичній діяльності. Для їх закріплення доцільно використовувати різноманітні дидактичні ігри (складання кросвордів, використання пантоміми, застосування ідей різних телевізійних ігор тощо). Можна також пропонувати учням нові терміни та їх описові тлумачення заносити до словника термінів. Така словникова робота має методичну

#### 4. Закріплення нового матеріалу

Записати формулу для знаходження:

- а) загальної кількості тканин (у комірку B5);
- б) середньої ціни (у комірку C5)

	A	B	C	D
1	Назва	Ціна, грн.	Кількість товару, м	
2	Ситець	9	50	=B2*C2
3	Вовна	70	30	=B3*C3
4	Шовк	50	10	=B4*C4
5		=B2+B3+B4	=(C2+C3+C4)/3	

#### 5. Практична робота

Тепер Ви зможете ознайомитися із введенням формул на практиці, а також навчитеся використовувати функції для більш швидкого знаходження суми і середнього значення. Під час практичної роботи не забувайте про правила техніки безпеки, дотримуйтесь інструкції до практичної роботи, слідкуйте за своєю поставою, акуратно здійснюйте введення інформації.

#### Методичні рекомендації до практичної роботи

1. Відкрити файл **Rivers.xls**, розташований у папці **Мои документы\Завдання**.
2. Записати в комірку D2 формулу для обчислення частки довжини Дністра в Україні. Перевірте правильність обчислення за зразком, наведеним нижче.
3. За допомогою автозаповнення поширити формулу для всіх річок.
4. У комірку B8 записати формулу для обчислення сумарної довжини всіх перелічених річок.
5. У комірку D8 записати формулу для обчислення середньої частки довжини річок в Україні.
6. Відформатувати таблицю згідно зі зразком:

- 1) натиснути знак «=»;
- 2) клацнути по комірці B2, при цьому комірку буде обведено мерехтливою рамкою, а адреса комірки з'явиться у формулі;
- 3) натиснути знак «\*»;
- 4) клацнути по комірці C2;
- 5) натиснути клавішу [Enter].

Після натиснення [Enter] введення формули припиняється, у комірці відобразатиметься результат виконання обчислень (450). Саму формулу можна побачити у рядку формул.

Запишемо:

Ввести формулу можна так:

- 1) натиснути «=»;
- 2) клацнути лівою кнопкою миші по комірці;
- 3) натиснути знак дії;
- 4) клацнути лівою кнопкою миші по комірці;
- 5) повторити пункти 3, 4 для всіх операцій;
- 6) натиснути [Enter].

Аналогічні формули вводяться і в комірки D3, D4:

	A	B	C	D
1	Назва	Ціна, грн.	Кількість товару, м	
2	Ситець	9	50	=B2*C2
3	Вовна	70	30	=B3*C3
4	Шовк	50	10	=B4*C4

Зверніть увагу, що формули в комірках D2, D3, D4 дуже схожі: вони вимагають перемножити значення двох сусідніх зліва комірок. У такому випадку для спрощення введення даних можна скористатися автозаповненням.

Запишемо.

Для автозаповнення слід:

- 1) клацнути лівою кнопкою миші по першій комірці з даними;
- 2) підвести вказівник миші до маркера автозаповнення, щоб вказівник набув вигляду тонкого чорного хрестика;
- 3) утримуючи ліву кнопку миші, протягти на необхідний діапазон.

цінність і сприяє систематизації та кращому запам'ятовуванню нових понять і термінів.

Ознайомлення учнів з поняттям процесора можна почати з його узагальненого опису. Головне — розуміння учнями призначення цього пристрою та його функцій.

Вивчення призначення запам'ятовуючих пристроїв доцільно починати з евристичної бесіди, у якій з'ясувати розуміння терміну «пам'ять». Далі доцільно пояснити, що запам'ятовуючі пристрої комп'ютера поділяють на внутрішні і зовнішні. При цьому можна провести аналогію із пам'яттю людини: внутрішня пам'ять — власна (біологічна) пам'ять людини, зовнішня — засоби для записування і зберігання повідомлень. Далі слід з'ясувати, з чого складається окремо внутрішня й окремо зовнішня пам'ять. Слід зазначити, що поділ на внутрішню і зовнішню пам'ять не відповідає розміщенню пристроїв всередині чи зовні системного блоку (жорсткий диск розміщується у сучасному комп'ютері всередині системного блоку, але відноситься до зовнішньої пам'яті). Після ознайомлення зі структурою пам'яті доцільно повідомити про одиниці вимірювання ємності запам'ятовуючих пристроїв. Важливим також є ознайомлення учнів із зовнішнім виглядом і функціональним призначенням різних пристроїв введення-виведення.

Особливо складними для учнів при вивченні апаратної складової інформаційної системи є питання, що стосуються функціональної схеми роботи комп'ютера, магістрально-модульного принципу та інших принципів функціонування пристроїв комп'ютера.

При вивченні принципів побудови учням достатньо знати, що до складу комп'ютера входить системний блок та під'єднані до нього пристрої введення-виведення; крім того, у спеціальні роз'єми (слоти) на материнській платі можуть встановлюватися контролери (спеціальні мікросхеми додаткових периферійних пристроїв, до яких під'єднуються зовнішні пристрої), різні пристрої пов'язані каналами передавання послідовностей сигналів (пропедевтика каналів зв'язку в мережах).

Важливо звернути увагу учнів на такі принципи функціонування комп'ютерів фон-нейманівської архітектури:

- *двійкового кодування* — повідомлення та програми подаються двійковими кодами;
- *програмного управління* — усі арифметико-логічні та управляючі операції здійснюються за програмами, що зберігаються в оперативній пам'яті;

- *адресності* — дані та програми знаходяться в окремих полях простору для зберігання кодів повідомлень в ОЗУ, кожен з яких має свою адресу — місце його знаходження в загальному просторі для зберігання кодів повідомлень;
- *магістрально-модульний* — до інформаційної магістралі (шини даних) можуть під'єднуватися додаткові периферійні пристрої.

Логічним завершенням теми «Інформаційна система» є ознайомлення з поняттям мережі, оскільки значна частина сучасних інформаційно-комунікаційних технологій ґрунтується на використанні комп'ютерних мереж. Учні повинні засвоїти, що першопричина створення мереж — бажання економити ресурси. Слід пояснити учням, що комп'ютери, які об'єднані в мережу, мають значно ширші можливості, ніж комп'ютери, які працюють окремо.

### **Завдання для самостійної роботи**

Для демонстрації важливості поняття моделі і моделювання як методу пізнання можна навести приклади з літератури. Наприклад, відомий літературний герой, детектив Шерлок Холмс стверджував, що «складність полягає в тому, щоб виділити з маси вигадок і домислів марних тлумачів і репортерів безперечних, непорушних фактів. Встановивши вихідні факти, ми починаємо будувати, ґрунтуючись на них, нашу теорію і спробуємо визначити, які моменти у даній справі можна вважати вузловими». Які суттєві ознаки моделі можна проілюструвати за допомогою цієї цитати? Наведіть власні приклади з літератури чи мистецтва для ілюстрації «життєвості» базових понять інформатики.

### **Запитання для самоконтролю**

1. Які особливості поняття інформація? Перелічіть етапи формування цього поняття.
2. Наведіть приклади, що демонструють взаємозв'язок понять інформація і повідомлення, інформація і шум.
3. Які особливості формування поняття «носії повідомлення», уявлень про способи подання і передавання повідомлень?
4. Які особливості висвітлення питань оцінювання й вимірювання інформації?
5. Перелічіть інформаційні процеси. Які особливості вивчення інформаційних процесів?
6. Висвітліть різні точки зору на цілі вивчення моделювання у загальноосвітній школі. Яку з них Ви вважаєте більш обґрунтованою?

➤ Як можна позбутися тексту у комірці D1?

*(Збільшити ширину стовпця C або встановити перенесення по словах у комірці C1)*

➤ При друці лінії сітки електронної таблиці за замовчуванням не відображаються. Як «розграфити» частину електронної таблиці, що містить дані?

*(Виділити діапазон A1:C4; клацнути по стрілці біля кнопки «Границы» панелі інструментів; клацнути по «Все границы»)*

### **3. Викладення нового матеріалу**

Нехай, наприклад, необхідно обчислити вартість кожного товару і занести дані у стовпець D. Які числа тут повинні стояти? *(Очікувана відповідь: 450; 2100; 500. Учитель вписує числа в комірки електронної таблиці).* Уявімо, що у комірці C3 знайдено помилку. Тут замість 30 повинно стояти 20. Нам необхідно змінити дані, але тоді повинен змінитися і результат. Тобто змінюючи комірку з даними нам необхідно перевірити і всі «залежні» комірки. А якщо таких комірок багато? Добре було б, якби табличний процесор сам здійснював такі перерахунки.

Така можливість є, якщо використовувати формули. Як ми вже знаємо, комірка може містити текст, числа або формули. Формула завжди починається із знака дорівнює і є виразом, що описує обчислення в комірках. Сьогодні на уроці ми з Вами навчимося використовувати формули для розв'язування задач у середовищі табличного процесора. Отже, запишемо тему уроку «Використання функцій та операцій».

Повернемося до розгляду задачі. Для того, щоб знайти вартість кожного товару, нам необхідно перемножити його ціну на кількість. Наприклад, для обчислення вартості ситця слід перемножити вміст комірок B2 і C2. Отже, формула матиме вигляд:  $= B2 * C2$ . Для того, щоб ввести формулу у комірку можна ввести такий вираз з клавіатури. Але при цьому легко припуститися помилки, наприклад, внести не ту адресу комірки. Тому вводити такі формули зручніше іншим способом:



Варт	##	Характеристики
------	----	----------------

### Фронтальне опитування

- З чого складається робочий аркуш?  
(З рядків і стовпців, на перетині яких утворюються комірки)
- Як позначаються рядки, стовпці, комірки електронної таблиці?  
(Рядки — натуральними числами,  
Стовпці — латинськими літерами  
A,...,Z,AA, ...IV  
Адреса комірки складається з позначення стовпця і номера рядка)
- Як ввести дані в комірку?  
(Зробити комірку поточною, ввести дані з клавіатури, натиснути [Enter])
- Чи завжди вміст комірки збігається з тим, що відображається у комірці?  
(Ні)
- Пояснити особливості відображення даних у домашньому завданні.  
(Якщо текст довший, ніж вміщується у комірку і наступна комірка зайнята, то відображається частина тексту; числа, запис яких не вміщується у комірку відображаються як ##; якщо комірка після комірки з текстом порожня, то текст відображається і в цій комірці)

➤ Дано таблицю:

	A	B	C	D
1	Назва	Ціна, грн.	Кількість товару, м	
2	Ситець	9	50	
3	Вовна	70	30	
4	Шовк	50	10	

- Які комірки у першому рядку містять дані?  
(A1, B1, C1)

7. Охарактеризуйте різні види моделей залежно від мети моделювання.
8. Проаналізуйте різні означення моделі, інформаційної моделі.
9. Перелічіть типи задач у галузі інформаційного моделювання.
10. Які цілі вивчення теми «Інформаційна система»? За якими напрямками здійснюється опанування змістової лінії «Комп'ютер»?
11. Перелічіть етапи формування поняття комп'ютер. З якими поняттями слід пов'язати поняття комп'ютер?
12. Які особливості поняття комп'ютер повинні засвоїти учні? Яка мета введення поняття інформаційної системи?
13. Перелічіть етапи навчання роботи на комп'ютері.
14. Коротко охарактеризуйте основні групи питань, пов'язаних із вивченням апаратури та принципів дії комп'ютера.
15. Які питання є найбільш складними для учнів при вивченні апаратної складової інформаційної системи? На що слід звернути увагу при висвітленні принципів дії комп'ютера?

## Лекція № 10

### МЕТОДИКА ОЗНАЙОМЛЕННЯ З ОКРЕМИМИ ВИДАМИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

#### Особливості вивчення операційної системи

##### Основні завдання теми

Використання комп'ютера людиною відбувається за схемою: задача (інформаційна потреба користувача) — добір та ініціалізація програми — опрацювання вхідних даних — аналіз результатів.

Важливо протягом всього курсу інформатики формувати розуміння того, за допомогою яких програмних засобів які інформаційні задачі можна розв'язувати і за допомогою яких засобів такі задачі можна розв'язувати ефективніше (швидше, якісніше, точніше тощо). При цьому учні повинні відрізнити задачі, які можна розв'язати за допомогою операційної системи, від задач, для розв'язування яких необхідно використати прикладне програмне забезпечення загального чи спеціального призначення.

Тема «Операційна система» важлива незалежно від того, які ставляться кінцеві цілі вивчення інформатики — користувачькі, програмістські чи спеціалізовані за відповідними профілями. При цьому важливо навчити учнів не конкретних команд окремо взятої операційної системи, а принципів роботи з операційною системою та її призначенням, оскільки завдання сучасного курсу інформатики полягає у формуванні вмінь орієнтуватися в можливостях використання різних операційних систем з подальшим самостійним вивченням особливостей будь-якої конкретної.

**Основна мета вивчення теми «Операційна система»:** сформулювати уявлення про операційну систему як набір програм, призначених для виконання певних конкретних функцій управління апаратними і програмними засобами інформаційної системи, і висновок про те, що працювати з сучасною інформаційною системою без операційної системи неможливо.

Як особливості вивчення теми можна відзначити необхідність звернути увагу на вивчення принципів роботи користувача з будь-якою операційною системою (незалежно від інтерфейсу), «перевіряючи» ці принципи демонстраційною роботою з конкретною операційною системою.

### Поняття файлу

Одним із основних понять теми є поняття *файлу*, введення якого передуює ознайомленню учнів з операційною системою. Суттєві ознаки цього поняття:

- 1) поіменована сукупність повідомлень, яка розглядається як єдине ціле (може бути навіть порожньою);
- 2) повідомлення можуть містити текстові, графічні, звукові дані;
- 3) повідомлення повинні бути закодовані, щоб їх можна було опрацювати за допомогою комп'ютера;
- 4) повідомлення можуть зберігатися у зовнішній пам'яті комп'ютера або опрацюватися в оперативній пам'яті;
- 5) така сукупність повідомлень є єдиним цілим;
- 6) один набір повідомлень не пов'язаний з іншим набором;
- 7) кожна сукупність має свою назву.

Таким чином, файл — це сукупність повідомлень, які можуть зберігатися та опрацюватися за допомогою комп'ютера як єдине ціле окремо від інших подібних сукупностей.

## Приклад план-конспекту уроку з теми «Табличний процесор»

**Тема уроку:** Використання функцій та операцій для опрацювання даних, поданих у таблиці

**Мета уроку:**

**дидактична:** показати необхідність використання формул при роботі з табличним процесором, ознайомити з правилами введення формул, формувати навички опрацювання табличних даних за допомогою функцій та операцій;

**розвивальна:** розвивати алгоритмічне мислення, навички аналізу та синтезу;

**виховна:** виховувати дисциплінованість, акуратність.

**Обладнання:** комп'ютери, приклади таблиць на дошці.

**Програмне забезпечення:** табличний процесор Microsoft Excel.

**Тип уроку:** комбінований.

### План уроку

1. Організаційний момент (2 хв)
2. Перевірка домашнього завдання. Актуалізація опорних знань (8 хв)
3. Викладення нового матеріалу (10 хв)
4. Закріплення нового матеріалу (5 хв)
5. Практична робота (15 хв)
6. Підведення підсумків, повідомлення домашнього завдання (5 хв)

### Хід уроку

#### 1. Організаційний момент

Привітання, повідомлення теми і завдань уроку.

#### 2. Перевірка домашнього завдання. Актуалізація опорних знань

Як домашнє завдання учні повинні були зобразити в зошиті, що відображатиметься в комірках таблиці

Вартість	755000	Характеристики	
----------	--------	----------------	--

після зменшення ширини стовпців удвічі.

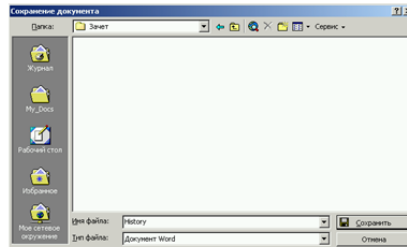
На дошці зображено правильне виконання домашнього завдання:

## Приклад

**Завдання:** зберегти поточний документ **History** у підкаталозі **Зачет** поточної папки

### Порядок дій

1. Обрати вказівку **Файл\Сохранить как...**
2. Двічі клацнути по значку папки **Зачет** для її відкриття



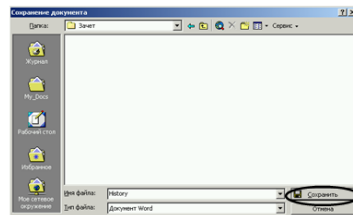
8

## Приклад

**Завдання:** зберегти поточний документ **History** у підкаталозі **Зачет** поточної папки

### Порядок дій

1. Обрати вказівку **Файл\Сохранить как...**
2. Двічі клацнути по значку папки **Зачет** для її відкриття
3. Натиснути **Сохранить** діалогового вікна



8

Основні характеристики файлу (ім'я, тип, розширення, місце зберігання, обсяг) доцільно ввести в ході евристичної бесіди. Розглядаючи типи файлів, варто зупинитися на типах подання повідомлень, що зберігаються у файлах.

Практика свідчить, що поняття шляху до файлу є досить складним, особливі утруднення викликає запис шляху до файлу починаючи не з кореневого, а з поточного (активного) каталогу. Для полегшення сприймання доцільно провести аналогію із правилами запису поштових адрес, звернувши увагу на призначення адрес, розділові знаки. Щоправда, після введення в Україні нової системи запису адрес, проведення такої аналогії ускладнюється протилежним порядком запису поштових адрес (від конкретного до загального) і адрес файлів (кожний наступний підкаталог міститься в попередньому).

### Поняття операційної системи

Поняття *операційної системи* складне й абстрактне, важко сприймається, тому на першому етапі доцільно ознайомити учнів з призначенням, а потім — більш детально вивчати характеристики та набір вказівок. При цьому варто спиратися на відомі приклади управління деякою системою, узагальнити суттєві ознаки людини, яка управляє роботою деякої системи, потім перейти до конструктивного означення операційної системи. Слід наголосити, що основне призначення будь-якої операційної системи — управління апаратними і програмними складовими (ресурсами) інформаційної системи. Приклад управління деякою системою може бути використаний і для формування в учнів уявлень (на інтуїтивному рівні) про основні складові операційної системи (файлова система, драйвера зовнішніх пристроїв, процесор вказівок).

Після з'ясування основних складових операційної системи можна пояснити ознаки, за якими класифікуються операційні системи:

- кількість користувачів, які одночасно працюються із системою;
- кількість задач, які можуть розв'язуватися одночасно;
- кількість розрядів адресної шини;
- базовий спосіб використання операційної системи;
- ресурси, які мінімально необхідні для роботи;

- інтерфейс користувача.

Важливим є ознайомлення з поняттям *інтерфейсу* користувача як сукупності засобів, за допомогою яких користувач може управляти роботою відповідного середовища, що має певні функції. Можна також ознайомити учнів із видами користувацького інтерфейсу: командний (найпростіший, забезпечує подання на екран системного запрошення для введення команд), WIMP (Windows Image Menu Pointer з використанням вікон, меню та вказівників), SILK (Speech Image Language Knowledge — за відповідною вказівкою голосом відбувається переміщення від одних пошукових образів до інших за змістовими семантичними зв'язками).

## Особливості ознайомлення з операційною системою Windows

В основі методики ознайомлення з Windows (Linux) лежить «об'єктний» підхід:

- операційна система працює з об'єктами, призначення яких — зберігання інформації на комп'ютері (файли, папки, диски);
- об'єкти подаються в операційній системі в різному вигляді (значки та вікна);
- з файлами і папками незалежно від їх подання можна виконувати певні операції;
- всі операції виконуються з об'єктами, які у відкритому стані являють собою вікна;
- правила роботи з різними вікнами одного типу єдині.

Завдання вчителя на першому етапі ознайомлення з операційною системою — навчити учнів знаходити об'єкти, виділяти їх, викликати контекстні меню, відрізнити виділені об'єкти від невиділених, доступні вказівки від недоступних, вибирати за допомогою лівої кнопки миші об'єкт, вказівку, елементи розгляді властивостей вікна слід пояснити основні елементи та правила роботи з різними типами вікон:

- програмними (мають систему вказівок та головне меню, рядок статусу та прикладі вікна);
- прикладними (існують тільки в середовищі деякої програми, у межах якої дозволяється їх переміщувати);

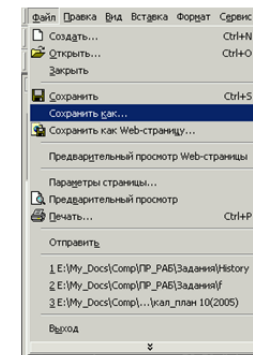
## Приклад

**Завдання:** зберегти поточний документ **History** у підкаталозі **Зачет** поточної папки

**Порядок дій**

1. Обрати вказівку

**Файл\Сохранить как...**



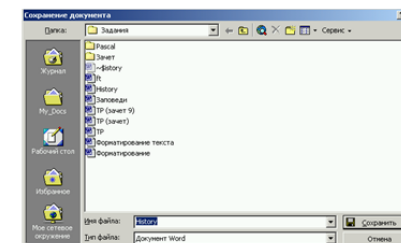
## Приклад

**Завдання:** зберегти поточний документ **History** у підкаталозі **Зачет** поточної папки


**Порядок дій**

1. Обрати вказівку

**Файл\Сохранить как...**



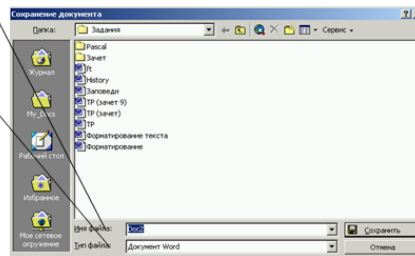
## Збереження документу

- 
- [Ctrl]+[S]
- Файл\Сохранить
- Файл\Сохранить как...

Якщо не перше збереження  
то вносяться зміни  
інакше ДИВ.

У рядку **Ім'я файла** вказується ім'я нового файлу чи змінюється існуюче

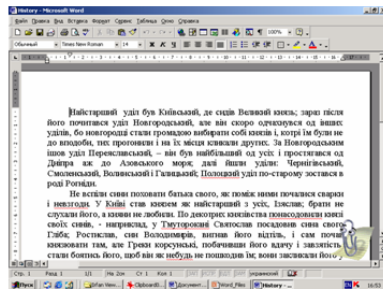
**Тип файло**  
дозволяє  
змінити формат  
файлу



## Приклад

**Завдання:** зберегти поточний документ **History** у підкаталозі **Зачет** поточної папки

**Порядок дій**



8

- діалоговими (містять спеціальні елементи, за допомогою яких користувач може змінити або уточнити параметри роботи в програмному середовищі);
- інформаційними (містять повідомлення про помилку, довідку, пораду чи попереджувальну інформацію; не дозволяється змінювати їх розміри, можуть призупинити роботу програми).

Спочатку доцільно навчити учнів виконувати основні дії з вікнами: відкривати, закривати, розкривати на повний екран, переміщувати, змінювати розміри, робити активним.

Принциповим і важливим є навчання правил роботи з кожним типом вікон. Найбільш складними для засвоєння є правила роботи з діалоговими вікнами, на яких доцільно окремо зупинитися. Розгляд особливостей цього типу вікон рекомендують проводити на прикладі вікна зміни властивостей робочого столу, оскільки при налаштуванні параметрів робочого столу учнів можна познайомити з усіма основними елементами діалогових вікон, крім того цей матеріал є цікавим для учнів, сприяє формуванню пізнавального інтересу. Щоправда, після вивчення цього матеріалу учні нерідко прагнуть щоразу налаштувати робочий стіл «під себе», що незручно для обслуговування комп'ютерів.

Дії з файлами та папками рекомендується вивчати не за допомогою менеджера файлів, а за допомогою контекстного меню на робочому столі. Це дає можливість привчити використовувати контекстне меню, а також продемонструвати різні способи виконання дій. Далі доцільно пояснити призначення кнопки *Пуск* та розглянути правила виконання програм і документів, завершення роботи з операційною системою, навчити створювати найпростіші документи за допомогою менеджера файлів, а також продемонструвати різні способи виконання дій з об'єктами в середовищі *Провідника*. Розпочинати вивчення доцільно із ознайомлення з правилами виконання операцій за допомогою протягування об'єктів, і лише пізніше ознайомити із призначенням кнопок панелі інструментів, а далі і призначенням основних вказівок головного меню.

При організації практичних робіт на відпрацювання навичок виконання операцій з файлами та папками слід давати чіткі конкретні вказівки щодо присвоєння імен файлам і папкам, їх

розташування. Така організація полегшує керування роботою учнів.

Важливим є питання навчання учнів пошуку інформації, оскільки сьогодні вміння людини успішно виконувати якісний інформаційний пошук набуває все більшого значення. Саме тому одним із основних завдань шкільного курсу інформатики є формування вмінь та навичок пошуку потрібної інформації.

Для здійснення пошуку в операційній системі Windows вбудовано спеціальний розділ до меню Пуск. Крім того, із задачами пошуку інформації учні стикаються і при роботі з довідковою системою (ознайомлення з правилами роботи з довідкою та навчання вчасно звертатися до її послуг доцільно здійснювати на перших уроках теми).

Для закріплення вмінь пошуку слід добирати вправи двох типів:

1. Знайти за допомогою довідкової системи інформацію про конкретне поняття.

2. Знайти місцезнаходження файлу за ім'ям, розміром, вмістом тексту, датою та часом останньої зміни, типом, обсягом.

Спочатку слід пропонувати шукати за однією ознакою, потім розглядати сукупність параметрів (при цьому звертати увагу та використання логічних операцій *і* та *або*).

Наприкінці вивчення теми можна ознайомити учнів із налагодженням роботи операційної системи, надавши можливість змінити параметри роботи миші, клавіатури, екрана тощо.

## Ознайомлення з можливостями глобальної мережі Інтернет

Відомості про принципи будови глобальної мережі Інтернет є світоглядними, тому слід їх ретельно пояснити учням.

*Основні навчальні завдання* полягають у формуванні уявлень та знань про:

- закономірності та об'єктивну необхідність виникнення глобальної мережі;
- призначення Інтернету як всесвітнього комунікаційного середовища;
- основні ресурси Інтернету;

## Приклад

**Завдання:** відкрити текстовий документ з іменем **F**, розташований у надкаталозі поточного каталогу

**Порядок дій**

1. Вікнікати вікно відкриття файлу (див. слайд 4 )

2. Натиснути кнопку 

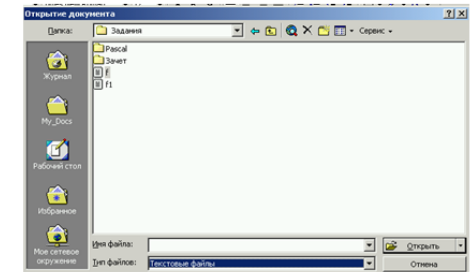
для переходу в надкаталог

3. Відкрити список **Тип**

**файлов**

4. Використовуючи смугу прокрутки, знайти **Текстовые файлы**

5. Клацнути лівою кнопкою миші



## Приклад

**Завдання:** відкрити текстовий документ з іменем **F**, розташований у надкаталозі поточного каталогу

**Порядок дій**

1. Вікнікати вікно відкриття файлу (див. слайд 4 )

2. Натиснути кнопку 

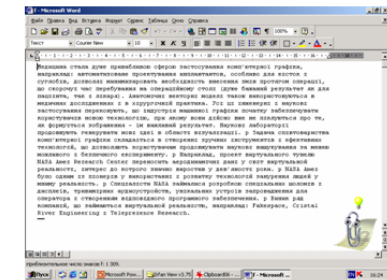
для переходу в надкаталог

3. Відкрити список **Тип**

**файлов**

4. Використовуючи смугу прокрутки, знайти **Текстовые файлы**

5. Клацнути лівою кнопкою миші




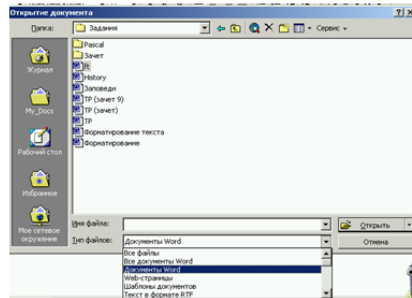
6. Двічі клацнути лівою кнопкою миші по значку файлу **F** для його відкриття

## Приклад

**Завдання:** відкрити текстовий документ з іменем **F**, розташований у надкаталозі поточного каталогу

### Порядок дій

1. Вікні відкриття файлу (див. слайд 4)
2. Натиснути кнопку  для переходу в надкаталог
3. Відкрити список **Тип файлів**




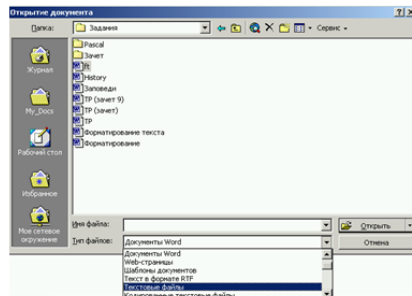
6

## Приклад

**Завдання:** відкрити текстовий документ з іменем **F**, розташований у надкаталозі поточного каталогу

### Порядок дій

1. Вікні відкриття файлу (див. слайд 4)
2. Натиснути кнопку  для переходу в надкаталог
3. Відкрити список **Тип файлів**
4. Використовуючи смугу прокрутки, знайти **Текстовые файлы**



6

- необхідність використання спеціального програмного забезпечення для кожного ресурсу мережі;
- принципи об'єднання комп'ютерів у мережі;
- необхідність певного спеціального апаратного та програмного забезпечення для під'єднання до мережі Інтернет;
- необхідність певних стандартів: адресація, протоколи;
- наявність каналів зв'язку;
- існування двох типів комп'ютерів — серверів і робочих станцій;
- використання сервісних програм двох типів — програм-серверів і програм-клієнтів.

Ознайомлення з можливостями Інтернету доцільно проводити у два етапи:

- розгляд світоглядних питань організації та розвитку Інтернету: історія виникнення, функції спеціальних пристроїв для об'єднання комп'ютерів, створення єдиної системи адрес, система правил передавання інформаційних та програмних ресурсів, а також апаратні, програмні та інформаційні ресурси мережі;
- формування знань, умінь і навичок щодо користування основними послугами Інтернету та правилами роботи зі спеціальним програмним забезпеченням.

*Внутрішньпредметні зв'язки:* з'ясування джерел одержання людиною інформації (будь-які прояви оточуючого світу, спілкування між людьми); засоби зберігання, подання, передавання повідомлень та показ їх використання для надання послуг Інтернет. Важливо, щоб учні розуміли, що в глобальній мережі Інтернет немає ні президента, ні директора, вона функціонує завдяки наявності певних стандартів взаємодії комп'ютерів та встановлених на них програм. Світоглядне значення має розуміння необхідності розробки спеціальних пристроїв (мостів, шлюзів, маршрутизаторів) та створення єдиної системи адрес для об'єднання комп'ютерів так, щоб різна будова локальних мереж та їх віддаленість не мали значення.

При поясненні нових термінів, назв спеціальних пристроїв і програм не слід намагатися давати їх формальні означення та пояснення. Вводити такі поняття доцільно, спираючись на асоціативне мислення учнів. Наприклад, мости та шлюзи можна

порівняти із відповідними інженерними спорудами, протокол із певними правилами спілкування (при зустрічі посадових осіб чи при відвідуванні лікаря), канали зв'язку — з транспортними системами вантажних та пасажирських перевезень.

При вивченні правил використання глобальної мережі Інтернет учні зустрічаються з досить великою кількістю англomовних термінів, тому рекомендується пропонувати їх переклад. По-перше, це дозволяє зрозуміти зміст поняття, по-друге, допомагає учням поповнити англomовний словниковий запас. Ознайомлення із принципами функціонування Інтернету рекомендується в першому наближенні, без зайвих деталей ознайомити учнів із основними послугами, які надаються користувачам глобальної мережі Інтернет: послуги доступу до файлів (файлові архіви), інформаційні послуги (веб-сервіс), послуги повідомлень (електронна пошта, телеконференції, списки розсилання, примусове розсилання), інтерактивні послуги (міжнародна лінія комунікацій у реальному часі IRC, комунікаційні послуги реального часу ICQ).

Учні повинні усвідомити, що Інтернет для користувача є джерелом одержання послуг. Коли говорять про вміння використовувати Інтернет у своїй професійній діяльності, мають на увазі не «вміння взагалі», а знання правил використання деякої конкретної служби або набору найчастіше використовуваних служб. В інформаційних технологіях поняття «служба» розуміють як дві програми, що взаємодіють між собою певним чином, одна з цих програм називається сервером, інша — клієнтом (при цьому слід звернути увагу на відмінності відповідно з комп'ютером-сервером та людиною-клієнтом).

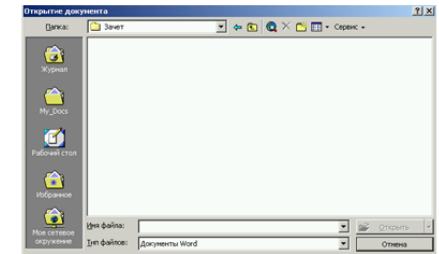
Ознайомлювати учнів з електронною поштою методично обґрунтовано за такою схемою: поняття електронної адреси, основні складові електронного листа, поняття поштової програми, яка забезпечує роботу користувача з поштою; основні послуги поштової програми, додаткові послуги поштових програм (адресна книга, правила упорядкування листів, фільтрування тощо), налагодження облікового запису та робота з поштовими серверами. Особливу увагу слід звернути на постановку проблемних запитань, які базуються на аналогіях з життєвими добре знайомими учням завданнями. Для ефективного вивчення теми потрібно, щоб у класі була

## Приклад

**Завдання:** відкрити текстовий документ з іменем **F**, розташований у надкаталозі поточного каталогу

### Порядок дій

1. Викликати вікно відкриття файлу (див. [слайд 4](#))




## Приклад

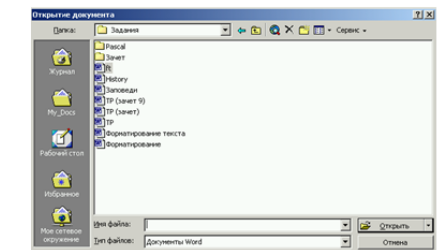
**Завдання:** відкрити текстовий документ з іменем **F**, розташований у надкаталозі поточного каталогу

### Порядок дій

1. Викликати вікно відкриття файлу (див. [слайд 4](#))

2. Натиснути кнопку 

для переходу в надкаталог

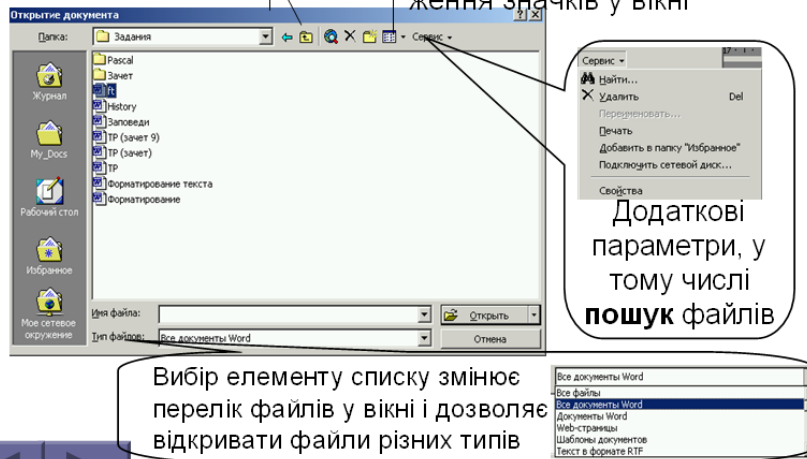




## Вікно відкриття документу

На рівень угору

Зміна способів відображення значків у вікні



Вибір елемента списка змінює перелік файлів у вікні і дозволяє відкривати файли різних типів

Додаткові параметри, у тому числі пошук файлів

## Приклад

**Завдання:** відкрити текстовий документ з іменем **F**, розташований у надкаталозі поточного каталогу

**Порядок дій**



6

налагоджена та організована робота моделі електронної пошти, щоб учні могли не тільки відправляти, але й приймати повідомлення, ознайомлюватися з ними, давати відповідь. Для сервера локальної мережі можна застосовувати операційну систему Windows NT, яка містить програму поштового сервера.

Після ознайомлення учнів з електронною поштою та поштовою програмою доцільно розглянути призначення телеконференцій та роботу з ними. При цьому слід зазначити, що крім терміна телеконференції як синоніми вживаються терміни групи новин, новини Usenet, news. Враховуючи подібність телеконференцій та електронної пошти, доцільно запропонувати учням завдання на порівняння цих послуг. Учні також повинні зрозуміти наявність груп у телеконференціях, імена яких мають ієрархічну структуру.

Під час вивчення послуги World Wide Web слід підкреслити, що принцип її дії базується на трьох основних компонентах: формат подання HTML-документів, HTTP-протокол передавання тексту та поняття гіпертекстового посилання. Важливим є також те, що для створення і перегляду веб-документів використовуються різні програми.

Як уже зазначалося, одним із завдань шкільного курсу інформатики є навчання здійснювати результативний пошук інформації, що потребує від них знань про способи її подання в електронному вигляді, технології зберігання та передавання за допомогою комунікаційної мережі Інтернет. Учні повинні навчитися орієнтуватися в інформаційному просторі мережі Інтернет, ознайомитися з механізмами пошуку інформації в тематичних каталогах та індексних базах даних, навчитися будувати запити і правильно оцінювати знайдену інформацію, уточнювати пошуковий запит. При навчанні методів пошуку інформації в Інтернеті спочатку необхідно пригадати основні інформаційні та комунікаційні послуги мережі Інтернет, які є джерелами для одержання інформації, і підкреслити, що кожен із них може стати середовищем пошуку відповідних даних, у кожному з яких існують спеціальні правила організації такого процесу. Учні повинні одержати спочатку загальне уявлення про пошукові служби, з'ясувати їх призначення та структуру. За принципом дії розрізняють кілька видів пошукових систем, найбільш поширеними є пошукові каталоги, рейтингові системи

й індексні бази даних. Вивчаючи мову запитів певної пошукової системи доцільно спочатку визначати результати пошуку за запитом теоретично, а потім підтверджувати висунуту версію практично.

## Самостійна робота студентів

### Матеріал для самостійного опрацювання


#### Особливості вивчення основ роботи з дисками

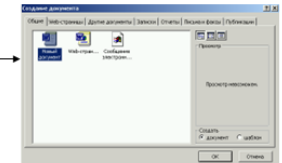
Тема «Основи роботи з дисками» передбачає ознайомлення учнів із правилами розміщення інформації на магнітних дисках, вивчення правил роботи з окремими службовими програмами, антивірусними програмами та програмами-архіваторами. Основні принципи розміщення інформації на магнітних дисках є досить складними для розуміння учнями, тому їх детальне вивчення не передбачено в класах універсального, філологічного, суспільно-гуманітарного, художньо-естетичного, спортивного профілів. Зазначені вище питання розглядаються у рамках теми «Інформаційна система».

Призначення дискових накопичувачів, їх основні характеристики та принципи роботи вивчаються під час ознайомлення учнів з апаратним забезпеченням комп'ютера. Якщо тема «Основи роботи з дисками» розглядається окремо, то необхідно повторити цей матеріал. Далі доцільно пояснити учням принципи поділу дисків на доріжки та сектори, поняття кластера. Для зрозуміння принципів розміщення інформації на диску можна навести аналогію з упорядкуванням місць на стадіоні: уся сукупність місць розбивається на сектори, кожен з яких має свій номер, у кожному секторі нумеруються ряди, а в ряду — місця. Таке порівняння дозволяє учням дійти висновку, що кожне місце на диску, до якого записується інформація, має свою адресу (щоправда, спосіб адресації на диску дещо інший: диск ділиться на доріжки, а доріжки, у свою чергу, — на сектори).

Поняття кластера, FAT-таблиці вважаються необов'язковими для запам'ятання, їх можна не виносити на тематичну атестацію. У той же час ці знання є світоглядними, базовими, які допомагатимуть швидше сприймати багато іншої інформації. Для полегшення їх сприйняття доцільно використовувати наочність, аналогії, порівняння з подібною організацією даних у життєвих ситуаціях. При введенні поняття кластера слід пояснити учням, що розмір сектора є стандартним і для збільшення обсягу

## Створення файлу

1. У вікні папки в контекстному меню обрати *Документ Microsoft Word*
2. У вікні Word:
  1. Кнопка  на панелі інструментів **Стандартная**
  2. Комбінація клавіш [Ctrl] +[N]
  3. Команда меню **Файл\Создать...**  
(можна обрати шаблон, майстер)




3

## Відкриття файлу

1. У вікні папки двічі клацнути по значку файлу (запускається Word і відкривається файл)
2. У вікні Word:

Якщо файл нещодавно був відкритий, то обрати його ім'я в меню **Файл**

інакше

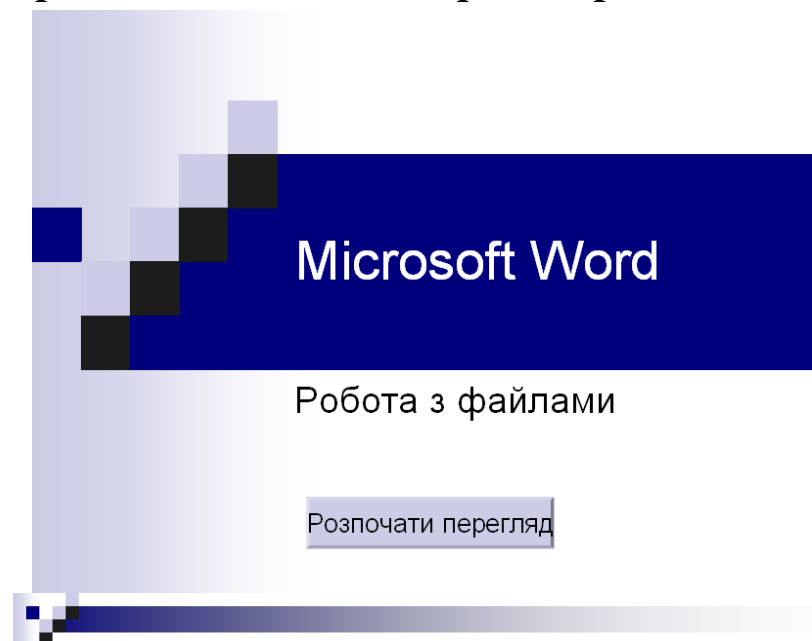
  - 
  - [Ctrl] +[O]
  - **Файл\Открыть...**

Викликається вікно відкриття документу (див. наступний слайд)



4

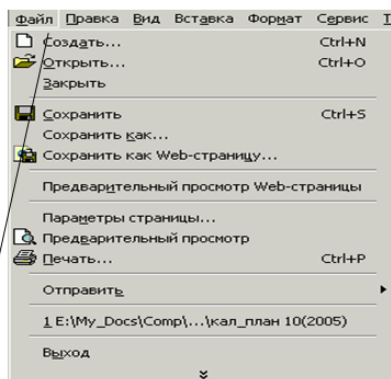
## Приклад змісту навчальної комп'ютерної презентації з «Текстовий редактор»



### Основні дії з файлами

- Створення
- Відкриття
- Збереження

Для виконання дій з файлами призначено пункт меню **Файл**



2

пам'яті, який може бути адресований, необхідно, щоб мінімальний обсяг пам'яті, який має свою адресу (кластер), складався більше, ніж з одного сектора. У сильних класах можна провести відповідні розрахунки кількості можливих адрес, обсягу пам'яті, що може бути адресований при наданні адреси кожному сектору. Доцільно також ознайомити учнів із поняттям FAT-таблиці, яка містить адресу кожного кластера та інформацію про те, чи кластер зіпсований, вільний, заповнений, або адресу кластера, у якому знаходиться наступна частина інформації даного файла.

У рамках вивчення теми «Основи роботи з дисками» слід також ознайомити учнів з основними вказівками операційної системи для роботи з дисками. Ознайомлення учнів доцільно будувати дедуктивно: спочатку роз'яснити призначення вказівки, а потім запропонувати виконати конкретні вправи. Вважається доцільним розглядати такі вказівки як форматування, сканування, дефрагментація диску, стискання диску і даних на диску. Останнє поняття тісно пов'язане з поняттям архівування, проте в літературі зустрічається різне співвідношення обсягів цих понять. У деяких джерелах (наприклад, у підручнику «Методика навчання інформатики» Н.В. Морзе) програми стискання інформації поділяють на дві групи — програми резервного копіювання (створення копій вихідної інформації на резервних носіях) і програми-архіватори (стискання одного або кількох файлів даних з метою економії місця на диску), в інших (посібник «Інформатика» для учнів 7–11 класів О.Ю. Гаєвського) стверджується, що за допомогою архівації можна вирішувати як завдання зменшення обсягу файлів, так і завдання резервного копіювання. Для ознайомлення учнів з поняттями стискання й архівування можна спочатку ознайомити із відповідними вказівками операційної системи, а пізніше розвинути ці знання при вивченні програм-архіваторів.

Перш ніж почати ознайомлення учнів з характеристиками конкретної програми-архіватора, слід з'ясувати їх основне призначення, процедури, що лежать в основі їх роботи. Також доцільно розглянути типові функції архіваторів: створення нового архіву, відкриття раніше створеного архіву, занесення файлів до архіву, переписування файлів з архіву, видалення файлів з архіву, перегляд вмісту архіву, перевірка виконання процесу архівування, поновлення вмісту архіву, перегляд вмісту окремих файлів архіву, розгортання архіву.

Практика свідчить, що після ознайомлення учнів з узагальненими правилами роботи з програмами-архіваторами

вивчення конкретної програми може відбуватися індуктивно за допомогою конкретних завдань. При формуванні навичок створення архіву за допомогою вказівок програми-архіватора (а не контекстного меню) необхідно звернути увагу на відмінності розташування архівного файлу і файлів, які вміщуються в архів.

Вивчення комп'ютерних вірусів викликає в учнів особливу зацікавленість. Враховуючи це, доцільно вивчення даної теми проводити у формі уроку-семінару, на якому учні самостійно під керівництвом учителя пояснюватимуть основні поняття теми. У бесіді доцільно порівняти біологічні й комп'ютерні віруси, знайти спільне і відмінне. Вивчення матеріалу про комп'ютерні віруси передбачає не тільки ознайомлення учнів з цим поняттям, але й формування уявлень про профілактику заражень комп'ютера вірусами, навичок використання антивірусних програм. При цьому спочатку необов'язково демонструвати роботу конкретної антивірусної програми, основне — розуміння того, що можна зробити з її допомогою (особливо враховуючи простоту використання сучасних антивірусів, відсутність необхідності їх налаштування для пересічного користувача).

Під час організації практичних робіт теми доцільно добирати завдання таким чином, щоб вони дозволяли повторювати попередній матеріал, у першу чергу роботу з файлами і каталогами (створення папок, виділення групи файлів, копіювання файлів, пошук файлів за різними параметрами тощо).

### Вивчення мови розмітки гіпертексту

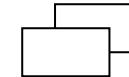
Перш ніж створювати HTML-документ або знайомити учнів з можливостями HTML-редактора, необхідно повторити поняття гіпертексту, його основні ознаки, чим відрізняється звичайний текст від гіпертексту. Слід пригадати, які програми призначені для перегляду гіпертекстових документів, звернути увагу учнів на те, що такі програми не дозволяють створювати веб-сторінки, розробка яких може здійснюватися за допомогою спеціальних редакторів.

Потрібно відзначити, що використання професійних редакторів веб-сторінок у шкільному курсі інформатики вважається недоцільним, оскільки більшість редакторів коду мають досить громіздкий інтерфейс, а візуальні редактори генерують HTML-код, часто незрозумілий початківцям. Тому рекомендується здійснювати редагування HTML-коду за допомогою найпростішого текстового редактора (наприклад, *Блокноту*) і переглядати створені таким чином сторінки за

11. Пензлем квадратної форми намалювати на ялинці лампочки гірлянди.

12. Намалювати під ялинкою подарунки:

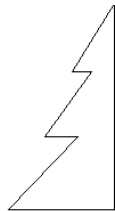
- 1) Встановити для інструменту **Линия** середню товщину.
- 2) Вибрати інструмент **Прямоугольник** із заповненням та межею.
- 3) Вибрати колір межі прямокутника (основний колір) жовтий і колір заповнення прямокутника (колір фону) червоний.
- 4) Намалювати прямокутник.
- 5) Встановити білий колір фону.
- 6) Скопіювати прямокутник для одержання такого розташування:



- 7) Застосувати до зображення прямокутників інструмент **Масштаб**.
  - 8) Вибрати інструмент **Линия**.
  - 9) З'єднати вершини прямокутників, щоб одержати зображення паралелепіпеда.
  - 10) Вибрати інструмент **Заливка** і темно-червоний колір.
  - 11) Виконати заливку бічної та верхньої грані подарунка.
  - 12) За допомогою **Ластика/Цветного ластика** знищити невидимі лінії паралелепіпеда (перевірте колір фону!).
  - 13) Повернутися до попереднього масштабу.
  - 14) Обрати **Линию** найбільшої товщини та намалювати стрічки на подарунку.
  - 15) Аналогічно намалювати ще один подарунок, використовуючи **Скругленный прямоугольник**.
13. Оформити подарунки за допомогою інструменту **Распылитель**.
14. Біля ялинки додати текст «З Новим роком!», змінивши розмір шрифту та гарнітуру (зверніть увагу на вибір кирилиці).

## Рекомендації до виконання завдання

1. Встановити розмір *полотна* 15×10 см, використовуючи *атрибути малюнка*.
2. Вибрати в палітрі основний колір зелений.
3. Вибрати інструмент **Многоугольник** із заповненням без контуру.
4. Намалювати половину ялинки такого вигляду (для одержання вертикальних та горизонтальних ліній малювати, утримуючи клавішу [Shift]):



5. Зняти встановлення непрозорого фону (пункт меню **Рисунок**).
6. Виділити намальовану ялинку за допомогою інструменту **Выделение**.
7. Скопіювати виділений фрагмент на вільне місце. Для цього перетягти його, утримуючи клавішу [Ctrl].
8. Відобразити скопійований фрагмент *зліва направо* і перетягти так, щоб він сумістився з першою половиною ялинки.
9. Намалювати на ялинці різнокольорові кульки. Для цього:
  - 1) Вибрати інструмент **Эллипс** із заповненням без межі.
  - 2) Вибрати колір в палітрі.
  - 3) Намалювати на ялинці круг, утримуючи клавішу [Shift].
  - 4) Повторити пункти 2), 3) для кількох кульок.
  - 5) Вибрати інструмент **Кисть**, форма пензля кругла (середній розмір).
  - 6) На кожній кульці поставити пензлем білого кольору кілька точок, як відблиск.
10. Використовуючи пензель форми \ або /, намалювати на ялинці дощик та серпантин.

допомогою браузера. Такий підхід сприяє розумінню учнями різниці між програмами для створення веб-сторінок і програмами для їх перегляду, дозволяє краще зрозуміти принцип розмітки тексту за допомогою елементів HTML. Проте слід зауважити, що в такому випадку учням необхідно запам'ятовувати значну кількість тегів та послідовності операцій для створення того чи іншого елемента веб-сторінки. Крім того, складність розробки HTML-документів за допомогою найпростішого текстового редактора полягає також у відсутності реалізації принципу WYSIWYG («що бачиш, те й отримуєш»), до якого учні звикли при вивченні інших видів прикладного програмного забезпечення. У той же час створення HTML-документів без використання спеціальних (особливо візуальних) редакторів сприяє розумінню принципу складання найпростіших програм.

Після актуалізації знань учнів про гіпертекст доцільно пригадати, що кожна HTML-сторінка має свою унікальну URL-адресу в Інтернеті.

При введенні поняття мови розмітки гіпертексту доцільно навести ознаки, які забезпечують її популярність: відносна простота мови, незалежність HTML-документу від комп'ютера й операційної системи, підтримка гіпертексту. Слід наголосити, що основна особливість створення веб-документу полягає в тому, що автору документу невідомо, на якій моделі комп'ютера, у якій операційній системі і якими програмами цей документ відобразатиметься на екрані користувача. Цей «рівень незнання» не дозволяє жорстко задавати параметри форматування HTML-документу. Значна частина тегів призначена не для точного задання параметрів тексту, а для встановлення призначення окремих його елементів, а браузер користувача сам «приймає рішення» про їх відображення.

При ознайомленні учнів з основними принципами мови розмітки гіпертексту доцільно продемонструвати за допомогою браузера заздалегідь створену просту гіпертекстову сторінку, яка містить заголовок, відформатований текст, таблицю, списки, малюнок, посилання на іншу сторінку. Разом з учнями доцільно на сторінці знайти основні структурні елементи. Далі, не закриваючи вікно браузера, слід відкрити цю сторінку за допомогою Блокнота, проаналізувати структуру документу, специфіку опису окремих елементів.

Необхідно звернути увагу учнів на те, що HTML є мовою розмітки, спеціальні вказівки впливають на спосіб відображення частин документа. «Контейнер», що містить дані, називають елементом, початковий і кінцевий маркер елемента називають

тегами. Важливо, щоб учні засвоїли правила оголошення елемента, навчилися розрізняти початковий і кінцевий тег, відмінності імені елемента, атрибуту, значення атрибуту. Слід також звернути увагу учнів, що одна з головних ідей мови HTML — можливість вкладання елементів.

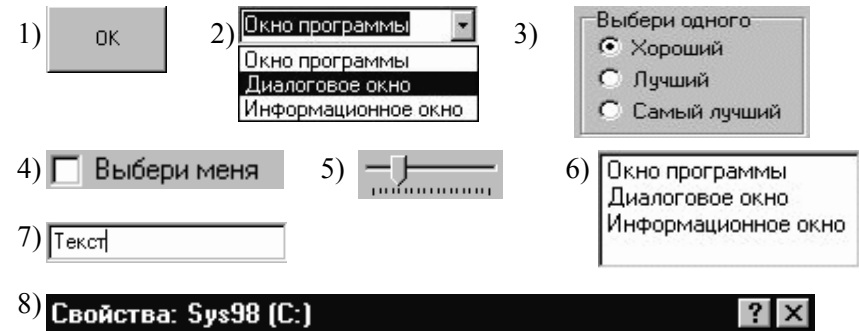
На початковому етапі вивчення HTML учні повинні засвоїти поняття структури гіпертекстового документу, навчитися читати готові HTML-сторінки. Для формування уявлень про структуру документу для аналізу учням слід пропонувати не складні сторінки, а невеличкі, що містять небагато різних елементів та основного тексту. Разом з учнями доцільно аналізувати структуру документу, визначати подвійні та одинарні теги, пробувати визначати призначення окремих елементів тощо.

Важливо також, щоб учні навчилися створювати власні HTML-документи, особливу увагу слід звернути на задання розширення файлу. Застосування прийому експериментальної перевірки результатів дії нових елементів дає можливість сформувати в учнів навички працювати одночасно з двома вікнами — вікном браузера та вікном редактора гіпертексту, постійно зберігати вміст HTML-файлу та поновлювати його зовнішній вигляд у середовищі браузера.

При складанні учнями власних HTML-документів слід звернути їх увагу на доцільність розташування тегів у документі «сходінками». При введенні нових елементів не потрібно намагатися відразу подавати їх повний формат, із атрибутами елементів доцільно знайомити учнів поступово. Для кожного окремого елемента слід пропонувати учням для розгляду спочатку конкретні приклади сторінок, які містять лише ті елементи, які вже вивчені раніше чи зараз вивчаються, а потім для закріплення цього елемента пропонувати конкретні завдання для самостійного виконання. Практика свідчить, що при невеликій кількості вивчених елементів виконання вправ не викликає зацікавленості в учнів, тому не варто пропонувати учням виконувати багато однотипних завдань.

### **Завдання для самостійної роботи**

1. Вище зазначалося про доцільність використання порівняння операційної системи та людини, що керує роботою деякої системи. Розглянемо роботу готелю, управління яким здійснює адміністратор (аналог операційної системи). Які функції повинен виконувати адміністратор? Як це співвіднести із функціями операційної системи? Розгляньте можливість продовження аналогії: файли — ..., файлова



Мал. 4

**Зауваження.** Враховуючи велику кількість зображень у тестовому завданні, можна використати електронну форму подання завдань.

### **Приклад завдань практичної роботи з теми «Графічний редактор»**

**Тема:** Використання основних інструментів у середовищі графічного редактора

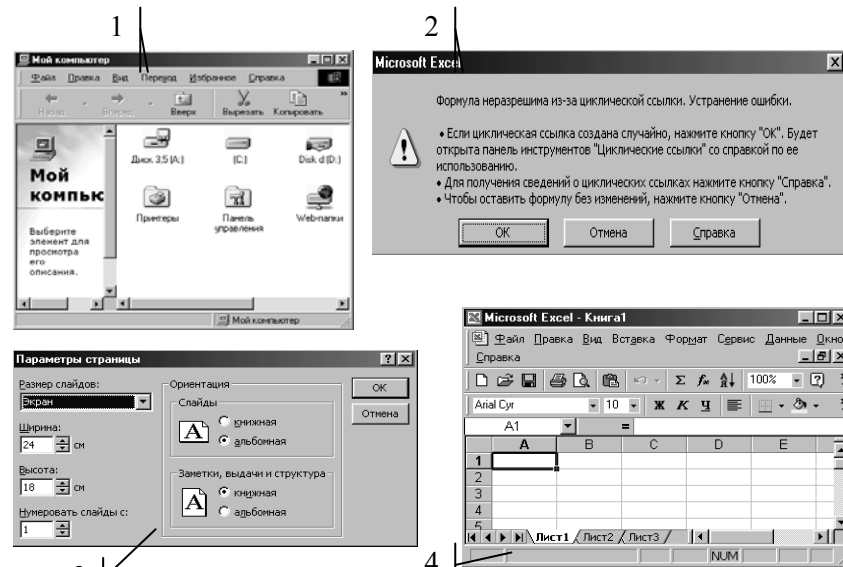
**Завдання.** Створити малюнок згідно зі зразком (мал.5)



Мал.5  
91

г) «Захоплення» мишею якого елемента дозволяє переміщувати вікно? (1)

7. Серед вікон на мал. 3 укажіть діалогове.



Мал. 3

8. Укажіть номер відповідного елемента керування на мал. 4.

- а) перемикач (3)
- б) список, що розкривається (2)
- в) повзунець (5)
- г) елемент керування, який можна «натиснути» (1)
- д) елемент керування, призначений для введення текстів і чисел (7)
- е) елемент керування, який дозволяє встановлювати параметр, незалежно від інших параметрів (4)

система — ..., командний процесор — .... Наведіть власні приклади такої аналогії.

2. Проаналізуйте означення архівації і стиснення інформації в різних підручниках і зобразіть співвідношення обсягів цих понять за допомогою діаграм Ейлера-Венна.

### Запитання для самоконтролю

1. Яка основна мета вивчення теми «Операційна система»?
2. Назвіть суттєві ознаки поняття файлу.
3. Які аналогії доцільно використовувати при введенні поняття шляху до файлу? операційної системи?
4. Назвіть ознаки, за якими класифікують операційні системи.
5. Який підхід лежить в основі методики ознайомлення з Windows? У чому його особливості?
6. Які особливості роботи з різними типами вікон операційної системи Windows? Який із типів є найбільш складним для засвоєння учнями?
7. Охарактеризуйте особливості вивчення дій з файлами і папками.
8. Яким чином доцільно ознайомлювати учнів з правилами розміщення інформації на дисках?
9. У якій формі доречно проводити ознайомлення учнів з комп'ютерними вірусами?
10. Охарактеризуйте внутріпредметні зв'язки при вивчення теми «Основи роботи з дисками».
11. Які основні навчальні завдання вивчення принципів будови глобальної мережі Інтернет?
12. Які особливості введення нових термінів в темі «Глобальна мережа Інтернет»?
13. Які питання теми «Глобальна мережа Інтернет» є, на Вашу думку, світоглядними?
14. Висвітліть методичні особливості навчання роботи з електронною поштою та телеконференціями.
15. Порівняйте особливості навчання пошуку інформації при вивченні операційної системи та глобальної мережі Інтернет.
16. Які редактори рекомендується використовувати при вивченні мови розмітки гіпертексту?

17. Охарактеризуйте особливості ознайомлення учнів з основними елементами HTML.

## Лекція № 11–12

# МЕТОДИКА НАВЧАННЯ ПРИКЛАДНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ЗАГАЛЬНОГО ПРИЗНАЧЕННЯ

## Загальні особливості навчання прикладного програмного забезпечення загального призначення

Як уже зазначалося, сучасний комп'ютер є інструментом для обробки різних видів інформації, а не лише здійснення обчислювальних процесів. Наявність будь-якого інструменту ставить питання про навчання техніки його використання. Якщо ж використання інструменту має масовий характер, то таке навчання одержує статус загальноосвітнього. Визнавши, що комп'ютер є універсальним інструментом обробки інформації, доводиться визнавати також необхідність навчання техніки використання комп'ютера для обробки різних видів інформації. Таким чином, навчання роботи з графічним редактором, текстовим і табличним процесором забезпечує оволодіння учнями відповідної техніки.

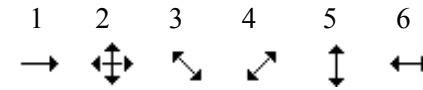
Науково обґрунтована організація процесів одержання необхідних продуктів з вихідної сировини засобами відповідної техніки розглядається технологією. Технологія ґрунтується на науці, нерозривно з нею пов'язана.

У компетенцію технології входить технологічний ланцюжок перетворення вихідної сировини у потрібний продукт, тобто режими обробки і послідовність цих режимів. Саме з цієї точки зору текстовий, табличний процесор, бази даних є інструментами технології. Вивчаючи їх, навчаючись за їх допомогою розв'язувати конкретні задачі, ми опановуємо

- б) Яка з перелічених кнопок відсутня у вікні, розгорнутому на весь екран?

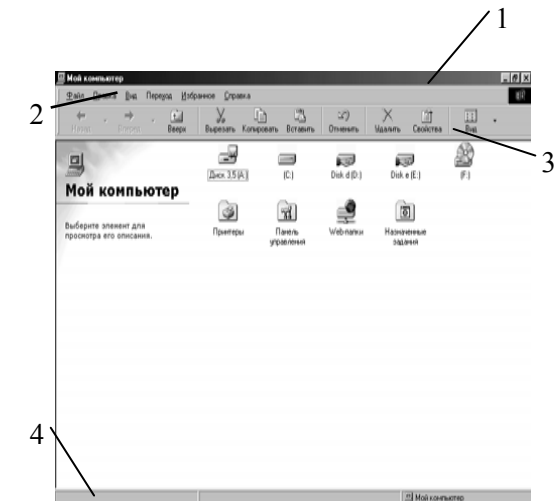
(Кнопка 4)

4. Який вигляд має курсор миші при зміні ширини вікна?



(6)

5. Запишіть назви елементів вікна, позначених на мал. 2 цифрами.



Мал. 2

(1 — рядок заголовку, 2 — рядок меню,

3 — панель інструментів, 4 — рядок стану)

6. Укажіть номер елемента на мал. 2, що відповідає правильній відповіді:

- а) Який елемент містить кнопки, що відповідають діям, які найчастіше виконуються?

(3)

- б) Де вказується назва відкритого об'єкта?

(1)

- в) Де вказується коротка інформація про поточний стан документу чи об'єкту?

(5)



13. Пристрій, призначений для введення графічної інформації.  
(Сканер)
14. Накопичувач на магнітних стрічках.  
(Стример)
15. Проміжна пам'ять між процесором і оперативною пам'яттю, яка прискорює обмін даними між ними.  
(Кеш-пам'ять)

## Приклад тестового завдання з теми «Операційна система»

Тема: «Елементи інтерфейсу операційної системи Windows XP»



Мал. 1  
комп'ютера

1. Укажіть, які підписи мають на мал.1 такі об'єкти (використовуються стандартні значки ОС Windows XP):

а) папка Мої документи (3)

б) папка, у яку вміщуються файли, що знищуються (6)

в) елемент, з допомогою якого можна одержати доступ до дисків

(2)

2. Виберіть із значків на мал.1 ярлики.

(Мої документи, 4, 5)

3. а) Які дії відбуваються при натисненні на такі кнопки?



(1) згорнути у кнопку на панелі задач

2) відновити розміри вікна, 3) закрити вікно,

4) розгорнути на весь екран)

інформаційну техніку й одночасно елементарні акти, так би мовити «атоми», інформаційної технології.

Як уже зазначалося раніше, у вузькому розумінні інформаційні технології — це сукупність методів, засобів і прийомів, що використовуються для пошуку, накопичення, опрацювання, зберігання, подання, передавання різних видів інформації за допомогою засобів обчислювальної техніки і зв'язку, а також способів їх раціонального поєднання із безмашинними процесами опрацювання інформації. Для інформаційно-комунікаційних технологій характерна наявність доброзичливого середовища роботи користувача.

Метою використання ІКТ є якісне формування і використання інформаційного продукту відповідно до потреб користувача.

Виходячи з потреб використання ІКТ, можуть бути сформульовані такі вимоги до знань і умінь учнів:

- учні повинні знати поняття інформаційно-комунікаційних технологій, назви та призначення основних систем програмного забезпечення сучасних ІКТ; повинні вміти зафіксувати предметну галузь і її об'єкти, вибрати програмний засіб і дібрати (або розробити) технологію для розв'язування даної задачі з конкретної предметної галузі; застосовувати пакети прикладних програм навчального призначення і користуватися текстового і графічного редакторів, електронних таблиць, баз даних для розв'язування задач з конкретних предметних галузей.

### Мета вивчення розділу в шкільному курсі інформатики

- 1) ознайомити учнів з поняттям інформаційних технологій;
- 2) сформулювати поняття про технологію як про сукупність методів, засобів і прийомів, що використовуються для розв'язування задач з конкретної предметної галузі;
- 3) оволодіти основними навичками роботи з комп'ютером;
- 4) показати роль і місце інформаційних технологій у сучасному суспільстві.

### Методичні особливості розділу

1. Використати класифікацію інформаційних технологій з метою вибору для вивчення програмних засобів і технологій розв'язування задач з конкретних предметних галузей.
2. Розробити систему вправ щодо використання ІКТ для розв'язування задач з різних предметних галузей. При цьому слід використовувати вправи, які не тільки забезпечать оволодіння основними інструментальними засобами, але й дозволять продемонструвати можливості використання ІКТ в практичній діяльності.
3. Виділити основні дидактичні технологічні одиниці (послідовності кнопок, які треба натиснути, щоб виконати деяку елементарну дію у конкретному програмному засобі).
4. При навчанні розв'язування будь-яких навчальних задач необхідно дотримуватися етапів обчислювального експерименту, що дозволить реалізовувати «навчання через задачі», яке проводиться за схемою: задача — теорія — задача.
5. Використати програмні засоби єдиного інтерфейсу користувача.
6. Бажано використовувати комп'ютерні моделі.
7. Основний метод — метод доцільно дібраних задач та метод демонстраційних прикладів на основі широкого використання інтерактивних технологій.

При вивченні прикладного програмного забезпечення загального призначення учитель повинен усвідомлювати, що вивчення технології у рамках загальної (а не професійної) освіти означає формування в учнів розуміння того, як відбувається технологічний процес, які стадії він проходить. Учитель повинен розрізняти, які знання є загальноосвітніми, а які суто технічними, які знання допоможуть в майбутньому опанувати аналогічні програми. Не варто захоплюватися забезпеченням засвоєння виключно певного набору дій у конкретній програмі. Акцентування уваги на вивчення деталей конкретного технологічного інструменту може бути навіть шкідливим, оскільки інструменти комп'ютерних технологій постійно

## ДОДАТКИ ПРИКЛАДИ МЕТОДИЧНИХ МАТЕРІАЛІВ З РІЗНИХ ТЕМ ШКІЛЬНОГО КУРСУ ІНФОРМАТИКИ

### Приклад інформатичного диктанту з теми «Інформаційна система»

1. Центральний пристрій, який здійснює виконання програми.  
(Процесор)
2. Сукупність пристроїв комп'ютера, з яких він складається.  
(Апаратне забезпечення)
3. Мітка, що вказує поточну позицію на екрані.  
(Курсор)
4. Характеристика процесора, що визначає кількість електричних імпульсів в одиницю часу.  
(Тактова частота)
5. Вид пам'яті, що зберігає дані протягом одного сеансу роботи.  
(Оперативна)
6. Мінімальна одиниця зображення на екрані, точка, що світиться.  
(Піксель)
7. Інша назва графопобудувача.  
(Плоттер)
8. Набір програм, що керують діями комп'ютера.  
(Програмне забезпечення)
9. Вона може бути постійною, до неї відносяться вінчестер і компакт-диски.  
(Пам'ять)
10. Пристрій візуального відображення інформації на екрані.  
(Монітор)
11. Вони бувають матричними, струминними, лазерними.  
(Принтер)
12. Залежно від віддаленості їх поділяють на локальні, регіональні, глобальні.  
(Комп'ютерна мережа)

24. Морзе Н.В. Основи інформатики. Екзаменаційні білети: запитання та відповіді. – К.: Курс, 1999. – 150 с.
25. Морзе Н.В., Мостіпан О.І. Інформатика. Державна підсумкова атестація. 11 клас. – Кам'янець-Подільський: Абетка-Нова, 2003. – 104 с.
26. Перельман Я.И. Живая математика. – М.: ОГИЗ, 1947. – 184 с.
27. Рамський Ю.С., Іваськів І.С., Ніколаєнко О.Ю. Вивчення Web-програмування в школі: Навч. посібник. – Тернопіль: Навчальна книга – Богдан, 2006. – 200 с.
28. Руденко В.Д. Курс інформатики. Основи алгоритмізації і програмування: Навчальний посібник / За ред. В.М. Мадзігона, В.Ю. Бикова. — К. : Фенікс, 2002. — 200 с.
29. Руденко В.Д., Макарчук О.М., Патланжоглу М.О. Практичний курс інформатики / За ред. В.М. Мадзігона. – К.:Фенікс, 1997. – 304 с.
30. Руденко В.Д., Макарчук О.М., Патланжоглу М.О. Комп'ютер та його програмне забезпечення. Курс інформатики (частина 1)/За ред. В.М. Мадзігона — К.: Фенікс, 2001. — 370с.
31. Руденко В.Д., Макарчук О.М., Патланжоглу М.О. Базовий курс інформатики ( Кн. 1: Основи інформатики ). – К. : ВНУ, 2006. – 320 с.
32. Руденко В.Д., Макарчук О.М., Патланжоглу М.О. Базовий курс інформатики ( Кн. 2: Інформаційні технології). – К. : ВНУ, 2006. – 368 с.
33. Семакин И.Г., Плаксин М.А. Информационные системы в базовом и профильном курсах информатики // Информатика. – 2004. – № 34-38.
34. Черняхівський В.В. Збірник задач з основ алгоритмізації: Навч. посіб. для 10-11 кл. загальноосвітн. шкіл. – Львів: ВНТЛ, 1997. – 196 с.

змінюються, удосконалюючись у своїх функціях. Школяр же, із самого початку «прив'язаний» до деякого конкретного програмного продукту, виявляється його заручником. Набагато важливіше, щоб учні розуміли загальні принципи ведення діалогу, вміли використовувати довідкову систему, коректно реагували на повідомлення програм тощо. У майбутньому такі загальноосвітні вміння дозволять швидко пристосовуватися до змін у програмному забезпеченні, орієнтуватися в інших програмах аналогічного призначення.

Для формування узагальнених умінь і навичок доцільно практикувати вправи, при виконанні яких учні формулюють правила виконання тих чи інших дій. Таке «проговорення» алгоритмів (як і при вивченні системного програмного забезпечення) не тільки забезпечує міцність знань, але й сприяє формуванню алгоритмічного мислення, розвиваючи навички зрозумілого й точного опису послідовностей дій, покрокового їх виконання, навчаючи передбачати результати виконання дій. Складання, аналіз, виконання таких алгоритмів є пропедевтикою формального виконання алгоритмів і програм при вивченні основ алгоритмізації і програмування.

Усі прикладні програмні засоби загального призначення слід подавати з позицій — об'єкти та програмні засоби сучасних ІКТ, за допомогою яких користувач досліджує інформаційні об'єкти — інформаційні моделі, тобто варто підходити до висвітлення цих питань через призму діяльності людини: суб'єкт — об'єкт — мета — засоби — діяльність — результати — оцінювання результатів — прийняття рішень.

Об'єктами опрацювання є різні набори даних — текстові, графічні, музичні, таблиці тощо. Засобами для опрацювання об'єктів є прикладні програми, створені спеціально для такого опрацювання.

#### **Схема ознайомлення з прикладним програмним забезпеченням загального призначення**

1. Демонстрація характеристик використання середовища, з'ясування його призначення.
2. Аналіз об'єктів, типів повідомлень, способів отримання результатів опрацювання повідомлень.

3. Ознайомлення з основними складовими інтерфейсу та формування вмінь аналізувати вміст основних його складових.
4. Правила роботи із вбудованою довідковою системою.
5. Ознайомлення з основними функціями та режимами роботи середовища.
6. Вивчення конкретної програми (за окремою схемою).
7. Теоретичне узагальнення основних режимів роботи та функцій середовища.
8. Теоретичне узагальнення на рівні основних вказівок.
9. Виконання аналогічних завдань в середовищі іншої програми такого самого призначення.

Опанування прикладного програмного забезпечення загального призначення відноситься до практичного компоненту змісту навчання інформатики в школі. Тому при вивченні цієї теми важливим є контроль за формуванням умінь і навичок виконання дій у відповідних середовищах. Такий контроль можливий за результатами виконання практичної роботи. Іноді вчителі здійснюють перевірку практичних робіт наприкінці уроку. Проте при такій організації процесу оцінювання у більшості випадків виставлення оцінки є досить поверховим, оскільки учителю важко оцінити правильність виконання роботи, яка характеризується значною кількістю критеріїв. Детальна ж перевірка, тим більше поєднана з опитуванням учнів щодо способів виконання всіх або окремих завдань, вимагає значного часу, при цьому увага вчителя протягом кількох хвилин приділяється виключно одному учневі, решта часто не мають роботи. Тому, як свідчить практика, перевірку правильності виконання практичних завдань краще здійснювати у позаурочний час за результатами виконання робіт, збережених учнями на жорсткому диску. При необхідності перевірити правильність вибору способу виконання дій, виклику діалогових вікон тощо доцільно вводити окремі завдання, які перевіряються вчителем на уроці, решта завдань перевіряються в позаурочний час. Слід відзначити, що при такій організації практичних робіт великого значення набуває формування навичок роботи з файлами, оскільки учні стикаються з необхідністю зберігати файли у визначеному місці, відкривати існуючі файли і зберігати їх у зміненому вигляді зі зміною імені і/або розташування.

12. *Зарецька І.Т., Гуржій А.М., Соколов О.Ю.* Інформатика: Підручник для 10-11 класів загальноосвітніх навчальних закладів. У 2-х частинах. (Част.2) — К.: Форум, 2004. — 288 с.
13. *Караванова Т.П.* Інформатика: Методи побудови алгоритмів та їх аналіз. Не обчислювальні алгоритми: Навч. посіб. для 9–10 кл. із поглибл. вивч. інформатики. — К.: Генеза, 2007. — 216 с.
14. *Караванова Т.П.* Інформатика: основи алгоритмізації та програмування: 777 задач з рекомендаціями та прикладами: Навч. посіб. для 8–9 кл. із поглибл. вивч. інформатики / За заг. ред. М.З.Згуровського. — К.: Генеза, 2006. — 286 с.
15. Концепція програми інформатизації загальноосвітніх навчальних закладів, комп'ютеризації сільських шкіл // Комп'ютер у школі і сім'ї. — 2000. — №3. — С.3-10.
16. *Ланчик М.П., Семакин І.Г., Хеннер Е.К.* Методика преподавания информатики: Учеб. пособие для студ. пед. вузов. — М.: Издательский центр «Академия», 2001. — 624 с.
17. *Малев В.В.* Общая методика преподавания информатики: Учеб. пособие. — Воронеж: ВГПУ, 2005. — 271 с.
18. *Малева А.А.* Практикум по методике преподавания информатики / А.А. Малева, В.В. Малев. — Воронеж: ВГПУ, 2006. — 148 с.
19. Методическое пособие по информатике // Зарецкая И.Т., Семенова Т.В., Соколов А.Ю. — Х.: Факт, 2004. — 192 с.
20. *Морзе Н.В.* Методика навчання інформатики: Посібник для студентів пед. університетів. — К.: Курс, 2002. — 895 с.
21. *Морзе Н.В.* Методика навчання інформатики. Ч.1. Загальна методика навчання інформатики. — К.: Навчальна книга, 2003. — 254 с.
22. *Морзе Н.В.* Методика навчання інформатики. Ч.2. Методика навчання інформаційних технологій. — К.: Навчальна книга, 2003. — 287 с.
23. *Морзе Н.В.* Методика навчання інформатики. Ч.3. Методика навчання основних послуг глобальної мережі Інтернет. — К.: Навчальна книга, 2003. — 196 с.

## СПИСОК ЛІТЕРАТУРИ

1. *Володіна І.Л. Основи інформатики: Підручник для 7 кл. загальноосвітн. шкіл / І.Л. Володіна, В.В. Володін, Ю.О. Дорошенко, Ю.О. Столяров. – Харків: Гімназія, 2007. –*
2. *Володіна І.Л. Основи інформатики : 8 кл. : експерим. навч. посіб. / І.Л. Володіна, В.В. Володін, Ю.О. Столяров. – К. : Генеза, 2008. – 264 с.*
3. *Гаєвський О.Ю. Інформатика: 7–11 кл.: Навч. посіб. – К.: А.С.К., 2005. – 512 с.*
4. *Гейн А.Г. Методика преподавания современного курса информатики. – М.: Педагогический университет «Первое сентября», 2004.*
5. *Глинський М.Я. Інформатика: 8-11 класи. Навчальний посібник для загальноосвітніх навчальних закладів: у 2-х книгах.— Книга 1. Алгоритмізація і програмування. Мова Паскаль.— Львів.: Деол, 2002. — 200 с.*
6. *Глинський М.Я. Інформатика: 8-11 класи. Навчальний посібник для загальноосвітніх навчальних закладів: у 2-х книгах. — Книга 2. Інформаційні технології. 2-е видання. — Львів.: Деол, 2002. — 256 с.*
7. *Ершов А.П. О предмете информатики // Вестник АН СССР. – 1984. – №2. – С.112-113.*
8. *Изучение основ информатики и вычислительной техники: Пособие для учителя / А.В. Авербух, В.Б. Гисин, Я.Н. Зайдельман, Г.В. Лебедев. – М.: Просвещение, 1992. – 302 с.*
9. *Информатика: Підручник для учнів 10-11 кл. загальноосв. серед. шк.. / А.Ф. Верлань, Н.В. Апатова. – К.:Форум, 2001. – 200 с.*
10. *Информатика. Програми для загальноосвітніх навчальних закладів. – Запоріжжя: Прем'єр, 2003. – 304 с.*
11. *Зарецька І.Т., Гуржій А.М., Соколов О.Ю. Інформатика: Підручник для 10—11 класів загальноосвітніх навчальних закладів. У 2-х частинах. (Част.1) — К.: Форум, 2004. — 392с.*

Найбільш складним для багатьох учнів стає зміна саме розташування файлу, особливо у випадках, коли необхідна папка для збереження не відображається у діалоговому вікні (розташована на іншому диску, має з поточною папкою спільний надкаталог). Якщо відповідні навички не сформовані, то можуть втрачатися результати роботи (що унеможливує їх перевірку учителем) або пошкоджуватися вихідні файли. Для забезпечення вихідних файлів від випадкового знищення чи заміщення результатами роботи учнів для підготовлених учителем файлів завдань доцільно встановлювати доступ «лише для читання».

### Особливості вивчення графічного редактора

Ця тема часто вивчається першою серед прикладного програмного забезпечення загального призначення, оскільки:

- 1) предметна галузь добре знайома;
- 2) швидко одержуються результати діяльності;
- 3) кожен учень може виявити творчість та особисті якості;
- 4) не враховуються прогалини у знаннях з інших галузей.

Програмою передбачено вивчення учнями не тільки вказівок конкретної програми, але й ряду теоретичних положень, зокрема, питання про склад і функціонування технічних засобів комп'ютерної графіки, подання зображень у пам'яті комп'ютера. Для свідомого використання засобів комп'ютерної графіки рекомендується ознайомити учнів із апаратними засобами, необхідними для нормальної роботи графічного редактора: графічний адаптер, графічний дисплей, — хоча ці відомості і не є обов'язковими для вивчення.

На початку вивчення теми рекомендується сформувати уявлення про графічні об'єкти, необхідність вивчення графічного редактора для створення і редагування графічних об'єктів. Слід також звернути увагу на розрізнення растрового та векторного принципів побудови графічних зображень. Іноді в літературі також виділяють ще один вид комп'ютерної графіки — фрактальну. При узагальненні слід домагатися, щоб учні усвідомили основні характеристики графічного редактора, не плутали з програмою для побудови діаграм і графіків, навчилися вільно працювати з основними режимами графічного редактора і могли самостійного переходити до опанування інших графічних

редакторів. Учні повинні розуміти не тільки призначення, але й правильний вибір програми.

Вивчення графічного редактора слід починати з вивчення його інтерфейсу та вбудованої системи довідок.

При підготовці практичних завдань слід мати на увазі такі особливості. Недоцільно використовувати тільки «вільні» завдання, оскільки в такому разі не розкриваються можливості графічного редактора. Для вивчення кожної вказівки (2–3 вказівок) необхідно готувати конкретні практичні завдання, що ускладнюються. Як форму подання завдань можна рекомендувати не словесну, а малюнок. Завдання теми можна поділити на 3 групи:

- 1) використання 1–2 інструментів (без пояснення);
- 2) використання кількох інструментів з указуванням порядку виконання;
- 3) комплексні завдання (використання кількох інструментів без пояснення порядку виконання).

При підготовці комплексних завдань учитель повинен чітко уявляти, які знання, уміння і навички використовуватиме учень при виконанні завдання. Наприклад, створення пейзажу, зображеного на мал. 1, за допомогою графічного редактора Paint вимагає використання всього 4 інструментів: *Лінія*, *Прямокутник*, *Еліпс*, *Розпилювач* (залежно від способу виконання може використовуватися також *Залиття*). При цьому переважна частина роботи здійснюється за допомогою інструментів *Лінія* і *Прямокутник*. Таке завдання можливо на проміжному етапі вивчення графічного редактора, але не доречно для систематизації чи перевірки умінь і навичок.

2. У чому полягає операційне мислення?
3. Яка мета навчання основ алгоритмізації? Яким питанням приділяється основна увага?
4. Проаналізуйте особливості вивчення основ алгоритмізації в класах різних профілів.
5. Які компоненти виділяють у змісті навчання основ алгоритмізації? Що означає побудувати алгоритм?
6. Перелічіть основні класи вправ, що використовуються при вивченні основ алгоритмізації і програмування.
7. Порівняйте формальний підхід до вивчення мови програмування та “програмування за зразком”.
8. Перелічіть основні шляхи розв’язання проблеми вибору мови програмування у шкільному курсі інформатики. У чому складність вибору конкретних мов програмування?
9. Проаналізуйте переваги і недоліки найпопулярніших мов програмування високого рівня з точки зору їх використання в шкільному курсі інформатики.
10. У чому особливості середовищ візуального програмування?
11. Які переваги і недоліки використання систем візуального програмування при вивченні основ алгоритмізації і програмування?
12. Які переваги використання навчальної алгоритмічної мови в шкільному курсі інформатики?
13. Які особливості ознайомлення учнів з поняттям змінної?
14. Яким питанням слід приділити увагу при вивченні основних понять мови програмування?
15. Назвіть типові помилки при вивченні вказівки розгалуження. Які вправи доцільно використовувати для їх попередження?
16. У чому складність розуміння учнями циклічних операторів?
17. На які моменти слід звернути увагу при вивченні різних видів циклів?
18. Охарактеризуйте типові помилки при вивченні табличних величин.
19. Які питання стилю програмування доцільно розглядати при вивченні різних тем розділу “Основи програмування”?
20. Проаналізуйте різні підходи до визначення місця теми “Звернення до процедур і функцій”.
21. Які основні труднощі виникають в учнів при вивченні теми “Звернення до процедур і функцій”?

Обмеження, що накладаються мовою Borland Pascal на розмір використовуваної пам'яті і виникли в результаті орієнтації на DOS-модель пам'яті, не витримують жодної критики (64 Кб на всі глобальні змінні, стільки ж на розмір стека і 200–400 Кб на динамічні змінні, рідко використовувані в шкільних курсах). Ну і, нарешті, неможливість простого і швидкого створення віконних додатків з різними елементами інтерфейсу, що відповідають сучасним вимогам.

Ще один аспект, на якому слід зупинитися при виборі мови програмування для використання в школі, — законність використання. Для мов групи Паскаль вільно поширюваними є BP 7.0 і FreePascal. Останній компілятор разом з середовищем програмування копіюють (на жаль, не завжди вдало) всі особливості BP 7.0, знімаючи при цьому обмеження на розмір використовуваної пам'яті. Слід також зазначити, що на багатьох офіційних змаганнях з програмування, зокрема, на міжнародній олімпіаді з інформатики, і інтернет-сайтах, які перевіряють розв'язання задач з програмування, використовується саме цей компілятор. Для мов C і C++ вільно можна використовувати компілятор gcc (gpp) і середовище розробки RHIDE. Для використання ж таких середовищ візуального програмування, як Visual Basic, Borland Delphi або Visual C++, необхідне придбання ліцензії. При цьому для ранніх версій будь-якого із згаданих програмних продуктів ліцензія обійдеться істотно дешевше, а на початкових етапах навчання програмуванню, якими в школі переважно й обмежуються, підійде практично будь-яка версія.

### **Завдання для самостійної роботи**

1. Для створення мотивації вивчення основ алгоритмізації важливо не обмежуватися обчислювальними алгоритмами, а демонструвати алгоритми, які використовуються (часто неусвідомлено) в інших галузях діяльності, при розв'язуванні задач повсякденного життя. Наведіть приклади алгоритмів (лінійних, розгалужених, циклічних), які учні можуть використовувати у повсякденному житті чи своїй навчальній діяльності.

2. Запропонуйте систему вправ на пошук помилок у програмах (фрагментах програм) для однієї з тем розділу «Основи алгоритмізації та програмування».

### **Запитання для самоконтролю**

1. Які вміння і навички необхідні параметричному користувачеві для успішного спілкування з комп'ютером? У яких ще галузях необхідні ці вміння?



Мал.1

Значно більша кількість інструментів повинна бути використана для виконання зображення на мал. 2. Зокрема, мішок Діда Мороза будується за допомогою *Олівеця*, *Пензля* і *Розпилювача*, подарунок — *Прямокутника із заокругленими кутами* та *Лінії*, сніжинки — *Еліпса* і *Лінії*, рамка може бути створена *Прямокутником* і *Многокутником*, текст додається за допомогою відповідного інструмента. Крім того, необхідною є зміна параметрів інструментів (товщина лінії, розмір символів, накреслення), для акуратної побудови сніжинок доцільно використовувати інструмент *Масштаб*, раціональним шляхом виконання завдання є копіювання сніжинок та використання розтягу для однієї з них. Таким чином, для правильного розв'язання завдання вимагається використання більшості інструментів і вказівок графічного редактора.



Мал. 2

При оцінюванні практичних робіт слід мати на увазі, що один і той самий результат може бути досягнутий за допомогою різних інструментів. Так, на мал. 2 куточки рамки можуть бути створені як за допомогою інструменту *Многокутник*, використовуючи стиль заливки *без контуру*, так і за допомогою інструменту *Лінія* з наступним заливкам кольором лінії (можливий також варіант використання інструменту *Многокутник*, але із стилем конкур без заливки, заливка здійснюється відповідним інструментом).

### Особливості вивчення текстового процесора

Основна мета під час вивчення теми — ознайомити з можливостями використання текстового процесора як програми, призначеної для опрацювання текстових повідомлень за допомогою комп'ютера. Об'єктом опрацювання в текстовому процесорі є текст (можливо, із вбудованими об'єктами).

Світоглядне значення мають такі питання:

- виникнення текстового редактору для спрощення роботи з текстом;
- функції текстового редактора: введення тексту до пам'ятовуючих пристроїв, перегляд, опрацювання та відтворення тексту;

надзвичайній популярності якого сприяв розвиток сімейства компіляторів фірми Borland, починаючи від Turbo Pascal і закінчуючи Delphi. Не всі новини, привнесені фахівцями Borland в класичний Паскаль, здаються Вірту виправданими, проте і вони зокрема привели до того, що Паскаль довгі роки посідав одне з провідних місць серед професійних мов розробки різних додатків, а проект Delphi надав йому нове дихання.

З методичної точки зору Паскаль дійсно добре підходить на роль навчальної мови. Він дозволяє познайомитися з більшістю понять сучасного програмування, освоїти як різні типи, так і структури даних. Програми на Pascal легко читаються, а один з найважливіших принципів сучасного програмування — легкість для читання більш важлива, ніж стислість коду (звичайно, якщо це не приводить до уповільнення роботи програми більш ніж удвічі-утричі), адже над сучасними програмними комплексами працюють цілі колективи програмістів, і їм необхідно швидко орієнтуватися в коді один одного. Не випадково при описі різних алгоритмів в більшості навчальної літератури, у тому числі і західної, використовується саме Паскаль або схожий з ним псевдокод. Легкість для читання Паскаль дуже доречна і для вчителя при перевірці програм, написаних школярами. Крім того, синтаксис мови влаштований так, що своєю строгістю фактично вимушує писати правильні програми. Це вигідно відрізняє Паскаль, наприклад, від C++. Не випадково на факультеті обчислювальної математики і кібернетики Московського державного університету ім. М.В. Ломоносова курс програмування починається саме з вивчення мови Паскаль, причому до того, як він відверіти, що середовище Borland Pascal 7.0 було випущено ще 1993 року, фірмою Borland воно давно не підтримується, внаслідок чого все менше сумісне з сучасними високошвидкісними комп'ютерами і операційними системами: першим перестав коректно працювати улюблений школярами стандартний бібліотечний модуль CRT, що безпосередньо звертається до апаратних засобів комп'ютера; потім програми, що "зациклилися" або просто довго працюють, стало неможливо перервати з середовища програмування — для цього доводиться вдаватися до засобів операційної системи, що іноді приводить до втрати незбережених даних; нарешті, виконання з середовища команди збереження Save (F2) і реальний запис операційною системою того ж файлу на жорсткий диск з кешпамяті настільки рознесе в часі, що інколи виникають неприємні колізії (слід відзначити, що сучасні програми, що мають функції редагування і збереження, подібної проблеми не



змінної  $b$ , потім значення  $a$  порівнюється з нулем). Інший приклад: вираз виду

$$a < b < c.$$

є некоректним для числових і символічних типів мови Паскаль і не буде пропущений компілятором, але цілком сприймається компіляторами з мови C (C++), проте має зміст, відмінний від подвійної нерівності в математиці: спочатку виконується порівняння  $a$  і  $b$ , результатом якого виявляється або 1 (істинно), або 0 (хибно). Потім вже це число (1 або 0) порівнюється із  $c$ . Ситуацію, у якій таке порівняння має сенс, придумати практично неможливо, крім уже згаданих вище складнощів, що виникають при написанні програм на C++, особливо на початковому етапі його освоєння, можна зіткнутися зі складнощами технічного порядку: налаштування середовища програмування вимагає певних навичок і часто нетривіальне в умовах шкільної локальної мережі, а витратити на неї дорожочісний час уроку — не дозволено розкіш. Засоби налагодження при цьому не такі зручні, а повідомлення компілятора про різного роду помилки не такі інформативні, як в середовищах програмування для мови Паскаль фірми Borland. Крім того, більшість відомих компіляторів з мови C++ містить ті або інші помилки, внаслідок чого навіть найпростіші програми час від часу на практиці виявлялися непридатними, а вставка в них, наприклад, незначимого оператора на зразок  $i = i$ ; приводила до бажаного результату. Список же відомих помилок компілятора BP 7.0 дуже короткий, і жодна з таких помилок при розв'язуванні навчальних задач зустрітися не може. Останнє дуже важливе на початковому етапі навчання програмуванню: учень повинен чітко розуміти, що коли його програма не працює, то це відбувається вже ніяк не з вини компілятора або операційної системи.

Проте слід відзначити, що мова C++ досить успішно викладається в окремих школах.

### **Паскаль**

Як відомо, мова Паскаль була створена на початку 70-х років минулого сторіччя видатним фахівцем у галузі комп'ютерних наук Ніклаусом Віртом саме як мова для вивчення програмування. Основою для побудови синтаксичних конструкцій цієї мови послужив широко поширений у той час Algol (algorithimical language). Вірт продовжив свою роботу над створенням методично обґрунтованої мови програмування, запропонувавши громадськості спочатку мову Modula-2, а потім об'єктно-орієнтований Oberon. Проте останні дві мови не одержали широкого розповсюдження на відміну від мови Паскаль,

- зв'язок оперативної пам'яті із середовищем при роботі з документом (екран можна вважати своєрідним вікном, через яке користувач переглядає текст, що зберігається в оперативній пам'яті; крім основної пам'яті, де зберігається текст, використовуються ще й додаткові «аркуші» для збереження видалених фрагментів, вмісту буферу, довідник, словник).

Після демонстрації можливостей текстового процесора рекомендується повторити питання кодування текстової інформації, у тому числі поняття кодової таблиці. Далі доцільно розглянути функції текстового процесора і тільки після цього переходити до вивчення режимів роботи та вказівок конкретного текстового процесора.

Оскільки текстовий процесор вивчається першим з офісних програм, то слід приділити увагу формуванню вмінь, загальнозначущих для всіх прикладних програмних засобів загального призначення: запуск, завершення роботи програми, збереження, пошук файлів, правила роботи з інтерфейсом, вигляд курсору в різних режимах, робота з буфером, фрагментами, об'єктами, робота з «підлеглими» вікнами, використання довідки, шаблонів тощо. Необхідно ще раз звернути увагу учнів на поняття інтерфейсу, запропонувати побудувати схему інтерфейсу стандартного текстового процесора (меню, рядок стану, рядок повідомлень, робоче поле).

Доцільно також ознайомити учнів із змінами зовнішнього вигляду курсору. Враховуючи досить великий обсяг матеріалу, на початку можна пропонувати учням довідкові таблиці.

Ознайомлення із вказівками текстового процесора доцільно починати із вказівок відкриття файлів (у тому числі й пошук документів) та їх збереження. Виконання завдань полегшується, якщо робота з файлами досить детально розглядалася при вивченні графічного редактора.

Після вказівок роботи з файлами розглядаються правила переміщення по тексту, правила роботи з основними об'єктами, їх виділення. При ознайомленні з процесом редагування доцільно підкреслити, що редагування можна виконувати вручну та використовуючи відповідні вбудовані засоби.

Слід звернути увагу на розрізнення понять редагування і форматування, оскільки нерідко учні під редагуванням

розуміють будь-які зміни, що відбуваються з документом: як зміни змісту (тобто власне редагування), так і зміни зовнішнього вигляду (форматування).

При вивченні форматування доцільно, продемонструвавши відповідні приклади, навчити за зовнішнім виглядом тексту розрізняти використані параметри форматування. Спочатку доцільно навчити учнів змінювати зовнішній вигляд символів, а потім перейти до вивчення правил форматування абзаців, звернувши при цьому увагу на особливості тлумачення цього терміну в інформатиці.

Деякими навчальними програмами передбачено також навчання учнів використовувати шаблони документу, працювати з певною структурою. Для цього необхідно ознайомити учнів із форматуванням тексту за допомогою стилів. Слід відзначити, що поняття стилю не є очевидним для учнів, оскільки переваги використання стилів відчутні при роботі з великими текстами, а невеликі тексти (які найчастіше й використовуються у навчальних цілях) неважко відформатувати, безпосередньо встановивши необхідні параметри для конкретних абзаців. Тому навчання форматування за допомогою стилів доцільно проводити, використовуючи відносно великі за обсягом тексти. При вивченні із редагуванням і форматуванням документів доцільно не тільки формувати навички застосування вказівок текстового редактора, але й ознайомлювати учнів із правилами введення тексту (розбиття тексту на рядки, використання пробілів тощо), оформленням типових документів.

Досить великий інтерес учнів викликає робота із малюнками та фігурним текстом у середовищі текстового редактора. Дещо складнішими виявляються питання форматування об'єктів (визначення положення, обтікання, зміни заливки тощо), виконання дій над об'єктами (зокрема, групування).

При формуванні навичок роботи в текстовому редакторі слід уникати «кнопочного навчання», вивчаючи конкретні вказівки у конкретному текстовому редакторі, пояснювати учням як опрацьовується текст.

### Основні типи завдань

1. Ввести текст та подати у певному вигляді. Недоліком таких завдань є складність самооцінки, втрата часу на

програмування з GoTo призводить до формування поганого стилю, виправити який надалі дуже складно. Проте еволюція мови, починаючи з Quick Basic і закінчуючи Visual Basic, призвела до того, що сама мова стала мало відрізнятися, наприклад, від мов Pascal і Delphi відповідно, і категоричні вислови про непридатність його використання в навчальних цілях навряд можна вважати коректними. Проте при виборі цієї мови слід враховувати, що на багатьох олімпіадах високого рівня з інформатики і програмування в списку допустимих мов програмування Basic переважно відсутній зовсім (міжнародна олімпіада школярів з інформатики, студентський чемпіонат світу з програмування).

Якщо ж звернутися до статистики використання тих або інших мов програмування на міжнародному ринку праці, то Visual Basic в даний час посідає почесне друге місце, поступаючись лише Visual C++. Проте це далеко не так у нас в країні, де аналогічну Visual Basic нішу програміста посідає Delphi. Крім того, компанія Microsoft заявила про припинення випуску нових і підтримки старих версій мови Visual Basic у зв'язку з переходом на платформу .NET. Єдиним незаперечним аргументом на користь вибору даної мови як базової для вивчення програмування взагалі є те, що саме вона використовується для написання макрокоманд в сучасних офісних додатках. І якщо вивчення, наприклад, електронних таблиць ведеться на рівні, що допускає вільне володіння цією мовою, то вибір Visual Basic для освоєння алгоритмізації і програмування може бути виправданим. В даному випадку виявляється можливим побудувати інтегрований курс одночасного освоєння як інформатичних технологій, так і програмування.

Переваги й недоліки використання цих мов у шкільному курсі інформатики розглядатимемо разом, оскільки основною відмінністю C++ від C є підтримка об'єктно-орієнтованої парадигми, що залишається поза увагою при вивченні процедурного програмування. C розроблена професійними програмістами як зручна для них мова, що визначає як переваги, так і недоліки її вибору для вивчення в школі. Ця мова надає можливість компактного, ефективного розв'язування задачі, але наслідком цього є велика ймовірність помилок, що важко визначаються. Для їх виключення необхідний високий ступінь самоконтролю та майстерності програміста. Чого варта, наприклад, конструкція `if (a = b)...`, яка, будучи синтаксично коректною, має в C досить далеку від інтуїтивного значення семантику (змінній *a* надається значення

набагато меншою, оскільки типові задачі на використання підпрограм — факторіал, степінь — вимагають знайомства з циклами або розгалуженнями. У той же час раннє ознайомлення з поняттям підпрограми дозволяє з самого початку вивчення програмування виробляти правильний стиль розробки програм — програмування «згори донизу», але лише в тому випадку, коли протягом вивчення наступних тем учитель звертається до використання підпрограм, демонструючи правила розробки алгоритму.

Тема є досить складною для розуміння школярами. У першу чергу складнощі пов'язані із реалізацією методу покрокової деталізації: задача по суті розбивається на ряд підзадач, кожна з яких розв'язується незалежно від інших, а взаємодія забезпечується передачею параметрів. Утруднення в цьому випадку викликає і саме розбиття задачі на підзадачі, і абстрагування від змісту задачі при розв'язуванні підзадач.

Іншою проблемою є нерозуміння учнями механізму передачі параметрів та можливості на його основі викликати одну підпрограму кілька разів з різними параметрами. Тому досить типовою помилкою є спроба описувати стільки підпрограм, скільки різних наборів параметрів передбачається для виклику цієї підпрограми. До цієї проблеми близько стоїть і проблема розрізнення параметрів-значень і параметрів-змінних.

Поняття підпрограми пов'язане з поняттям рекурсії. Проте якщо підпрограми вивчаються до операторів розгалуження, то ознайомити з поняттям рекурсії в рамках теми «Звернення до алгоритмів та функцій» не вбачається можливим, адже для коректної реалізації рекурсії необхідно вказати умову виходу з неї.

## **Самостійна робота студентів**

### **Матеріал для самостійного опрацювання**

#### **Методичні особливості різних мов програмування**

##### ***Basic, Quick Basic і Visual Basic***

Довгі роки вважалось, що мова програмування Basic є з методичної точки зору непридатною для навчання навіть засадам програмування майбутніх професіоналів, оскільки

набір. Такі завдання рекомендується використовувати при узагальненні та повторенні.

2. Завантажити файл та подати його у конкретній формі, вимоги до зовнішнього вигляду (зразок, очікуваний результат опрацювання повідомлень) тексту учні одержують роздрукованими. За рахунок завантаження файлу з диску більше часу можна витратити на опрацювання повідомлень, а не введення тексту. При цьому завдання можуть бути подані з різними рівнями підказок щодо їх виконання: зразок і послідовність вказівок для виконання завдання;
  - 2) надається зразок і список вказівок для його одержання, але алгоритм виконання завдання відсутній;
  - 3) пропонується зразок та узагальнюючі вказівки;
  - 4) пропонується тільки зразок, вказівки виконання завдання відсутні.

Для формування навичок виконання дій у текстовому редакторі рекомендується індуктивний підхід із використання методу доцільних задач. Учні, розв'язуючи навчальні задачі, здобувають знання про правила виконання тих чи інших дій. Для узагальнення вивченого матеріалу рекомендується, щоб після кожної вправи учні вголос промовляли порядок виконання, вибирали раціональніший шлях виконання завдання.

На лабораторних роботах можна пропонувати письмово оформлювати узагальнюючий висновок. При цьому повинні бути чіткі вимоги до оформлення роботи, бажано без конкретних вказівок — узагальнені речення.

При доборі завдань для практичних робіт слід мати на увазі, щоб усі елементи завдання можна було перевірити після завершення роботи (у крайньому разі передбачити пункти, що вимагають продемонструвати проміжні результати роботи вчителю). Розглянемо приклад фрагменту практичної роботи:

1. Відкрити файл *Завдання.doc*.
2. Замінити всі літери «z» у тексті літерою «я» за допомогою відповідної вказівки текстового процесора.
3. Для другого абзацу тексту встановити шрифт *Arial*.
4. Встановити синій колір для символів першого абзацу.
5. Для всього тексту встановити шрифт *Impact*, розмір — 12.

З наведеного фрагменту практичної роботи легко бачити, що після завершення виконання всіх п'яти завдань неможливо перевірити правильність виконання пункту 3 (встановлення шрифту *Arial* для одного з абзаців), адже після виконання пункту 5 весь текст матиме гарнітуру *Impact*. Також при правильному виконанні пункту 2 за кінцевим документом не можна з'ясувати, як саме він був виконаний: з використання вказівки *Правка\Замінить...* чи вручну. Проте якщо кількість замін у тексті значна, то ручна заміна вимагатиме багато часу, і учень не встигне виконати інші завдання. Внаслідок цього учитель не матиме можливості з'ясувати рівень сформованості інших умінь і навичок. Тому завдання такого типу варто розташовувати в кінці роботи або контролювати хід їх виконання, не дозволяючи втрачати надто багато часу.

Після ознайомлення учнів з правилами роботи в середовищі текстового процесора доцільно проводити підсумкові комплексні лабораторні роботи, давати учням індивідуальні творчі завдання.

## Особливості вивчення табличного процесора

У методичній літературі зустрічається різне розуміння терміну «електронна таблиця». Раніше електронною таблицею називали програму, призначену для опрацювання даних, поданих у табличному вигляді. З часом електронна таблиця почала розглядатися як об'єкт опрацювання, а засіб опрацювання називатися табличним процесором. Сьогодні провідні методисти вважають, що електронна таблиця повинна розглядатися як сховище (частина запам'ятовуючого простору) для зберігання відповідним чином структурованої сукупності даних різних типів. Проте в літературі (у тому числі й підручниках) зустрічається і розуміння електронної таблиці як спеціального виду програми. Основна мета програми — це візуальна презентація електронних таблиць на першому етапі загальної схеми вивчення прикладного програмного забезпечення загального призначення полягає у формуванні в учнів уявлень про можливості подання і форматування текстової та числової інформації, що зберігається в електронній таблиці, виконання автоматизованих обчислень та зміну обчислювальних результатів. Важливо продемонструвати

іншому разі модифікація програми (наприклад, зміна вимірності тих же масивів) стає дуже трудомісткою або й неможливою. Використання констант відноситься до стилю програмування. До цих же питань можна віднести і питання про те, чи слід при описі масивів вводити для них у розділі опису типів відповідні нові типи даних. Як відомо, у мові Паскаль, як і в ряді інших мов, дозволяється описувати змінні типу масив як безпосередньо при описі змінних, так і за допомогою введених раніше типів. З точки зору стилю програмування, особливо при розробці великих програм, нерідко вважають більш доцільним використовувати другий спосіб. Справа в тому, що відповідність типів у цій мові перевіряється не за їх фактичним описом, а за назвою, тому, наприклад, не вважатимуться змінними одного типу змінні, описані таким чином.

```
var
    a:array[1..10] of real;
    b:array[1..10] of real;
```

На початкових етапах роботи з масивами зі складнощами, пов'язаними з подібним описом, можна і не зіткнутися. Адже єдина доступна операція для змінних типу масив, які належать до одного типу, — це операція присвоєння одного масиву іншому, але вона рідко використовується. Реальні проблеми виникають при передачі масивів як параметрів процедур та функцій. Тут введення спеціального типу для подальшого опису є необхідним. Але якщо не планується на уроках до написання великих і складних програм, то такий опис можна не використовувати.

## Звернення до процедур та функцій

Місце теми «Звернення до процедур та функцій» у розділі «Основи алгоритмізації та програмування» не є чітко визначеним. Дехто пропонує вивчати підпрограми після вивчення основних конструкцій мови (у тому числі операторів повторення і розгалуження), а іноді навіть і після вивчення структурованих типів даних. У такому разі під час вивчення теми можна використовувати досить значний перелік задач. Інша точка зору, підтримана в діючих програмах, передбачає ознайомлення з підпрограмами на ранніх етапах вивчення мови програмування, одразу після вивчення основних понять мови. У такому випадку кількість задач, які можуть бути розв'язані, є

Доцільно також порівняти цикли з перед- та післяумовою:

- умова циклу ПОКА є умовою продовження циклу, а умова циклу ДО — умовою його завершення;
- цикл з передумовою може не виконатися жодного разу, а цикл з післяумовою обов'язково виконується принаймні один раз;
- службові слова циклу з постумовою відіграють роль операторних дужок.

### Табличні величини (масиви)

Знайомство з таким об'єктом, як масив пов'язано з подоланням учнями чергового психологічного бар'єру. І даний бар'єр тим вищий, чим нижча математична підготовка школярів. На початкових етапах роботи з масивами учні часто плутають такі поняття як «індекс елемента масиву» та «значення елемента масиву», навіть коли при першому поясненні матеріалу створюється враження, що вони у даних термінах розібралися. Тому необхідно виконувати достатню кількість вправ на розрізнення цих понять. У цьому розумінні корисно розглядати задачі на обробку нецілочислових масивів. Адже саме в останніх і самі елементи, і їх індекси найчастіше збігаються. **Слід звернути увагу на те, що всі елементи масива повинні мати однаковий тип і особливо на те, що кількість елементів масиву повинна визначитися під час його опису і надалі не змінюється.** При цьому варто пояснити школярам, що обробка масивів, розмір яких слід визначати при виконанні програми, програмується таким чином: при описі відповідної змінної типу масив пам'ять відводиться від масив максимального розміру, реальна розмірність запитується програмою, а всі цикли з обробки масивів використовують уже значення цієї реальної змінної. При цьому максимальний розмір визначається або за умовою задачі, або самостійно з позицій здорового глузду.

При використанні масивів також доцільно звернути увагу на використання констант у текстах програм. У правильно написаній програмі кожна константа може зустрічатися тільки один раз: або в розділі констант, або на початку програми (підпрограми) для надання значення відповідній змінній. Значення всіх похідних констант повинні записуватися тільки як вирази від введених раніше імен констант або змінних. В

не тільки різні можливості форматування окремих об'єктів, а й динамічність електронної таблиці, тобто автоматичну зміну результатів обчислень при зміні аргументів.

Для з'ясування призначення і особливостей табличного процесора рекомендується разом з учнями визначити суттєві та несуттєві ознаки цієї програми. Учні повинні зрозуміти, чому таблиці називаються електронними, чим вони відрізняються від таблиць, які використовуються в текстовому редакторі.

Ознайомлення з інтерфейсом табличного процесору полегшується, якщо використовуються програмні засоби єдиного інтерфейсу користувача і при вивченні текстового процесора сформовано міцні знання учнів з цього питання. У цьому випадку залишається пояснити особливості інтерфейсу табличного процесора, звернувши увагу на робоче поле та об'єкти електронної таблиці (комірки, аркуші, діаграми тощо). На цьому етапі доцільно провести порівняння інтерфейсів текстового і табличного процесорів, а також звернути увагу на вигляд **Важливо вивчити** учнів знаходити комірку за її адресою, визначати вміст комірки. При визначенні вмісту комірки слід звернути увагу, що в комірці може відображатися лише частина її вмісту, повний вміст можна побачити в рядку введення. Слід пояснити особливості відображення тексту, якщо сусідня комірка порожня, щоб запобігти помилок некоректного об'єднання комірок чи пропуску окремих комірок при заповненні **Для подальшої роботи з табличним процесором необхідно навчити виділяти як окремі комірки, так і діапазони комірок.** При цьому слід звернути увагу на те, що активна комірка не маркується при виділенні, інакше діти намагаються «зафарбувати» всі комірки діапазону.

Можливості форматування комірок можна запропонувати учням самостійно дослідити при виконанні спеціальних завдань, а потім провести узагальнення розглянутих можливостей. Особливу увагу слід звернути на числові формати, оскільки інші параметри форматування аналогічні відповідним параметрам форматування комірок у середовищі текстового процесора.

Важливою особливістю табличного процесору є можливість проведення обчислень. Тому при вивченні даної теми необхідно особливу увагу звернути саме на виконання обчислень. Рекомендується використовувати зразки таблиць (можливо, з

підказками), у яких уже внесено результати обчислень, що дозволить учням контролювати правильність своїх дій. Крім такого підходу, корисно також розбирати разом з учнями перехід від деякої прикладної задачі (інформаційної потреби користувача) до електронної таблиці. Це дозволяє посилити прикладну спрямованість знань (адже в практичній діяльності користувач не завжди зустрічається із готовою табличною моделлю), а також розвивати змістову лінію інформаційного моделювання. При цьому важливо, щоб задачі були цікавими для учнів; правильний добір задач дозволяє також реалізовувати міжпредметні зв'язки, вирішувати виховні завдання уроку.

Наведемо приклад такої задачі. «На двох комп'ютерах різної швидкодії реалізовані два різні алгоритми впорядковування чисел. Алгоритм, реалізований на комп'ютері з швидкістю 10 000 операцій за секунду, вимагає виконання  $1000\log_2 n$  операцій, де  $n$  — кількість чисел. Алгоритм, реалізований на комп'ютері з швидкістю 100 000 000 операцій за секунду, вимагає виконання  $n^2$  операцій. Скласти таблицю, що відображає залежність часу сортування від кількості  $n$  чисел на кожному з комп'ютерів ( $n$  змінюється від 1 000 до 20 000 з кроком 1 000). Побудувати графіки такої залежності в одній системі координат». Запропонована задача не тільки вимагає застосування операцій і функцій табличного процесора, але й сприяє формуванню алгоритмічної культури, оскільки одержана у результаті розв'язування задачі діаграма наочно демонструє, що ефективність алгоритму нерідко має більш суттєве значення, ніж якість апаратного забезпечення. Крім розрахунків за формулами, побудови діаграм, учням слід визначити розташування даних у таблиці.

Дещо складнішим є розв'язування такої задачі, ідея якої запозичена з книги «Жива математика» відомого популяризатора науки Я.І.Перельмана. «Ще на початку минулого століття деякі підприємці застосовували досить оригінальний спосіб розповсюдження свого товару, пропонуючи придбати, наприклад, велосипед за 10 крб. замість 50. Проте за 10 крб. вислався не сам велосипед, а лише 4 квитки, які потрібно було продати по 10 крб. своїм 4 знайомим. Зібрані таким чином 40 крб. потрібно було надіслати фірмі, і тільки тоді

відступи для розміщення вкладених структур. Деякі методисти рекомендують з цього моменту не приймати програми школярів, які написані суцільним текстом.

### **Вказівка повторення**

Оператори циклу є найбільш значимими конструкціями у мовах програмування. Саме вони роблять мови програмування високого рівня більш привабливими у порівнянні, наприклад, з електронними таблицями. Вони найбільш складно сприймаються школярами. Для того, щоб навчитися легко застосовувати різні оператори циклу при розв'язуванні тих чи інших задач, учням доводиться долати психологічний бар'єр, оскільки майже неможливо провести аналогію між даними конструкціями і вже знайомими їм елементами з інших галузей знань. мові Паскаль існує два принципово відмінних типи операторів циклу: оператор циклу з параметром та оператори циклу з умовами. Перший із них сприймається легше, використання його у практиці програмування дещо простіше. Але з іншого боку, частина механізму його дії прихована від програміста, а використання в операторі циклу значень його параметра для більшості учнів може здатися неочевидним, що може призводити до поверхового використання. Крім того, при вивченні циклу з параметром слід зазначити такі особливості:

- цей оператор зручно використовувати, якщо відома кількість повторень;
- початкове та кінцеве значення циклу обчислюються на початку виконання оператора і в наступному не обчислюються заново (ця особливість може викликати труднощі в подальшому при вивченні рядкових величин, коли кінцеве значення задається як довжина рядка, і остання змінюється у процесі виконання циклу);
- не рекомендується змінювати параметр циклу в тілі циклу (хоча формально це допустимо).

Важче сприймаються цикли з умовами (перед- та після-). Слід звернути увагу на ситуації «зациклення», які можуть виникати у випадку, коли значення змінних, що входять в умову, не змінюються в тілі циклу.

- ще раз проговорити алгоритм виконання програми для учні, які не зрозуміли цю теми на перших уроках;
- звернути увагу учнів на типові помилки, виявлені вчителем.

### Вказівка розгалуження

Умовні оператори звичайно досить легко сприймаються учнями. При їх вивченні слід звертати увагу на такі моменти:

1. Відповідно до синтаксису Паскаля символ «;» є роздільником двох операторів, тому він не ставиться перед службовим словом **else**.
2. Оператор на кожній гілці повинен бути тільки один. Якщо необхідно виконати більше однієї дії, то слід використовувати складений оператор.
3. Умовний оператор є частиною програми, тобто перед і після вказівки можуть бути (і у більшості випадків присутні) інші оператори. Тому після виконання оператора відбувається не завершення програми, а перехід до наступного оператора.
4. При розробці програми слід передбачити всі можливі варіанти дій, а при виконанні програми для заданого набору вхідних даних виконується одна (!) із «гілок» оператора.
5. Необхідно розрізнити операцію надання значення ( $:=$ ) та операцію перевірки рівності ( $=$ ). Для формування поняття простої умови та правил її запису доцільно використовувати завдання таких типів:
  - описати задані умови за правилами алгоритмічної мови;
  - перекласти умову з «довільної» форми на алгоритмічну мову;
  - знайти помилку в умові.
6. Слід звертати увагу на пріоритет операцій при виконанні складених умов, особливо порядок виконання логічних операцій та операцій порівняння.

Уведення умовного та особливо складеного операторів — гарний привід поговорити з учнями про стиль написання програм. Для мови Паскаль існують кілька розповсюджених способів запису структурованих програм, але всі вони містять

надсилався велосипед; отже, покупцеві він обходився, дійсно, всього лише у 10 крб., решта 40 крб. сплачувалися ж не з його кишені. Квитки, придбані за 10 крб., можна було обміняти у фірми на 5 таких самих квитків; іншими словами, їх власник мав можливість зібрати 50 крб. для купівлі велосипеда, який йому обходився тільки в 10 крб., тобто у вартість квитка. Нові володарі квитків у свою чергу одержували від фірми по 5 квитків для подальшого розповсюдження і т.д. Припустивши, що безпосередньо у фірми придбали квитки 50 чоловік, визначити, скільки людей буде залучено до акції через 10 її «турів». Скільки з цих людей одержать велосипеди? Чи зможуть придбати велосипеди всі ті, хто має квитки?»

Виховне значення задачі полягає в тому, що у ній демонструється принцип афер, аналогічних до фінансових пірамід. Розрахунки, проведені учнями самостійно, показують, що уже через 10 «турів» акції нею буде охоплено близько 100 млн. чоловік (більше, ніж населення двох України), з яких тільки близько 20 млн. одержать бажаний велосипед. Таким чином, школярі мають змогу переконатися, що акції такого типу спричиняють збитки багатьом із її учасників, які виплачують фірмі різницю між вартістю товару та оплатою за них кожного покупця. Розв'язування розглянутої задачі вимагає побудови інформаційної моделі. Учні повинні з'ясувати, які величини характеризують дану ситуацію, які зв'язки між цими величинами, як подати встановлені зв'язки за допомогою електронних таблиць, що виконання обчислень нерідко є складним для учнів, оскільки вимагає застосування знань з математики у дещо зміненому вигляді (потрібно не лише пригадати відповідну формулу, а й «перекласти» її у позначення табличного процесора, з використанням адрес комірок).

При вивченні обчислень в середовищі табличного процесора вправи повинні поступово ускладнюватися:

- спочатку це повинні бути вправи на виконання елементарних обчислень (без використання вбудованих функцій);
- потім поступово ввести функції без використання абсолютних адрес;
- далі на прикладах показати необхідність введення абсолютних посилань;

- запропонувати завдання на обчислення з використанням функцій, копіювання яких вимагає звернення до аргументів з абсолютними координатами.

Серед великої кількості вбудованих функцій табличного процесору Н.В.Морзе рекомендує звернути увагу на функції СЕРЗНАЧ (СРЗНАЧ), СУМ(СУММ), МАКС, НАЙБІЛЬШИЙ (НАИБОЛЬШИЙ), МІН (МИН), НАЙМЕНШИЙ (НАИМЕНЬШИЙ), РАНГ, можна також розглянути функції для роботи з текстом. Останніми із списку вбудованих функцій доцільно розглядати логічні функції (їх вивчення не є обов'язковим для класів усіх профілів), які сприймаються дещо гірше внаслідок «нетрадиційного» типу результату. Особливо складною є функція ЯКЩО (ЕСЛИ), учні часто не розуміють схему роботи за такою функцією (яке саме значення вноситься і в яку комірку). Тому особливості логічних функцій необхідно з'ясовувати на достатній кількості прикладів і завдань.

Матеріал, що стосується вивчення абсолютних і відносних посилань, часто викликає утруднення. Оскільки відмінність у способах адресації помітна, наприклад, при перенесенні формул шляхом копіювання, то необхідно розглянути приклади зміни (або незміни) адреси при копіювання принаймні в сусідні комірки по горизонталі та вертикалі. Корисно також навчити учнів перевіряти правильність вибору абсолютних чи відносних посилань копіюванням у сусідні комірки. При цьому слід наголосити, що при виявленні помилки змінювати потрібно не скопійовану, а вихідну формулу.

Під час навчання обчислень у середовищі табличного процесору доцільно виконувати деякі завдання (наприклад, визначення результату виконання функції, визначення формули після її копіювання) усно або письмово без їх виконання на комп'ютері, що сприяє більш глибокому розумінню процесу обчислення.

Цікавим для учнів є вивчення засобів ділової графіки. Цей матеріал нескладний для засвоєння, але при його вивченні не слід обмежуватися формальними питання виконання алгоритму побудови діаграм та їх форматування. Необхідно також навчати учнів аналізувати одержані результати, добирати найбільш відповідний задачі тип діаграми.

значення. Можна також зазначити, що для змінних, описаних у розділі опису (глобальних змінних) на початку виконання програми виділяється відповідна пам'ять («підписуються скриньки» відповідного розміру), але вміст цієї пам'яті не визначений, тобто у відповідних комірках пам'яті спочатку будуть знаходитися значення, які залишилися після використання цієї пам'яті іншою програмою. Не зважаючи на те, що останні версії середовищ для розробки програм мовою Паскаль переважно обнуляють значення змінних при відведенні пам'яті, але відбувається це не в усіх випадках.

Крім того, при вивченні змінних, слід звернути на правильність записів ідентифікаторів мовою Паскаль. При цьому слід мати на увазі, що нерідко досвід позначення величин у математиці, фізиці тощо веде до того, що для змінних числового типу учні використовуються позначення з однієї літери, а для літерного можливо і буквосполучення. Для формування навичок використання ідентифікаторів доцільно використовувати вправи підведення під поняття (вибрати серед запропонованих послідовностей символів допустимі ідентифікатори).

Слід також звернути увагу на правила синтаксису: правила опису констант та змінних, запис коментарів, правила розділення ідентифікаторів, структуру програми, забезпечити засвоєння набору допустимих операцій та функцій. Стосовно останнього слід звернути увагу на вміння визначати тип результату дій. важливих умінь при роботі з середовищем програмування є вміння налагоджувати програми, правильно інтерпретувати повідомлення транслятора, здійснювати пошук помилок. При колективному розв'язуванні задач на складання програм такі вміння формують недостатньою мірою, що в майбутньому ускладнює самостійну роботу учнів у середовищі програмування. З метою формування вмінь аналізу синтаксичної правильності алгоритмів доцільно практикувати завдання на пошук помилок у програмах та їх фрагментах.

Завдання на пошук помилок у програмі, як і визначення результату виконання програми, рекомендується пропонувати учням систематично як спільне завдання для класу на початку уроку (аналогічно «усному рахунку» у початкових класах). Виконання завдання дозволить учителю:

- з перших хвилин активізувати увагу учнів на уроці;



важливо, щоб учні зрозуміли принцип побудови і читання схеми, немає потреби демонструвати надто велику кількість різних блоків, у шкільному курсі інформатики переважно обмежуються блоками для зображення початку/кінця алгоритму, введення/виведення даних, виконання дій, перевірки умов.

Жоден алгоритм не можна вважати правильним і процес написання його закінченим, якщо він ретельно не перевірений шляхом виконання.

Виконання алгоритмів і програм є ефективною формою контролю знань учнів. Виконання алгоритмів можна також зробити засобом активізації діяльності учнів на уроці, якщо залучити до виконання кількох учнів (наприклад, у командах розглянути значення виконання алгоритмів полягає у розвитку акуратності, уважності, уміння доводити до кінця розпочату справу. Розвивається логічне і раціональне мислення.

Виконання алгоритму (програми) найбільш легко і наочно можна здійснити за допомогою таблиці значень (таблиця виконавця). У таких таблицях відображаються значення змінних і результати перевірок умов на кожному кроці виконання програми. Таблиці значень зустрічаються уже в підручнику за редакцією А.П.Єршова, В.М.Монахова, в іншій методичній літературі автори пропонують свої варіанти оформлення таблиць значень. Особливості вивчення деяких інших тем розділу розглянемо на прикладі мови програмування Паскаль.

### **Основні елементи мови**

На початку вивчення мови програмування повинні бути сформовані уявлення про загальні принципи побудови програм, що є запорукою подальшого навчання. Одним із важливих понять є поняття змінної. У програмуванні під змінною розуміють дещо, напряму не пов'язане із поняттям змінної у математиці. Скоріше, поняття змінної у програмуванні ближче до поняття параметра в математиці (наприклад, у рівняннях з параметрами). А головне, поняття змінної напряму пов'язане із комп'ютерною пам'яттю. Пояснення можна доповнити інтерпретацією поняття величини за А.П.Єршовим, проводячи аналогію зі скриньками, на які наклеєно унікальні етикетки (імена) і у яких щось зберігається (значення). Така аналогія також корисна при поясненні семантики операції надання

## **Особливості вивчення баз даних та СУБД**

Теорія баз даних — важливий розділ сучасної інформатики. Їх широке використання в різних галузях людської діяльності робить актуальною підготовку користувачів баз даних. Мета вивчення баз даних як сукупності засобів для зберігання структурованої інформації полягає в узагальненні та систематизації уявлень про можливості опрацювання великих масивів даних, формуванні відповідних теоретичних знань, з'ясуванні загальних принципів опрацювання структурованої інформації та оволодінні навичками опрацювання баз даних за допомогою конкретної СУБД.

*Основні типи навчальних задач* (відповідно до основних типів навчальної діяльності користувача з базами даних):

- 1) одержання інформації на основі даних з базами даних;
- 2) створення нової бази даних, підтримка одержаної моделі предметної галузі в певному стані;
- 3) оновлення бази даних.

Предметна галузь бази даних повинна бути відома і зрозуміла учням. У той же час використання різних предметних галузей дозволяє здійснювати міжпредметні зв'язки.

Крім традиційних, можна давати завдання подання нових даних у табличній формі з елементами статистичного опрацювання (такий підхід дозволяє здійснювати міжпредметні зв'язки та продемонструвати переваги використання комп'ютерних середовищ СУБД можна поділити на 2 частини:

- 1) робота з готовою базою даних для розуміння основних понять, властивостей об'єктів та операцій над ними;
- 2) вивчення можливостей використання конкретних СУБД та створення власних баз даних.

Після вивчення теми можна запропонувати творчу або проектну роботу на проектування бази даних, що є досить складним і творчим завданням.

На початку вивчення теми учні повинні з'ясувати та засвоїти теоретичні питання, пов'язані з базами даних. Поняття баз даних відоме учням інтуїтивно. Вони можуть пояснити зміст слова «база» як місце зберігання певних речей. Після ознайомлення з поняттям бази даних можна на конкретних прикладах роз'яснити, що при великому обсязі інформації її

доцільно структурувати за певними правилами та опрацювати за допомогою комп'ютера.

До основних теоретичних питань з баз даних слід також віднести знання типів баз даних (фактографічні та документальні) та моделей подання даних (ієрархічні, реляційні, мережеві). Для розуміння типів зв'язків між елементами можна запропонувати учням також вправи на визначення типу бази даних, поданої за допомогою конкретної структури.

У шкільному курсі інформатики розглядаються лише фактографічні реляційні бази даних. Це пов'язано не тільки з обмеженістю шкільного курсу, але і з тим фактом, що реляційні бази даних використовуються сьогодні найчастіше і є універсальними (будь-яка система даних може бути подана у вигляді таблиць). Найпростіша реляційна база даних містить одну таблицю, складніші містять кілька взаємопов'язаних таблиць. При ознайомленні зі структурою реляційних баз даних та відображення за їх допомогою предметною галузі, доцільно звернути увагу на такі моменти:

- будь-яка таблиця містить у собі інформацію про деяку реальну систему (процес) і, отже, є її інформаційною моделлю;
- запис у таблиці — інформація про конкретний об'єкт даної системи;
- значення поля кожного запису — визначена характеристика об'єкта.

Розуміння цих питань полегшить у майбутньому проектування баз даних і забезпечуватиме попередження помилок у процесі створення таблиць (нерозрізнення таблиці та її структури). Після розгляду основних питань теорії баз даних доцільно перейти до ознайомлення учнів із системою управління базами даних. Для демонстрації можливостей СУБД спочатку слід розглянути приклад конкретної бази даних, яка містить усі основні об'єкти, з якими працює СУБД. Для такого ознайомлення при вивченні СУБД MS Access можна використати базу даних *Борей*, яка входить до її стандартного набору. Перевагою вказаної бази даних є велика кількість записів і використання різних типів полів. Однак предметна галузь не є практично значущою для учнів, тому не варто у вивченні обмежуватися тільки цією базою даних.

- у посібнику Т.П.Караванової для класів з поглибленим вивченням інформатики список властивостей містить дискретність, скінченність, визначеність, зрозумілість, масовість, результативність, ефективність;
- у підручнику А.Ф.Верляня, Н.В.Апатової замість ефективності наведено формальність.

Неоднозначність у визначенні властивостей алгоритмів пояснюється не тільки тим, що теорія алгоритмів є досить молодою наукою, але й тим, що для потреб практики буває досить обтяжливо вимагати від алгоритмів обов'язкової наявності всіх зазначених властивостей. Саме поняття алгоритму також розвивається, звільняючись від тих чи інших обмежень. На сьогодні, наприклад, розглядається не тільки послідовне, але й паралельне виконання кроків алгоритму (у той же час властивість дискретності передбачає, що алгоритм складається з опису послідовності тактів обробки, побудованої таким чином, що в початковий момент задається вихідна ситуація, а в кожному наступний момент ситуація перетворюється на основі даних, одержаних у попередні такти обробки). Розглядаються також недетерміновані алгоритми (властивість детермінованості означає, що на кожному кроці однозначно визначено перетворення об'єктів середовища виконавця, одержаних на попередніх кроках алгоритму; цю властивість називають також визначеністю або однозначністю), у яких на черговому кроці може здійснюватися випадковий вибір з деякої множини варіантів.

Практично в усіх переліках властивостей присутня масовість алгоритму, яка означає, що є деяка множина початкових станів, які можуть опрацюватися даним алгоритмом. З точки зору практичної цінності алгоритмів важливо, щоб множина допустимих початкових ситуацій була досить великою. Проте навіть у навчальній діяльності школярів зустрічаються алгоритми, які не мають властивості масовості. Властивість формальності не має значної теоретичної цінності (адже всі алгоритмічні системи є формальними по своїй суті), проте важлива з методичної точки зору, оскільки підкреслює, що виконання послідовності операцій здійснюється «механічно», розглядаючи різні форми подання алгоритмів, слід ознайомити із основними правилами подання алгоритмів у вигляді блок-схем (граф-схем, структурних схем). При цьому

задається послідовність дій, допустимих для деякого виконавця, що забезпечує досягнення поставленої мети.

Слід відзначити наявність посилання у цьому описі поняття виконавця. Зрозуміло, що неможливо обговорювати виконання алгоритму без вказівки, для кого він призначений, а також середовища виконання.

Слід відзначити, роль виконавця алгоритму є дуже важливою в розділі «Алгоритмізація». Виконавці використовуються як методичний засіб, що дозволяє ввести основні поняття алгоритмізації і розпочати розв'язувати задачі в наочному середовищі, вільному від надмірних математичних чи інших складностей. Важливо зрозуміти, що виконавець характеризується середовищем, системою вказівок, елементарною дією, відмовами. При введенні поняття виконавця алгоритмів слід звернути увагу на такі питання:

- виконавцем алгоритмів може бути не тільки людина;
- будь-який виконавець складається з пристрою керування і «робочого інструменту»;
- кожен виконавець алгоритмів має обмежений набір допустимих дій;
- для розв'язування одних і тих самих задач виконавці з «біднішим» набором допустимих дій вимагають складніших і детальніших алгоритмів;
- різні класи задач вимагають різних наборів допустимих дій, різних виконавців.

Під наведений вище опис поняття «алгоритм» підходять найрізноманітніші інструкції (наприклад, з кулінарної книги), керівні вказівки тощо. Риси алгоритмів, які визнаються такими, що вирізняють їх з множини все можливих інструкцій, приписів, називають властивостями алгоритму. У різних посібниках виділяють різні набори властивостей. Наприклад:

- у підручнику авторського колективу А.М. Гуржій, І.Т. Зарецька, Б.Г. Колодяжний, О.Ю. Соколов названо такі властивості: масовість, скінченність, однозначність, правильність, ефективність;
- у підручнику для 7 класу (автори І.Л. Володіна, В.В. Володін, Ю.О. Дорошенко, Ю.О. Столяров) властивостями алгоритму є однозначність, ефективність, скінченність, результативність масовість;

При демонстрації можливостей СУБД MS Access доцільно звернути увагу на такі питання:

1) Можливість виконання різними способами основних операцій:

- пошук записів за умовою;
- поновлення в базі даних значень деяких полів;
- створення звіту;
- додавання, вилучення записів.

2) Можливість працювати з різними предметними галузями, які найчастіше використовують переваги

3) Можливість опрацювання даних різних типів.

Об'єкти Access рекомендується вивчати в такому порядку: файли — таблиці — запити — форми — звіти. Макроси та модулі можна вивчати при поглибленому вивченні інформатики.

Під час ознайомлення з кожним з об'єктів доцільно дотримуватися наведеної нижче методики.

1. За допомогою невеликих конкретних завдань та запитань на прикладі створеної бази даних продемонструвати основні засоби для роботи з об'єктом, пояснити вказівки.

2. Навчити працювати з кожним з об'єктів в режимі роботи з об'єктом та в режимі конструктора. Сформулювати уявлення про те, що об'єкт змінюється тільки в режимі конструктора, працювати слід в режимі об'єкта.

3. Продемонструвати різні способи створення об'єкта.

4. Використовувати індуктивний метод на частково-пошуковій основі та метод демонстраційних прикладів з підказками різного типу (від заповнених діалогових вікон до карток-підказок, точних і узагальнених алгоритмів виконання відповідних дій).

При організації практичних робіт слід звернути увагу на те, що збереження записів у базі даних відбувається автоматично, тому при необхідності використовувати готові файли для опрацювання необхідно або копіювати файл бази даних в іншу папку, або використати імпорт даних.

При вивченні таблиць слід звернути увагу учнів на режим конструктора, підкресливши, що цей режим не використовується для введення даних, він призначений для визначення структури таблиці — набору поіменованих полів, у яких описують властивості об'єктів. Доцільно звернути увагу учнів на те, що

першим кроком при створенні бази даних є саме визначення її структури у вигляді набору полів з певними характеристиками. При цьому наступна робота з базою даних залежить від коректності структури, оскільки робота проводиться з полями і записами, а не окремими комірками (як у табличному процесорі).

Працюючи в режимі таблиці, учні повинні навчитися: вносити з клавіатури дані до таблиці; редагувати дані в таблиці; упорядковувати дані за різними ознаками; здійснювати простий пошук даних; користуватися фільтрами для добору даних.

Запити є найважливішим і найскладнішим матеріалом для учнів. Однією з причин є недостатня увага до основ математичної логіки. Запити СУБД MS Access можна поділити на дві групи: запити, які не змінюють даних бази даних (запити на вибірку та перехресні запити), та запити, які вносять зміни до бази даних (запити на видалення, оновлення, додавання, створення таблиці, SQL-запити). Запити на вибірку мають спільні риси із фільтрами, тому для полегшення сприймання доцільно провести спочатку їх порівняння, початкові завдання давати аналогічні тим, які розглядалися при роботі з фільтрами. Запити на вибірку часто рекомендують вивчати у режимі конструктора, щоб учні краще зрозуміли принцип дії запиту. Складнішим для розуміння є використання операцій групування. Побудова запитів із параметрами, використання обчислювальних полів є матеріалом, досить складним для учнів. При вивченні перехресних запитів слід звернути увагу учнів на те, що його використання доцільне тоді, коли значення деякого поля (полів) повторюються, а також на правильність визначення того, значення яких полів використовуватимуться в обчисленнях, а які — як заголовки рядків та стовпців, оскільки типовою помилкою є винесення як заголовків рядків усіх трьох полів (і заголовків рядків, і заголовків стовпців, і поля, що використовується в обчисленнях). Сприйняття матеріалу полегшується, якщо при вивченні табличного процесора розглядалися питання побудови зведених таблиць. Створювати запит можна за допомогою майстра, а потім в режимі конструктора. Вивчення простіших запитів є досить складним, як вивчення запитів, іноді навіть вважається не обов'язковим для вивчення. Однак цей матеріал є практично значимим, а також дозволяє

Службові слова пс і кє є скороченням відповідно словосполучень «початок серії» і «кінець серії» і відіграють роль операторних дужок у мові програмування Паскаль.

У переважній більшості версій службові слова НАМ є скороченнями відповідних російських (українських) службових слів: алг (алгоритм), нач / поч (начало/початок), кон / кін (конець/кінець) тощо. У системі **ALGO**, яка є однією з перших українських (і україномовних) оболонок, службові слова є фактично українськими перекладами відповідних службових слів Паскаля. Зберігається також структура програми (включаючи опис об'єктів), проте вимоги синтаксису є менш жорсткими, ніж у Паскалі. Такий підхід полегшить у майбутньому перехід від НАМ до зазначеної мови програмування, тим більше, що система дозволяє здійснювати транслітерацію з Паскаля до використання навчальної алгоритмічної мови, що ґрунтується на використанні простих виконавців-роботів. Кожен з них призначений для конкретної, у тому числі навчальної задачі. За такого підходу саме з керування виконавцями-роботами починається вивчення програмування. Для цього розроблені відповідні мови та системи програмування (ЛОГО, виконавці Перевозчик, Плюсик, Чертежник, Кенгуренок, Кукарача). Тільки дізнавшись у грі з виконавцями про головні поняття, необхідні для програмування (команда, система команд, ім'я інформаційного об'єкту, розгалуження, повторення), можна розпочинати вивчення мов програмування, спочатку навчальних, а потім професійних.

## **Особливості вивчення окремих тем**

### **Алгоритм**

Основне завдання теми — формування в учнів поняття алгоритму. Це поняття, які і поняття інформація і комп'ютер, відноситься до найважливіших у шкільному курсі інформатики. Перш за все необхідно констатувати, що поняття алгоритму, як і поняття інформації, є неозначуваним, належить до числа основних початкових понять математики й інформатики, які не допускають визначення в термінах простіших понять. Зміст цього поняття полягає в тому, що за допомогою алгоритму

перевантажує пам'ять другорядними деталями конструкцій.

4. Є відкритою системою, що може розвиватися та змінюватися.
5. Точність і однозначність описів дозволяє перекладати алгоритми з НАМ на мову програмування.
6. Зручність для виконання людиною.

Проте існує і інша точка зору на викладання в шкільному курсі навчальної алгоритмічної мови. Такий підхід продуктивно використовувався у 80-90-х роках минулого століття за умов неналежного забезпечення шкіл комп'ютерною технікою, підручниками тощо. Сьогодні ситуація змінилася на краще. Для освоєння програмування достатньо вивчити десяток англійських слів і жодних проблем не буде, якщо методика навчання побудована правильно за принципом «навчання з метою розв'язування цікавих задач, а не заради вивчення мовних конструкцій», якщо є достатня мотивація навчання і кожний учень має нормальну книжку.

До того ж, певні методичні складнощі для вчителя у викладанні навчальної алгоритмічної мови становить відсутність єдиного стандарту останньої. Хоча її принципи є однаковими, різні методисти, автори підручників пропонують свої варіанти, які відрізняються деякими службовими словами, порядком їх використання в операторах. Наприклад, у підручнику авторського колективу А.Г.Кушніренко, Г.В.Лебедев, Р.А.Сворень оператор циклу з параметром записується так:

нц для і от і1 до і2  
тіло циклу

кц

Службові слова нц і кц є скороченням відповідно словосполучень «начало цикла» і «конец цикла».

Цикл з параметром у посібнику В.В. Черняхівського має дещо інший вигляд:

для і від і1 до і2  
пс

тіло циклу

кс

підвищити пізнавальну активність учнів. Доцільно ознайомити не тільки зі створенням цих об'єктів за допомогою відповідних майстрів, але і познайомити з особливостями режиму конструктора. Можна також ознайомити зі створенням обчислювального елемента в формі.

Наприкінці вивчення теми, узагальнюючи одержані відомості роботи з базами даних, доцільно розглянути процес проектування баз даних. Цей процес є складним, вимагає визначення на основі опису деякої предметної галузі складу і структури даних, які повинні знаходитися в базі даних і забезпечувати обслуговування запитів і задач користувача.

## Самостійна робота студентів

### Матеріал для самостійного опрацювання

#### Навчання пошуку даних у табличному процесорі та СУБД

При роботі в середовищі табличного процесора, крім формування вмінь здійснювати пошук відповідних файлів за різними групами ознак (аналогічно до пошуку файлів у текстовому редакторі), передбачено навчання учнів використання засобів для відшукування інформації в базах даних, робота з якими підтримується в редакторі електронних таблиць. Вивчення найперших понять, пов'язаних з опрацюванням баз даних, та демонстрація деяких вбудованих засобів роботи з базами даних – форми, фільтри, розширені фільтри, підсумки;

2) формування вмінь учнів стосовно впорядкування та пошуку даних в базі даних.

У середовищі табличного процесора MS Excel добір з бази даних інформації, яка задовольняє певні критерії, можна здійснити, зокрема, за допомогою фільтрування даних. Учням необхідно показати, що відбирати із набору можна дані, які задовольняють певні умови, наприклад: певну кількість даних; дані, що збігаються з конкретним значенням; дані, значення яких більше чи менше заданого. При цьому під час формування критеріїв пошуку можна використовувати як одне поле, так і кілька; ключ для порівняння можна вибирати зі списку або вводити вручну. Засобів фільтрування доцільно вивчати автофільтр. Спочатку доцільно пояснити, як працює фільтр за

виділеним елементом, продемонструвати можливість послідовного застосування такого фільтра до різних стовпців. Далі розглядають особливості використання умов та підстановочних знаків, а також відбір певної кількості записів із заданою властивістю (5 найбільших елементів, 10 найменших тощо). На основі розглянутих прикладів та самостійної роботи учнів доцільно сформулювати узагальнююче правило використання автофільтра.

Після формування навичок використання автофільтра можна ознайомити учнів із використанням розширеного фільтра. Доцільно разом з учнями проаналізувати приклади використання розширеного фільтра, врахувавши, що при цьому в умови добору може входити кілька умов, що накладаються одночасно на одне поле, кілька умов, що накладаються одночасно на кілька полів, а також умови, що накладаються на значення, яке обчислюється за формулою — обчислювальні критерії. Останній випадок є найбільш складним для учнів, оскільки вимагає навичок використання формул, у тому числі й з абсолютними адресами. Фактично свідчить, що пошук даних за допомогою фільтрів є складним для засвоєння матеріалом, особливо це стосується використання логічних операцій під час побудови простих і складених умов пошуку даних, а також використання розширеного фільтра. Для кращого розуміння правил використання автофільтра (особливо з простими та складеними умовами) та розширеного фільтра можна практикувати розв'язування задач без використання комп'ютера: визначити результати виконання фільтрації записів за наведеними критеріями.

Матеріал стосовно використання різних видів фільтрів у табличному процесорі є пропедевтичним для застосування фільтрів і побудови запитів у СУБД.

Під час вивчення добору даних за допомогою фільтрів у середовищі СУБД MS Access слід звернути увагу, що при роботі з таблицями існує чотири види фільтрів: фільтр за виділеним фрагментом, звичайний фільтр, поле *Фільтр* для розширеного фільтра. Усвідомленню матеріалу сприяє порівняння різних видів фільтрів, визначення їх спільних і відмінних рис.

Ознайомлення із фільтром за виділеним доцільно провести індуктивним шляхом, проаналізувавши результати його застосування в різних умовах.

Для визначення особливостей звичайного фільтра спочатку слід пояснити учням, що являє собою такий фільтр: спеціальна таблиця, яка має структуру вихідної, але не містить жодного запису. У цій таблиці користувач вносить до відповідних полів

Досвід вивчення систем візуального програмування показує також ефективність такого підходу в ознайомленні учнів з основами алгоритмізації і програмування.

Проте існує точка зору, що візуальне програмування доцільно вивчати тільки після освоєння алгоритмічних конструкцій і розв'язування типових навчальних задач обчислювального характеру і задач пошуку інформації. Якщо ж алгоритмічні конструкції вивчати в рамках візуального програмування (а саме це пропонується в деяких програмах), то може статися як у прислів'ї «За лісом дерев не видно». Головна причина розмежування полягає в тому, що ці два підходи у програмуванні призначені для розв'язування принципово різних задач, тому сумішати їх хоч і можна (і це де-не-де обґрунтовується в зарубіжній методичній літературі), але недоцільно і неефективно.

### **Навчальна алгоритмічна мова**

Алгоритмічна мова, орієнтована на людину, є проміжною ланкою між мовою програмування та засобами описування алгоритмів.

*Функції навчальної алгоритмічної мови у навчанні основи алгоритмізації* (за А.Єршовим):

- 1) стандартизація описів алгоритмів шкільного курсу інформатики, що важливо для розуміння суті алгоритмізації, формування уявлень про властивості алгоритму;
- 2) пропедевтика мов програмування.

### **Переваги навчальної алгоритмічної мови**

1. Узгоджується із принципами структурного програмування, які вважаються «універсальною методологією програмування».
2. Відносна близькість до природної мови, що забезпечується національною лексикою. Зрозумілість службових слів учням дозволяє зосередити увагу на логічній структурі та сутності програми, а не зауважувати на обмеженість вказівок (присвоєння, розгалуження, повторення, виклик допоміжного алгоритму), що не

інтерес до вивчення учнів, які орієнтовані на таку професійну діяльність.

## **Методика вивчення систем візуального програмування**

Об'єктно-орієнтований підхід передбачає нове розуміння процесів обчислень та структурування даних.

Система візуального програмування базується на ідеї подійно-орієнтованого програмування: програма — це сукупність об'єктів реального або віртуального світу, з кожним з яких пов'язаний деякий обмежений набір дій.

Використання середовища візуального програмування дозволяє звести воедино «старий», математико-алгоритмічний, і новий, інформаційно-технологічний підходи до вивчення інформатики, які раніше були майже незалежними.

### **Основні принципи середовищ візуального програмування**

1. Відокремлення елементів (об'єктів) програми, пов'язаних з інтерфейсом, від алгоритмічної частини;
2. Швидкість і простота створення, модернізації інтерфейсу, у якому використовуються готові блоки, що реалізують деякі великі функції управління програмою;
3. Використання кодів, описаних іншими мовами програмування.

### **Переваги використання систем візуального програмування**

1. Можливість створювати прості програми, взагалі не знаючи, що таке програмування.
2. Можливість відразу спостерігати наслідки роботи.
3. Можливість відразу працювати з інтерфейсом, тобто відразу після ознайомлення з новим матеріалом можна переходити до створення простих програм.
4. Можливість організації роботи над проектами, свобода вибору методів розв'язування задачі, реалізація міжпредметних зв'язків.
5. Можливість розробки програм, що відповідають принципам функціонування сучасного програмного забезпечення в середовищі операційної системи з графічним інтерфейсом.

значення, які повинні міститися у шуканих записах. Спочатку доцільно сформулювати в учнів уміння за підготовленими умовами добору формулювати запити на добір записів, а уже потім пропонувати завдання для самостійного виконання на відбір необхідних запитів. Як підсумок доцільно сформулювати правило та випадки доцільного використання звичайного фільтра.

Найбільш складним для розуміння учнями є розширений фільтр, вивчення якого доцільно будувати у два етапи залежно від складності умов відбору, що використовуються. Слід відзначити, що робота із розширеним фільтром найбільш близька до роботи із запитом у режимі конструктора.

## **Особливості вивчення комп'ютерних презентацій**

Тема «Комп'ютерні презентації» внесена до шкільного курсу інформатики відносно недавно і не завжди висвітлюється у шкільних підручниках. Основна мета вивчення теми — ознайомити учнів з основними можливостями систем підготовки комп'ютерних презентацій, сформулювати вміння створення і демонстрації презентацій з використанням різних типів об'єктів (текстових, графічних, числових, звукових, відео), анімаційних ефектів та реалізацією переходів між слайдами за допомогою гіперпосилань і системи навігації. Детальніший розгляд питань створення комп'ютерних презентацій можливий у курсі за вибором «Основи комп'ютерних презентацій».

Слід відзначити, що вивчення комп'ютерних презентацій викликає в учнів інтерес, позитивні емоції, дає можливість реалізувати творчі здібності. Особливо приваблює можливість надання динамічності презентації за допомогою анімації та системи навігації. З іншого боку, тема має значний загальноосвітній, світоглядний, розвивальний потенціал: розвиток навичок подання певного матеріалу, своєї позиції, розвиток вмінь виділяти головне, грамотно використовувати наочність. Спочатку вивчення теми доцільно продемонструвати учням переваги використання комп'ютерних презентацій при поданні матеріалу, можливість розміщення як тексту, так і графічних об'єктів, звукового супроводу, «оживлення» презентацій за допомогою анімації, використання таких засобів керування переглядом презентації, як кнопки і гіперпосилання.

Тема «Комп'ютерні презентації» переважно вивчається після теми «Текстовий процесор», тому при достатньому рівні знань учнів щодо елементів інтерфейсу (особливо, якщо

використовуються програми єдиного інтерфейсу користувача) ознайомлення з інтерфейсом системи підготовки комп'ютерних презентацій можна проводити як повторення. При цьому необхідно звернути увагу на відмінності робочої області середовища системи підготовки комп'ютерних презентацій і середовища текстового процесора.

Після вивчення інтерфейсу доцільно ознайомити учнів з режимами роботи середовища, звернути увагу на особливості режимів створення презентацій та їх перегляду.

Навчаючи учнів створювати презентації, слід звернути увагу на відмінності створення нової презентації і створення нового слайду в поточній презентації, оскільки останнє нерідко викликає утруднення в учнів (при роботі з текстовим процесором нові сторінки додаються автоматично при збільшенні обсягу тексту, при роботі з презентацією автоматичного створення слайдів не відбувається).

Можливості і правила форматування тексту, способи додавання об'єктів у середовищі систем підготовки презентацій має багато спільного з виконанням тих самих дій у середовищі текстового процесора. Тому можна запропонувати учням самостійно з'ясувати можливості форматування тексту, порівнявши їх із вивченими раніше, навчитися додавати до презентацій різні об'єкти. По відношенню до опрацювання тексту в середовищі текстового процесора є можливість застосування анімаційних ефектів до об'єктів різних типів. Вивчення цих питань приваблює учнів можливістю створити живу презентацію. Складнощі найчастіше виникають при налаштуванні ефектів анімації метою виховання естетичного смаку слід розглянути питання оформлення презентації. Необхідно звернути увагу учнів на те, що головним є зміст презентації, а форма подання повинна допомагати користувачеві зрозуміти її сутність висвітлюваного питання, а не відволікати від нього. Для цього варто дотримуватися таких правил:

- не слід розміщувати на презентації дуже великі тексти;
- не слід переобтяжувати презентацію графічними об'єктами;
- тексти слід набирати і формувати так, щоб він легко читався;
- не слід зловживати анімаційними ефектами і кольоровою палітрою;
- слід витримувати презентацію в єдиному стилі.

буде надано цілком певних значень, що залежать від вхідних даних, і ця залежність цілком визначається алгоритмом.

Оскільки не будь-яка вказівка може бути виконана, то іноді виникає необхідність подати дану вказівку у вигляді скінченного впорядкованого набору вказівок. Важливим є розуміння, що ступінь деталізації залежить від виконавця.

### ***Вибір мови програмування при вивченні основ алгоритмізації та програмування***

До важливих питань методики навчання основ алгоритмізації відноситься вибір мови програмування. Основні шляхи розв'язання цієї проблеми:

1. Одна чи кілька мов програмування високого рівня.
2. Машинно орієнтовані мови.
3. Мова схем, конкретні мови програмування та схеми.
4. Спеціально розроблена навчальна алгоритмічна мова.

При виборі конкретних мов програмування вчитель може зіткнутися з такими проблемами:

- 1) важко визначити перспективну мову програмування, стандартні мови орієнтовані на вузькі галузі застосування;
- 2) жодна із стандартних мов не відображає достатньо з точки зору методики сучасну концепцію програмування;
- 3) відмова від вивчення мов програмування веде до неможливості використовувати технічну базу.

Для формування основ інформаційної культури рекомендується поєднувати основні ідеї різних напрямків вирішення проблеми вибору мов. Один із підходів полягає в тому, що на початку вивчення концепцій програмування навчання організовується на основі навчальної алгоритмічної мови і паралельно граф-схем (які доповнюють одна одну), після вивчення навчальної алгоритмічної мови вивчається мова програмування. Основна мета при такому переході — показати ідею ручного перекладу (трансляції) алгоритмів, записаних навчальною алгоритмічною мовою, на одну з мов програмування. При цьому використання мови і середовища програмування, які реально використовуються у сучасній практиці, у тому числі і професіоналами, дозволяють підвищити



спочатку видаються готові програми, повідомляється, що саме вони роблять, і пропонується написати схожу програму або змінити ту, що є, не пояснюючи до кінця ряд “технічних” або неістотних, з погляду вчителя, для розв’язування задачі деталей. При цьому говориться, що про точне значення відповідних конструкцій учні дізнаються пізніше, а поки що слід діяти аналогічно. Другий підхід дає можливість так званого “швидкого старту”, але створює небезпеку одержати напівграмотних користувачів середовища програмування, тобто людей, які використовують у своїй практиці достатньо складні конструкції, але не можуть чітко пояснити, чому в тому або іншому випадку потрібно застосовувати саме їх, і як вони працюють. У результаті рано чи пізно такі “програмісти” стикаються з помилками, виправити які вони просто не в змозі — їм не вистачає знань. Але і без хороших прикладів (зразків) при навчанні учнів програмування не обійтися. І чим молодше учні, тим більше прикладів необхідно наводити при описі мови (іноді навіть замінюючи ними строге визначення). Проте слід досягати того, щоб у результаті обговорення прикладу всі його деталі виявилися зрозумілими школярам (обов’язково потрібно пояснити, як і чому працює програма в цілому і кожна вказівка окремо, зокрема спираючись на вже вивчений формальний матеріал). У цьому випадку сильні учні дістануть можливість зрозуміти все досконально і зможуть використовувати одержані знання надалі, а середні — набудуть конкретних навичок і залишать для себе можливість повернутися при необхідності до формальних визначень пізніше.

Як теоретичну і методологічну основи, принцип побудови алгоритмів доцільно обрати ідеї і методи **структурного програмування**. Їх використання формує навички чіткого дотримання дисципліни праці при конструюванні алгоритмів, що сприяє розвиткові логічного мислення учнів уже на ранніх етапах вивчення основ алгоритмізації.

Важливо показати, що вказівку про виконання алгоритму отримання розв’язку, який за алгоритмом однозначно ставиться у відповідність аргументам, можна розглядати як окрему вказівку, за якою змінним, які представляють шукані результати,

## **Завдання для самостійної роботи**

1. Дослідіть та порівняйте вказівки пошуку даних і програм у різних середовищах, які вивчаються у шкільному курсі інформатики.

2. Використання задач яскраво вираженої життєвої спрямованості дозволяє, з одного боку, створювати мотивацію для розв’язування даних задач, а з іншого, здійснювати внутріпредметні зв’язки з темою «Інформаційна модель», адже розв’язування таких задач вимагає визначення того, яка інформація є істотною для розв’язування задачі і як ця інформація може бути описана за допомогою тих чи інших параметрів і зв’язків між ними. Запропонуйте прикладні задачі життєвої спрямованості, які можуть бути розв’язані за допомогою прикладних програмних засобів загального призначення. Для класів яких профілів будуть цікаві такі задачі?

## **Запитання для самоконтролю**

1. Яка мета вивчення прикладного програмного забезпечення загального призначення в школі? Сформулюйте вимоги до знань і умінь учнів при вивченні ІКТ.
2. Охарактеризуйте методичні особливості вивчення прикладного програмного забезпечення загального призначення у шкільному курсі інформатики.
3. Опишіть схему ознайомлення з прикладним програмним забезпеченням загального призначення.
4. Обґрунтуйте необхідність поєднання конкретики вивчення програмних продуктів із загальними принципами їх функціонування. Які вправи сприяють формуванню узагальнених умінь і навичок?
5. Охарактеризуйте особливості організації практичних робіт теми.
6. Вивчення яких теоретичних питань передбачено в темі «Графічний редактор»?
7. Охарактеризуйте особливості практичних завдань з тем «Графічний редактор», «Текстовий редактор».
8. Які питання мають світоглядне значення при вивченні текстового процесора?
9. На які питання слід звернути увагу при вивченні теми «Текстовий процесор», враховуючи місце теми в програмі?
10. Коротко охарактеризуйте методичні особливості вивчення можливостей текстового процесора.

11. У чому відмінності понять «електронна таблиця» і «табличний процесор»?
12. Які особливості ознайомлення учнів з інтерфейсом табличного процесора, можливостями форматування комірок?
13. Охарактеризуйте особливості вивчення обчислень в табличному процесорі.
14. Які особливості вивчення засобів ділової графіки, пошуку даних за допомогою фільтрів у табличному процесорі?
15. Які теоретичні питання теорії баз даних розглядаються в шкільному курсі інформатики?
16. На які питання слід звернути увагу при демонстрації можливостей СУБД MS Access?
17. Опишіть загальні методичні особливості ознайомлення з об'єктами СУБД MS Access.
18. Які особливості ознайомлення учнів з кожним з об'єктів СУБД MS Access?
19. Охарактеризуйте особливості вивчення різних типів фільтрів у СУБД MS Access.
20. Порівняйте особливості ознайомлення з фільтрацією даних у середовищах табличного процесора і системи управління базами даних?
21. Охарактеризуйте особливості вивчення комп'ютерних презентацій у шкільному курсі інформатики.

- 2) зафіксувати його за допомогою деякої системи позначень так званою формальною мовою.

При розробці алгоритму необхідно дотримуватися таких правил:

- 1) встановити чітку крок за кроком послідовність дій, зрозумілих людині;
- 2) визначити характер вихідних даних — прості чи структуровані, числові чи текстові тощо;
- 3) намагатися використовувати не конкретні числа, а позначення змінних;
- 4) вказати місце введення вхідних даних в комп'ютер та місце виведення результатів;
- 5) вказати всі формули розв'язування задачі й умови, за яких вони виконуються.

Вважається доцільним суміщення навчання основ алгоритмізації та навчання основ програмування, не відмовляючись від опису алгоритму рідною мовою.

Серед вправ розділу «Основи алгоритмізації та програмування» можна виділити такі основні класи:

- 1) **на тестування**: за текстом задачі та алгоритмом виконати алгоритм і з'ясувати особливості роботи за алгоритмом;
- 2) **на «відгадування» формулювання задачі**: за описом алгоритму виконати його і встановити призначення;
- 3) **на складання алгоритму**: за призначенням алгоритму написати текст алгоритму — скласти алгоритм і виконати його.

### *Підходи до вивчення програмування, використання ідей та методів структурного програмування*

При вивченні програмування можна виділити два підходи: формальний і «програмування за зразком». Перший заснований на формальному (строгому) описі конструкцій мови програмування (синтаксису мови і його семантики) тим або іншим способом (за допомогою синтаксичних діаграм, метамови або формального словесного опису, зокрема, семантики) і використанні при розв'язуванні задач тільки вивчених, а отже, зрозумілих елементів мови. При другому ж підході школярам

найпростіших алгоритмів, описувати найпростіші алгоритми навчальною алгоритмічною мовою та формально їх виконувати. Обсяг матеріалу для класів філологічного, суспільно-гуманітарного, художньо-естетичного, спортивного профілів значно менший: розглядаються виключно світоглядні питання поняття алгоритму, його властивостей, форм подання, базових алгоритмічних структур. Складання описів алгоритмів (ані мовою програмування, ані навчальною алгоритмічною мовою) не передбачається. Найбільш ґрунтовно (не беручи до уваги класи з поглибленим вивчення інформатики) змістова лінія основ алгоритмізації висвітлюється у класах фізико-математичного, технологічного, природничого профілів, у яких крім розгляду всіх питань, передбачених програмою для класів універсального профілю, вивчається складання програм однією з мов програмування високого рівня.

У програмах для 12-річної школи відмінності між рівнем стандарту й академічним рівнем у вивченні основ алгоритмізації і програмування менш відчутні, ніж розглянуті вище.

### **Основні елементи змісту у вивченні основ алгоритмізації**

У процесі вивчення основ алгоритмізації методисти рекомендують виходити з таких положень:

- процес вивчення основ алгоритмізації слід орієнтувати на використання комп'ютера як дидактичного засобу навчання;
- метою вивчення розділу є виділення змісту реального об'єкта — алгоритму — і правил його побудови, а не вивчення конкретної алгоритмічної мови, яка є лише одним із багатьох засобів формального подання ал-

У зорі на вивчення основ алгоритмізації можна виділити такі компоненти:

- навчання відомих алгоритмів і їх використання;
- навчання класичних алгоритмів;
- навчання побудови описів алгоритмів як з використанням, так і без використання відомих алгоритмів.

Побудувати алгоритм означає:

- 1) продумати план майбутньої діяльності;

## **Лекція № 13-14**

### **МЕТОДИКА НАВЧАННЯ ОСНОВ АЛГОРИТМІЗАЦІЇ ТА ПРОГРАМУВАННЯ**

#### **Методичні особливості навчання основ алгоритмізації**

##### ***Роль алгоритмічного й операційного мислення***

Викладання інформатики в школах нашої країни фактично починалося з викладання програмування. У той час навіть було проголошене гасло: «Програмування — це друга грамотність». Зазначимо, що комп'ютери в школах тоді практично були відсутні. А та техніка, якою оснащувалися школи в кінці 80-х — на початку 90-х років минулого століття, практичну складову курсу інформатики все одно мимовільно зводила до програмування. Одночасно з революційним розвитком апаратного і програмного забезпечення і оснащенням сучасною комп'ютерною технікою навчальних закладів курс інформатики зазнав істотних змін. Найяскравіша характеристика такої зміни — вимивання програмування зі шкільного курсу інформатики.

Проте контингент школярів, у яких інтерес саме до вивчення, а не знайомства з програмуванням високий, поза сумнівом, існує. У першу чергу це учні фізико-математичних шкіл, гімназій, ліцеїв. У більшості з них є як мотивація, так і здібності до освоєння програмування. Мотивація є і у вчителя — адже більшість сучасних олімпіад з інформатики є за своєю сутністю олімпіадами з програмування, а за успіхами учнів в олімпіадах часто роблять висновки про кваліфікацію вчителя, хоча у випадку з інформатикою це далеко не безперечно. Крім того, любов до програмування багато вчителів інформатики принесли зі своєї професійної діяльності, і звичайно ж їм хочеться передати цю любов і своїм учням.

Проте вміння і навички, що формуються при вивченні алгоритмізації, мають важливе значення не тільки для майбутніх програмістів. З кожним роком збільшується кількість задач, що можуть бути розв'язані за допомогою комп'ютера. Завдяки

розвитку програмного забезпечення відкривається доступ до комп'ютера так званим «параметричним користувачам», які можуть навіть не знати будови комп'ютера, ні алгоритму розв'язування задачі, ні мови, якою написана програма. Однак для успішного спілкування з комп'ютером навіть параметричному користувачеві необхідні такі вміння й навички, як:

- планування дій, виходячи з обмеженого набору засобів;
- формальний і повний опис усіх об'єктів як даних та результатів, визначення їх взаємозв'язків, тобто побудова інформаційної моделі задачі;
- швидкий пошук інформації;
- правильна побудова своїх команд машині.

Слід відзначити, що перераховані навички мають більш широке значення. Наприклад, вміння планувати необхідно не тільки користувачеві, але й інженеру, агроному, офіцеру, вчителю, економісту тощо. Навички моделювання, тобто передбачення властивостей створеної конструкції, найбільш важливо для творчих працівників, зокрема, технологів, інженерів. Правильна організація пошуку — запорука успіху в роботі будь-кого, хто звертається за потрібними відомостями до різноманітних сховищ людських знань: бібліотек, архівів, інформаційних систем. Нарешті, дисципліна спілкування в будь-якому людському колективі не менш важлива, ніж при «розмові» з комп'ютером. І справа тут не тільки в точності висловлювань чи ввічливості форми. Щоб бесіда вийшла діловою, продуктивною, потрібно враховувати, наприклад, рівень вивчення співрозмовника.

- формалізувати задачу;
- виділити в ній логічно самостійні частини;
- визначити взаємозв'язки цих частин;
- спроектувати алгоритм розв'язування за технологіями «згори-донизу» та «знизу-догори»;
- дібрати якомога ефективніший шлях отримання результату;
- інтерпретувати та аналізувати результати.

Важливим розділом людського знання є алгоритмічне керування технічними (зокрема, комп'ютерними) системам. Крім того, людина у своїй діяльності часто керується різними

алгоритмами. І розуміння сутності своєї власної алгоритмічної діяльності важливо для кожної людини.

### *Мета і завдання вивчення основ алгоритмізації та програмування*

*Мета* навчання основ алгоритмізації — навчити основних способів організації операцій і даних, а також застосувань базових алгоритмічних структур при складанні описів алгоритмів розв'язування різних задач. Необхідно навчити учнів основних способів організації дій в алгоритмах, основних способів організації даних, застосування алгоритмічних конструкцій при складанні алгоритмів розв'язування різних класів задач.

- виявленню загальних закономірностей і принципів алгоритмізації;
- основним етапам розв'язування задач за допомогою засобів сучасних інформаційних технологій;
- аналізові поставленої задачі, методам формалізації та моделювання реальних процесів та явищ;
- добору виконавця, виходячи з того, що він є також певним об'єктом із притаманними йому властивостями та набором допустимих операцій, які слід аналізувати з метою правильного використання;
- методам та засобам формалізованих описів дій виконавця, сучасним засобам їх конструювання та реалізації за допомогою комп'ютера.

Розділ «Основи алгоритмізації та програмування» розкриває важливість алгоритмів, їх роль у функціональному зв'язку понять «інформація — алгоритм — комп'ютер», що визначають процес автоматичного опрацювання інформації. На прикладах варто продемонструвати можливість формального виконання алгоритмів, що дозволяє використати виконавця-машину. Вивчення алгоритмічної мови дозволяє познайомити учнів із формалізованим записом алгоритмів.

Змістова лінія основ алгоритмізації у класах різних профілів відображена по-різному. У класах універсального профілю передбачається формування в учнів поняття алгоритму, уявлення про способи і форми його подання, базові алгоритмічні структури, а також навчання будувати графічні схеми