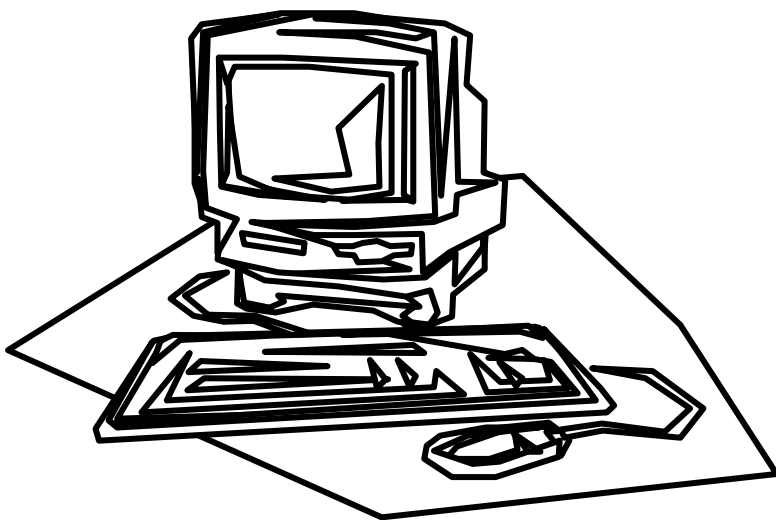


Міністерство освіти і науки України
Чернівецький національний університет
імені Юрія Федьковича

О с н о в и і н ф о р м а т и к и

Методичні рекомендації та практичні завдання



Чернівці
ЧНУ
2007

ББК 82.973.2я73
О-751
УДК 004 (076.5)

Друкується за ухвалою редакційно-видавничої ради
Чернівецького національного університету
імені Юрія Федьковича

О-751 **Основи інформатики:** Методичні рекомендації та
практичні завдання / Укл. Г.В.Мельник. – Чернівці: ЧНУ, 2007.
– 132с.

Методичні вказівки містять практичні завдання, теоретичні
рекомендації та вказівки до виконання лабораторних робіт з курсу «Основи
інформатики».

Для студентів коледжу ЧНУ.

ББК 82.973.2я73
УДК 004 (076.5)

© Видавництво "Рута"
Чернівецького національного
університету, 2007

Розділ 1. Інформаційна система та її складові.	5
Лабораторна робота № 1 Практичне дослідження роботи комп'ютера.	5
Розділ 2. Операційні системи та програми – оболонки	8
Лабораторна робота № 2. Дії з файлами, каталогами, дисками в операційній системі MS DOS . Основні операції в оболонці Norton Commander.	8
Лабораторна робота № 3. Дії з об'єктами в Windows. Налаштування параметрів панелі задач.	13
Лабораторна робота № 4. Головне меню Windows. Пошук об'єктів. Дії з об'єктами в програмі «Проводник».	19
Лабораторна робота № 5. Сервісні програми роботи з дисками. Архівація даних. Захист інформації від комп'ютерного вірусу.	23
Розділ 3. Прикладне програмне забезпечення загального призначення	28
Лабораторна робота № 6. Графічний редактор Paint.	28
Лабораторна робота № 7. Word. Створення документа.	30
Лабораторна робота № 8. Word. Форматування текстових документів. Дії з фрагментами тексту.	34
Лабораторна робота № 9. Word. Побудова таблиць та діаграм. Дії з об'єктами WordArt.	39
Лабораторна робота № 10. Excel. Створення документа.	46
Лабораторна робота № 11. Excel. Форматування та обчислення.	53
Лабораторна робота № 12. Робота з матрицями та списками. Побудова діаграм.	57
Лабораторна робота № 13. Проектування бази даних у середовищі СУБД Access. Створення таблиць БД.	60
Лабораторна робота № 14. Редагування структури таблиці й даних БД. Впорядкування, пошук та фільтрація даних.	65
Лабораторна робота № 15. Зв'язки між таблицями. Запити	69
Лабораторна робота № 16. Об'єкти БД – форми та звіти. Способи створення форм.	75
Лабораторна робота № 17. Створення презентації індивідуального завдання за допомогою Power Point.	80
Розділ 4. Комунікаційні засоби ЕОМ	84
Лабораторна робота №18. Робота в локальній мережі Windows.	84
Лабораторна робота №19. Робота в Інтернет з допомогою браузера Internet Explorer. Дії з повідомленнями в Outlook Express.	87
Розділ 5. Основи алгоритмізації та програмування	95
Лабораторна робота № 20. Тема: Робота в системі Turbo Pascal Робота з екраном	95
Лабораторна робота № 21-22. Робота в системі Turbo Pascal Типи даних.	100
Лабораторна робота № 23-24. Програми з розгалуженою структурою	106
Лабораторна робота № 25-26. Організація циклів.	111

Лабораторна робота № 27-28. Масиви. Алгоритми обробки таблиць.	118
Лабораторна робота № 29-30. Масиви. Методи сортування.	123
Лабораторна робота № 31-33. Процедури та функції.	125
Лабораторна робота № 34-35. Обробка рядків.	129

РОЗДІЛ 1. ІНФОРМАЦІЙНА СИСТЕМА ТА ЇЇ СКЛАДОВІ.

Лабораторна робота № 1 Практичне дослідження роботи комп'ютера.

Загальні вимоги по техніці безпеки та заходах по охороні праці при виконанні лабораторних робіт з курсу «Основи інформатики».

1. Не відлучатися від робочого місця без дозволу викладача, не допускати присутність сторонніх осіб на робочому місці.
2. Не вмикати обладнання без дозволу викладача, виконувати тільки доручену роботу.
3. Забороняється працювати на комп'ютері з мокрими руками, у вогкому одязі, при порушенні цілісності корпусів або ізоляції проводу.
4. При виявленні в процесі роботи вищевказаних дефектів, появи запаху гарі, чи незвичайних звуків, необхідно вимкнути комп'ютер та повідомити викладача.
5. Забороняється залишати включений комп'ютер, вставати та ходити по аудиторії.
6. Забороняється класти на комп'ютер книги, зошити та інші предмети чи опиратися на нього.

Спеціальні вимоги

1. Перед початком роботи:

Провірити приміщення згідно передбачених санітарних норм для проведення лабораторних робіт. Ознайомитися з приладами та обладнанням. Підготувати робоче місце у відповідності з вимогами інструкції для проведення лабораторних робіт. Перед ввімкненням приладу та споживачів енергії переконатися в надійності заземлення їх корпусів

2. Під час роботи:

При роботі з комп'ютером необхідно знати, що він має елементи, які знаходяться під напругою 220 В, тому категорично забороняється проводити розбір чи ремонт у ввімкненому стані. При роботі з обчислювальними комплексами необхідно переконатися, що всі встановлені в ньому заземлення, болти та клеми підключені до контуру заземлення. Необхідно знати, що джерелом електробезпеки є мережа живлення технічних засобів обчислювального комплексу.

Всі операції, по монтажу та демонтажу лабораторного обладнання повинні виконуватися при вимкненій вилці з мережі живлення. При вимкненні схем, які знаходяться під напругою, не можна торкатися заземлених частин схеми, приладу та інше. При експлуатації, або технічному обслуговуванні дисплею може виникнути небезпека враження електричним струмом.

Джерелом електробезпеки є джерело живлення дисплею та блок високої напруги монітора.

Заборонено під час роботи вимикати кабель, що з'єднує між собою системний блок з монітором, блоком клавіатури та зовнішніми приладами.

Заборонена робота на дисплеї при знятих кожухах та задніх стінках системного блоку чи монітору.

При попаданні студента під напругу студент, який знаходиться по сусідству повинен швидко вимкнути обладнання, вимкнувши відповідний прилад чи загальну напругу на щиті живлення та надати першу допомогу. Студентам заборонено ремонт обладнання в лабораторії.

3.Після закінчення роботи:

1. Вимкнути всі споживачі електричної енергії.
2. Здійснити прибирання робочого місця.
3. Здати робоче місце черговому по групі.
4. Черговому по групі здати лабораторію викладачу.

Методичні вказівки

Апаратні засоби — сукупність усіх пристроїв, з яких складається комп'ютер, або які можуть додаватися до нього за необхідності. Базове апаратне забезпечення ПК складають наступні пристрої: системний блок, монітор, клавіатура, маніпулятор «миша».

Системний блок: материнська плата; жорсткий диск («вінчестер»); пристрій для роботи з CD- та DVD-дисками; дисководи для гнучких дискет; блок живлення.

Материнська плата:

До складу материнської плати входять такі основні компоненти:

- процесор, який виконує всі розрахунки та оброблення інформації. Параметрами процесора є тактова частота і тип (модель). Тактова частота характеризує швидкість роботи процесора і виражається в мегагерцах. Останні моделі процесорів мають тактову частоту понад 2 ГГц. Процесори можуть бути таких типів (залежно від виробників): IBM, AMD, CELERON, Pentium (PRO, II, III, IV) тощо;

- оперативна пам'ять (RAM) (Random Access Memory) — містить програми ОС, прикладні програми, а також результати роботи процесора. Обмін інформацією відбувається дуже швидко, оперативно, але дані зберігаються тільки до перезавантаження або вимикання комп'ютера, тобто тимчасово. Ємність RAM — 16, 32, 64, 128, 256, 512 Мбайт;

- ROM (Read Only Memory) – постійний запам'ятовуючий пристрій (програма тестування ПК, BIOS, блок завантаження ОС);

- контролери (адаптери) — електроні плати для виконання обміну даними між процесором і зовнішніми пристроями (наприклад, адаптер монітора, адаптер портів для підключення принтера, миші, джойстика, контролери додаткових пристроїв - стримера, модему, сканера тощо). Контролери (адаптери) взаємодіють із процесором та оперативною пам'яттю через системну магістраль передачі даних, яка називається шиною;

- відеокарта — використовується для роботи з графічними

зображеннями. Має ємність пам'яті до 64 Мбайт;

- звукова карта — виконує оброблення звуків, мови;
- мережна карта — застосовується при роботі в комп'ютерній мережі;
- порт—засіб для підключення периферійних пристроїв до материнської плати. Порти бувають паралельні —LPT1 та послідовні — COM1. Останнім часом набуває популярності порт USB, який дає змогу підключати до 256 пристроїв.

Зовнішня пам'ять ПК:

- накопичувач на жорсткому магнітному диску («вінчестер») - пристрій для тривалого зберігання великих об'ємів інформації та швидкого доступу до неї. Сучасні жорсткі диски можуть зберігати обсяги інформації: 20; 60; 80 Гбайт, 1 Тбайт. Частота обертання шпинделя вінчестера становить 5400—7200 і 10 000 хв⁻¹. При її збільшенні продуктивність диска зростає, але необхідно встановлювати додатковий вентилятор для охолодження.
- накопичувачі на гнучких магнітних дисках - пристрій для зчитування і запису інформації на гнучкі диски, або дискети. Дискети характеризуються діаметром та ємністю. Розповсюджені – дискети діаметру 3,5”(1”=1 дюйм = 2,54 см). Стандартна ємність дискети – 1,44 М
- привід для зчитування інформації з оптичних (лазерних) дисків – CD ROM. Обсяг пам'яті лазерних дисків у декілька сот разів вища за обсяг магнітних дискет: до 700 Мбайт.

Монітор - пристрій для відображення текстової та графічної інформації на екрані (пристрій виведення). Стандартних моніторів немає, їх виготовляють багато фірм (Samsung, LG, Toshiya, Panasonic, Sony). Основними характеристиками монітору є:

- роздільна здатність (кількість пікселів на екрані по горизонталі та вертикалі);
- палітра (кількість кольорів та їх відтінків);
- розміри екрану по діагоналі (в дюймах).

На сьогодні монітор може відтворювати від 2 кольорів (монохромні монітори) до 256 (поліхромні монітори).

Клавіатура - це пристрій введення символічної інформації, а також керування ПК. Клавіші згуртовані в 4 поля:

- I - поле алфавітно-цифрових (символьних) клавіш;
- II - поле функціональних клавіш;
- III - цифрове поле;
- IV - поле управління.

Маніпулятор «миша» - це пристрій, за допомогою якого користувач керує позицією курсору на екрані монітора та обирає об'єкти для роботи.

Хід роботи

1. Отримати інструкції з правил техніки безпеки при роботі з ПК.

2. Ознайомитись з розташуванням основних блоків ПК. Зобразити у зошиті структурну схему будови ПК
3. Ознайомитись з клавіатурою ПК за допомогою спеціальної програми вивчення клавіатури.

Контрольні запитання

1. Які пристрої входять у базове апаратне забезпечення ПК ?
2. Які основні пристрої входять в системний блок?
3. Яке основне призначення мікропроцесора?
4. Що таке RAM і ROM?
5. Які переваги та недоліки зовнішньої пам'яті порівняно з внутрішньою?
6. В яких одиницях вимірюється розмір оперативної пам'яті. Вкажіть мінімальний розмір оперативної пам'яті для нормальної роботи сучасного ПК.
7. Які носії зовнішньої пам'яті вам відомі?
8. Який об'єм пам'яті може зберігатись на вінчестері? на лазерному диску?
9. Яка стандартна ємність дискети розміром 3,5" ?
10. Вкажіть основні характеристики монітора.
11. Опишіть призначення і схематичне розташування клавіш Enter, Esc, BackSpace, SpaceBar, Shift, CapsLock.
12. Назвіть основні характеристики монітора.

РОЗДІЛ 2. ОПЕРАЦІЙНІ СИСТЕМИ ТА ПРОГРАМИ – ОБОЛОНКИ

Лабораторна робота № 2. Дії з файлами, каталогами, дисками в операційній системі MS DOS . Основні операції в оболонці Norton Commander.

Методичні рекомендації

Операційна система MS DOS (MicroSoft Disk Operation System) з 1981 по 1995 рік була основною операційною системою для IBM-сумісних комп'ютерів. В наш час саме вона використовується для виконання ремонтних та налагоджувальних операцій.

За способом організації діалогу (інтерфейсу) з користувачем MS DOS відноситься до *неграфічних* операційних систем., управління в яких здійснюється за допомогою клавіатури. Зоною управління є один рядок екрану, який називається *командним рядком*.

1. Зовнішні і внутрішні команди

Усі команди MS DOS підрозділяються на внутрішні (ті, що обробляються командним процесором command.com) і зовнішні, що є системними утилітами. Так, повна версія MS-DOS поставляється з великим числом додаткових утиліт, що цілком можна розглядати як команди. У версії MS DOS, що входить до складу Windows, є тільки

підтримка внутрішніх команд і основних зовнішніх, необхідних лише для забезпечення базової функціональності і можливості установки системи.

2. Формат команди:

ім'я команди [параметри] — набираються як прописними так і рядковими латинськими літерами.

3. Робота з файлами.

Створення текстових файлів

copy con ім'я файлу

і почергово вводити рядки файлу, натискаючи вкінці Enter, а по завершенню F6 і Enter. Як результат, в рядку запрошення MS DOS з'явиться повідомлення:

1 files copied

і на диску з'явиться файл з I-Ф.

Видалення файлу

del ім'я файлу

Перейменування файлу

ren ім'я файлу1 ім'я файлу2

Копіювання файлу

copy ім'я файлу1 ім'я файлу2

Наприклад:

copy xxx.doc xxx.txt — копіювання файлу xxx.doc в поточному каталозі;

copy a:*.* — копіювання всіх файлів з кореневого каталогу диску a: в поточний каталог;

copy a:*.* d: — копіювання всіх файлів з кореневого каталогу диску a: в поточний каталог диску d: ;

Переміщення файлів в інший каталог

move [/Y] ім'я-файлу ім'я-каталогу

З перейменуванням:

move [/Y] ім'я-файлу [дисковод:][шлях]нове-ім'я-файлу

Якщо при переміщенні файлу в каталозі-одержувачі вже є файл із таким ім'ям, програма Move видає запит про дозвіл перезапису файлу. При наявності параметра /Y перезапис файлів виконується без яких-небудь запитів.

Наприклад:

move *.doc d: — пересилання файлів з розширенням .doc у поточному каталозі в поточний каталог диска D: ;

move /y *.doc arc — пересилання файлів з розширенням .doc у поточному каталозі в підкаталог ARC.

move thesis.bak a:\thesis.old - пересилання файлу thesis.bak в кореневий каталог диску A: з перейменуванням цього файлу;

4. Робота з каталогами:

Команда зміни дисководу

Формат команди

A: — перехід на дисковод A:

C: — перехід на дисковод C:

Зміна поточного каталогу

cd [дисковод:] шлях

Приклади:

cd \ — перехід у кореневий каталог поточного диска;

cd \exe \MS DOS — перехід в каталог \exe\MS DOS .

Перегляд каталогу

dir [дисковод:][шлях\][ім.'я-файлу] [параметри]

Найчастіше вживані параметри:

/P — по екранний вивід змісту. При вказівці цього параметра MS DOS , після заповнення екрану буде чекати доти, поки користувач не натисне будь-яку клавішу, після чого буде виводити наступну сторінку;

/W — вивід даних у стислому форматі: приводяться тільки імена файлів і підкаталогів, причому ці імена виводяться по п'ять у кожному рядку:

Приклади:

dir — вивести зміст поточного каталогу;

dir *.exe — вивести зведення про файли з розширенням .EXE з поточного каталогу;

dir /p — вивести зміст поточного каталогу з паузами при заповненні екрана;

dir /w — вивести зміст поточного каталогу в широкому форматі;

Створення каталогу

md [дисковод:]шлях

Пиклади:

md xxx — створення підкаталогу xxx у поточному каталозі;

md a:\work - створення підкаталогу work у кореневому каталозі диску a: .

Знищення (порожнього) каталогу

rd [дисковод:]шлях

5. Оболонки операційної системи.

Оболонка ОС – це спеціальна програма, надбудова над ОС, що дозволяє просто та зручно виконувати найважливіші команди ОС.

Оболонки MS DOS : Norton Commander, PC Shell, Xtree, MS DOS Navigator, Volkov Commander, Far Manager.

Функції оболонок:

- відображати на екрані вміст каталогів різних дисків;
- без набору команд швидко та зручно виконувати операції над файлами та каталогами;
- переглядати та редагувати текстові файли;
- виконувати команди так само, як вони виконуються в середовищі ОС.

Завантаження програми здійснюється файлом NC.EXE (VC.COM)

6. Вигляд екрану

Два прямокутних вікна, обмежених рамкою – панелі. Зверху – рядок меню, знизу – рядок підказки, нижче розміщується звичайне запрошення MS DOS . Там можна вводити звичайні команди MS DOS з можливістю коректування за допомогою клавіш: →, ←, Home, End, Ctrl+→, Ctrl+←, BackSpace, Ctrl+BackSpace, Del, Esc. Щоб вивести в командний рядок ім'я файлу чи каталогу, що підсвічені курсором на панелі, необхідно натиснути Ctrl+Enter.

Меню NC:

Left, Right – налагодження параметрів відображення на лівій та правій панелях.

Files – операції над файлами

Disk – операції над дисками.

Commands – операції загального призначення.

Options – налагодження робочого середовища.

Призначення функціональних клавіш

- | | | |
|------------|--------|--|
| F1 | Help | – допомога; |
| F2 | Menu | – виклик меню користувача; |
| F3 | View | – перегляд файлу; |
| F4 | Edit | – редагування файлу; |
| F5 | Copy | – копіювання файлу (каталогу); |
| F6 | RenMov | – перейменування або переміщення файлу (каталогу); |
| F7 | MkDir | – створення каталогу; |
| F8 | Delete | – знищення файлу (каталогу); |
| F9 | PullDn | – виклик меню NC; |
| F10 | Quit | – вихід. |

Вивід рядка з підказками значень функціональних клавіш можна скасувати натисканням **Alt+B**. Повторне натискання **Alt+B** відновлює рядок з підказками.

На кожній панелі може зображатися:

- зміст каталогу на диску;
- дерево каталогів на диску;
- підсумкова інформація про диск та каталог.

Одна з панелей є активною (чи поточною), інша – неактивною.

Клавіші керування панелями:

Tab – зробити активною іншу панель;

Ctrl+O – забрати/вивести панелі з екрану;
Ctrl+U – поміняти панелі місцями;
Ctrl+F1 – забрати ліву панель;
Ctrl+F2 – забрати праву панель;
Alt+F1 – обрати на лівій панелі інший диск;
Alt+F2 – обрати на правій панелі інший диск.

Хід роботи:

1. Завантажити командний рядок MS DOS .
2. Вивести вміст диску A:.
3. В кореновому каталозі диску C: створити новий каталог, ім'ям якого є ваше прізвище.
4. Перейти в створений каталог.
5. Створити в ньому файл, що містить інформацію: прізвище, ім'я, адреса, дата народження.
6. Перейменувати цей файл.
7. Перемістити цей файл в кореневий каталог диску A:.
8. Перейти в кореневий каталог диску A:.
9. Переглянути вміст диску A.
10. Перейти в кореневий каталог диску C:.
11. Закрити вікно командного рядка MS DOS .
12. Завантажити NC.
13. Одержати допомогу з NC.
14. Викликати меню NC.
15. Зробити ліву панель інформаційною. Визначити кількість вільного місця на диску C.
16. Створити в кореновому каталозі диску C свій каталог.
17. Перейти в нього.
18. Створити текстовий файл і записати в нього інформацію про свій розклад на вівторок.
19. Продивитися створений файл.
20. Перейменувати створений файл.
21. Скопіювати ваш каталог з усім вмістом на дискету.
22. Знищити з диску C створений каталог.
23. Вийти з NC.

Контрольні запитання

1. Які команди MS DOS називаються внутрішніми, а які - зовнішніми?
2. Якою командою можна скопіювати файл ?
3. Які параметри має команда dir?
4. Яке призначення та функції оболонок операційних систем? Наведіть приклади.
5. Яке призначення функціональних клавіш в програмі NC?

Лабораторна робота № 3. Дії з об'єктами в Windows. Налаштування параметрів панелі задач.

Методичні рекомендації

Операційна система Windows це набір драйверів і системних бібліотек. Коли завантажується комп'ютер, після старту ядра MS DOS завантаження передається завантажнику Windows. Він завантажує необхідні драйвери, які замінюють функції вводу-виводу MS DOS і системні бібліотеки, що відповідають за інтерфейс. Лише після цього запускається графічний інтерфейс і програма робочого столу Windows -explorer.exe. При цьому, для того щоб завантажувалися саме ті файли і програми, що потрібні, задіюються дані із системного реєстру Windows.

Системним реєстром називають спеціальну базу даних, у якій зберігається інформація про конфігурацію комп'ютера, про встановлені програми і навіть про переваги користувача. Робота як самої Windows, так і програм, що функціонують під її керуванням, багато в чому спирається саме на дані реєстру.

1. Режими завантаження Windows

Режимами роботи управляє стартове меню, яке з'являється на початку завантаження.

Стартове меню містить:

Normal (звичайний режим) – відбувається стандартне завантаження Windows.

Logged (створення протоколу завантаження) – аналогічний Normal, але створюється файл Bootlog.txt, в який заноситься інформація про завантажені драйвери та виконані команди.

Safe mode (безпечний режим) – використовується драйвер VGA і відсутня підтримка мережі.

Safe mode with network support (безпечний режим з підтримкою мережі) – аналог попереднього, але завантажує мережеві драйвери.

Step-by-step confirmation (покрокове підтвердження) – перед виконанням кожної команди питає підтвердження.

Command prompt only (тільки командний рядок) – не завантажується графічна оболонка.

Safe mode command prompt only (командний рядок в безпечному режимі) – аналог безпечного режиму, але без графічного інтерфейсу.

Previous version of MS-DOS (попередня версія MS-DOS) – завантажує попередню версію MS-DOS.

2. Об'єкти Windows

Під „*об'єктом*” розуміють все чим оперує ОС Windows: програму, групу програм, папку, документ, мережу, диск, пристрій тощо.

Файл — це іменована послідовність байтів довільної довжини. Оскільки із цього визначення випливає, що файл може мати нульову

довжину, то фактично створення файлу складається в присвоєнні йому імені й реєстрації його у файлової системі. Ім'я файлів можуть складатися з 255 символів (за виключенням таких символів \ / : * ? " < >), але краще надавати файлам імена за системою 8.3 (8 символів – власне ім'я файлу, 3 – розширення імені файлу).

Документом в ОС Windows називають будь-який файл, що містить дані: текст, графічне зображення тощо.

Каталоги — важливі елементи ієрархічної структури, необхідні для забезпечення зручного доступу до файлів, якщо файлів на носії занадто багато. Файли поєднуються в каталоги по будь-якій загальній ознаці, заданій їхнім творцем (по типі, по приналежності, по призначенню, за часом створення й т.п.). Каталоги низьких рівнів вкладаються в каталоги більше високих рівнів й є для них *вкладеними*. Верхнім рівнем вкладеності ієрархічної структури є *кореневий каталог* диска. Отже для користувача каталог це місце, де зберігається документи, програми та інші підкаталоги, а з точки зору ОС – це місце на диску, в якому зберігаються імена файлів, відомості про розмір, час останнього поновлення кожного файлу і т. д.

До появи операційної системи Windows при описі ієрархічної файлової структури використався введений вище термін **каталог**. З появою цієї системи був уведений новий термін — **папка**. Утім, що стосується обслуговування файлової структури носія даних, ці терміни рівнозначні: кожному каталогу файлів на диску відповідає однойменна папка операційної системи. Основна відмінність понять **папка** й **каталог** проявляється не в організації зберігання файлів, а в організації зберігання об'єктів іншої природи. Так, наприклад, в останніх операційних системах сімейства Windows існують спеціальні папки, що представляють собою зручні логічні структури, яким не відповідає жоден каталог диска.

Піктограма – це невеличка картинка, яку ставлять у відповідність об'єкту чи діям в ОС Windows. За зовнішнім виглядом піктограми можна визначити тип об'єкту чи дію, яким вона відповідає.

Ярлик – це посилання на програму, документ, диск, принтер або на будь який інший ресурс комп'ютера. Для того, щоб можна було відрізнити ярлик об'єкту від самого об'єкта, в лівому нижньому куті ярлика міститься маленька зігнута стрілочка. Один об'єкт може мати скільки завгодно ярликів, які можуть знаходитись в різних папках файлової системи. Це дозволяє заощадити дисковий простір, оскільки при копіюванні об'єкта, представленого піктограмою на новому місці створюється фізична копія файлу, а при використанні ярлика тільки файл із даними самого ярлика.

Меню – це короткий перелік допустимих дій в даному вікні чи з даним об'єктом. Окремі пункти меню відкриваються в підменю. Меню відкривають, клацнувши по потрібному пункту лівою клавішею миші. **Контекстне меню** відкривають клацнувши по об'єкту правою клавішею миші. Пункти контекстного меню залежать від об'єкту, з яким воно пов'язане. Використання контекстного меню значно спрощує та прискорює

процес роботи з об'єктами, оскільки немає потреби шукати відповідні команди у меню вікон.

3. Вікна ОС Windows

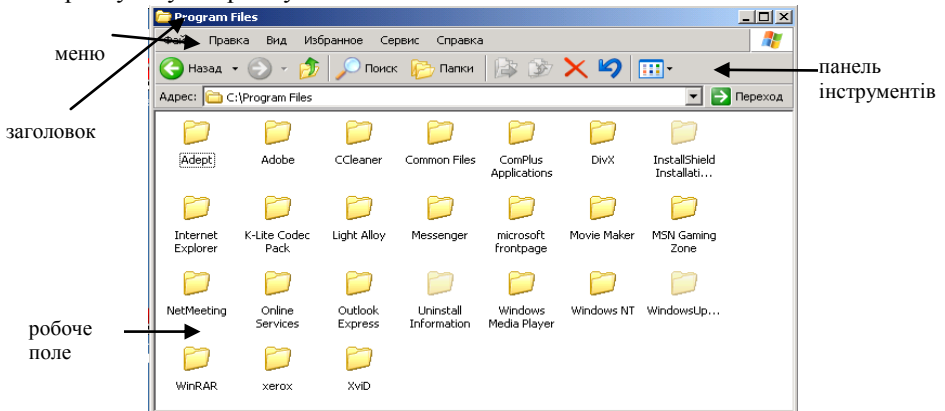
В ОС Windows розрізняють наступні типи вікон:

1. робочі;
2. діалогові;
3. вікна повідомлень.

Робочі вікна, це вікна в яких під рядком заголовка розташований рядок меню, чого немає в інших вікнах ОС Windows.

Перелік інтерфейсу робочого вікна:

- **Заголовок** містить кнопки, які дозволяють згорнути у піктограму, розгорнути на весь екран та закрити вікно. Вони розміщені по правому краю заголовку. В лівому кутку розміщена кнопка-піктограма на яку відкривається системне меню управління вікном.
- В **меню** вікна приведені всі операції, які можна в ньому виконувати.
- **Панелі інструментів** – піктограми, які відповідають окремим підпунктам меню.
- **Робоче поле** може містити ярлики інших об'єктів; в ньому виконується програма.
- **Смуги прокрутки** дозволяють зміщувати робоче в потрібному користувачу напрямку.



Діалоговим називають вікно, яке містить різні засоби діалогу з користувачем. Засоби діалогу в цих вікнах служать вкладки, перемикачі, прапорці, поля-списки, текстові поля, командні кнопки.

Вікна повідомлень містять повідомлення операційної системи, які користувач змушений прийняти до відома. Ці вікна завжди виводяться поверх інших вікон. Зазвичай при появі вікна повідомлень подальша робота неможлива до відповіді на запит та закриття вікна повідомлень.

З вікнами можна виконувати такі дії:

1. Перемішувати по екрану монітора, „зацепивши” мишкою за заголовок;

2. Збільшити чи зменшити, навівши курсор на край вікна (курсор при цьому змінить свій вигляд) і „потягнувши” мишкою.

Оскільки ОС Windows є багатозадачною системою, на екрані монітора можуть одночасно знаходитись декілька вікон програм, але ОС Windows „повинна знати”, до якої програми належать ті чи інші дії користувача. Тому в кожний момент часу тільки одне вікно є **активним**, а решта є **неактивними**. Заголовок активного вікна виділяється іншим кольором (як правило - синім), ніж заголовки неактивних вікон. Активне вікно завжди розташоване поверх неактивних вікон. Для того, щоб зробити вікно активним, треба натиснути мишею в будь-якому місці вікна.

4. Робочий стіл

Стартовий екран Windows являє собою системний об'єкт, який називається *Робочим столом*. **Робочий стіл** — це *графічне середовище*, на якому відображаються об'єкти Windows та елементи керування. У вихідному стані на Робочому столі можна спостерігати кілька екранних значків і Панель задач.

Інтерфейс *Робочого столу* містить:

1. панель задач;
2. ярлики та піктограми об'єктів.

Панель задач - це один з основних елементів управління Windows. Вона, як правило розташовується в нижній частині екрана – це смуга звичайно сірого кольору. На панелі задач розміщені:

- кнопка пуск, яка відкриває головне вікно Windows;
- ідентифікатор дати і часу;
- кнопка перемикання мови клавіатури;
- мінімізовані вікна відкритих програм та документів у вигляді кнопок з іменем об'єкту.

Панель задач можна перемістити, змінити розмір та захвати. Переміщення панелі виконується, „протягуванням” мишею у обране місце, а зміна розміру - „зачепленням” за край панелі. Змінити режим функціонування панелі можна через її контекстне меню (пункт **Свойства**).

Ярлики та піктограми, які встановлюються на робочому столі при завантаженні системи, це:

Мій комп'ютер Ця піктограма дозволяє відобразити список різних ресурсів комп'ютера, таких як жорсткі диски або пристрої читання гнучких дисків і компакт-дисків, а також будь-яких мережних ресурсів, до яких зроблене підключення, наприклад принтерів і мережних дисків. Для перегляду вмісту ресурсу двічі клацніть його піктограму

Сетевое окружение служить для доступу до комп'ютерів та принтерів які знаходяться в локальній мережі, а також для налагодження локальної мережі.

Корзина призначена для тимчасового зберігання вилучених файлів. Вона дозволяє відновлювати файли, вилучені помилково. Насправді вилучені файли не зникають із жорсткого диска доти, поки корзина не буде очищена. Щоб видалити файл, перетягніть його за допомогою миші на значок *Корзини*. Для очищення *Корзини* двічі клацніть її значок і виберіть у контекстному меню команду **Очисти корзину**. Необхідно періодично очищати *Корзину*, щоб звільнити місце на диску. Якщо видалити файл із гнучкого диска, він не зберігається в *Корзині*, а знищується без можливості відновлення.

5. Створення об'єктів.

Щоб створити об'єкт всередині папки, з контекстного меню треба вибрати пункт New (Создать) та те, що ви хочете створити: Папка, Ярлик, Текстовый документ, Точечный рисунок чи інш.. Додаткові пункти в цьому меню будуть з'являтися при встановленні на комп'ютер нових програм. Інший варіант створення об'єктів – меню: Файл → Создать...

При створенні папки та файлу достатньо ввести ім'я. Створення ярликів потребує додаткових дій: ввід шляху та імені об'єкта, для якого створюється ярлик, ввід імені та в деяких випадках (програма MS DOS) вибір значка.

6. Дії над об'єктами

Вікна папок можна використовувати для різноманітних дій над файлами: створення (контекстне меню, меню), копіювання (комбінація Copy-Paste, перетягування правою кнопкою миші, з допомогою меню), перейменування (контекстне меню, меню), переміщення (комбінація Cut-Paste, перетягування правою кнопкою миші, з допомогою меню), знищення (контекстне меню, меню, перетягування в Корзину. При перетягуванні лівою кнопкою миші створюється ярлик. Якщо виконана невірна дія, можна її відмінити з допомогою команди меню Undo (Правка – Отменить).

Для вибору групи об'єктів разом з мишею треба використовувати клавіші Shift(для вибору неперервної групи) та Ctrl (для довільної групи).

Хід роботи:

1. Завантажити Windows у режимі Normal.
2. Вивчити зовнішній вигляд Робочого столу (елементи оформлення та елементи керування).
3. Змінити зовнішній вигляд Робочого столу через контекстне меню, змінюючи при цьому:
 - Фон;
 - Заставка;

- Оформлення;
 - Налаштування області екрану.
4. Переглянути результати роботи. Змінити налаштування області екрану на попередні.
 5. Змінити зовнішній вигляд значків Робочого столу:
 6. Переглянути результати роботи. Через контекстне меню кожного з цих значків повернути їм попередній зовнішній вигляд:
 7. Перемістити піктограму «Мои документы» в інше місце робочого столу. Викликати контекстне меню для піктограми «Мои документы».
 8. Відкрити вікно папки «Мой компьютер».
 9. Вивчити структуру цього вікна. Визначити основні елементи вікна.
 10. Згорнути, розгорнути, відновити, перемістити та змінити розміри цього вікна.
 11. Змінити вигляд відображення інформації в цьому вікні.
 12. Впорядкувати значки в цьому вікні.
 13. Відключити та включити Панель інструментів, Рядок адреси, Рядок стану.
 14. Продивитися об'єм вільного місця на диску С.
 15. Продивитися вміст диску D.
 16. Включити режим відображення всіх характеристик файлів.
 17. На диску С створити папку з назвою відповідно Вашого прізвища.
 18. Створити текстовий документ та ярлик до нього у Вашій папці.
 19. Перемістити Вашу папку на Робочий стіл.
 20. Знищити Вашу папку.
 21. Створити на Робочому столі ярлик для програми Microsoft Excel.
 22. Запустити програму Microsoft Excel з допомогою ярлика.
 23. Змінити значок для цього ярлика.
 24. Знищити ярлик.
 25. Очистити кошик.
 26. Змінити розміщення Панелі задач на вертикальне.
 27. Додати німецьку мову в список доступних мов.
 28. Змінити системний час та дату.

Контрольні питання:

1. Які існують режими роботи Windows ?
2. Що таке контекстне меню?
3. Як можна змінити оформлення елементів Windows ?
4. Яке призначення програми “Корзина” ?
5. Які пункти меню мають програми «Корзина» та «Мой компьютер»?
6. Що розуміти під поняттям об'єкт Windows ?
7. Які властивості мають об'єкти “файл” та “папка” ?
8. Що спільного і чим відрізняються ярлик та значок?
9. З допомогою якого елемента Робочого столу можна переключатися між відкритими вікнами папок та програм?

10. В чому різниця дії кнопок «Копировать» і «Вырезать»?
11. Чим відрізняється активне вікно від пасивного(неактивного)? Як зробити вікно активним?
12. Назвати способи розміщення ярликів на Робочому столі.

Лабораторна робота № 4. **Головне меню Windows. Пошук об'єктів. Дії з об'єктами в програмі «Проводник».**

Методичні рекомендації

1. Головне меню Windows.

Головне меню — один з основних системних елементів керування Windows . Воно відрізняється тим, що незалежно від того, наскільки *Робочий стіл* перевантажений вікнами запущених процесів, доступ до *Головного меню* зручний завжди — воно відкривається щигликом на кнопці **Пуск**. За допомогою *Головного меню* можна завантажити всі програми, що встановлені під керуванням операційної системи або зареєстровані в ній, відкрити останні документи, з якими виконувалася робота, одержати доступ до всіх засобів налаштування операційної системи, а також доступ до пошукової й довідкової систем Windows.

У структуру Головного меню входять два розділи: **обов'язковий і довільний**. Довільний розділ розташований вище розділювальної риски. Пункти цього розділу користувач може створювати за власним бажанням. Іноді ці пункти утворюються автоматично при встановленні деяких додатків.

Обов'язкові пункти головного меню:

Програми - Відкриває доступ до ієрархічної структури, що містить покажчики для запуску додатків, що встановлені на комп'ютері. Для зручності користування покажчики поєднуються в категорії. Якщо категорія має значок у вигляді трикутної стрілки, у ній є вкладені категорії. Розкриття вкладених категорій виконується простим зависанням покажчика миші

Документи - Відкриває доступ до ярликів останніх п'ятнадцяти документів, з якими даний користувач працював на комп'ютері

Настройка - Відкриває доступ до основних засобів настроювання Windows, зокрема, до логічної папки Панель **управлення**. Служить також для доступу до папки **Принтеры**, через яку здійснюється встановлення принтерів і настроювання завдань для друку.

Найти - Відкриває доступ до засобів пошуку, що встановлені на комп'ютері. Основним є засіб **Файлы и папки**, за допомогою якого здійснюється пошук об'єктів у файловій структурі.

Выполнить - Цей пункт відкриває невелике вікно, яке містить командний рядок для завантаження програм. Його зручно використовувати для завантаження додатків MS-DOS .

Завершение работы - Коректний засіб закінчення роботи з операційною системою. Відкриває діалогове вікно *Завершение работы*, що

містить наступні пункти: • призупинити роботу комп'ютера; • виключити комп'ютер; • перезавантажити комп'ютер; • перезавантажити комп'ютер у режимі MS-DOS .

2. Пошук об'єктів

I. Пошук файлу по імені та типу.

Запуск пошукової системи: Главное меню: Пуск >Найти >Файлы и папки.

Пошук файлу по імені і типу - найбільш розповсюджений метод пошуку. Якщо ім'я файлу відомо разом з розширенням, його можна ввести в поле *Искать имена файлов или папок*; у поле *Где искать...* задається шлях доступу до папки, починаючи з якого передбачається вести пошук, і завантажити пошук командною кнопкою *Найти*. Можна задати пошук по цілому диску чи по всьому комп'ютеру, хоча такий пошук буде працювати не так швидко. Якщо ж коло пошуку можна хоч трохи звужити, папку для пошуку вибирають у списку, що розкривається. Щоб задати конкретну папку, виберіть пункт *Просмотр*.

II. Пошук файлу за датою створення чи зміни.

Цей зручний тип пошуку нерідко застосовують при аналізі причин ненормальної роботи комп'ютера чи програми. Так, наприклад, якщо з якихось причин нормальна робота деякої програми припинилася, можна довідатися, які системні файли одержали зміни останнім часом. Можна встановити, чи не співпадає час їхньої зміни з якоюсь подією, наприклад, з установкою на комп'ютері іншої програми, з аварійним перезавантаженням комп'ютера та інш.

Для настроювання подібних параметрів - *Параметры поиска*.

Якщо потім встановити прапорець *Дата*, побачите елементи управління для пошуку файлів, створених (змінених) у визначений період часу чи за останні кілька днів (місяців).

III. Пошук файлу по фрагменту вмісту або розміру.

Пошук по фрагменту вмісту відбувається досить повільно, оскільки системі доводиться не просто перевіряти атрибути файлів, але і переглядати їх вміст. Цей вид пошуку застосовують у випадках, коли про ім'я файлу невідомо нічого, але відомі ключові слова, що можуть зустрітись у ньому. Обмеження на розмір файлів, які переглядаються, дозволяє звужити коло пошуку і, тим самим, скоротити час пошуку. Фрагмент тексту, що міститься у файлі, можна ввести в поле *Искать текст*. Обмеження на розмір файлу задають точно так само, як при пошуку по даті - досить встановити прапорець *Размер* на відкритій панелі *Параметры поиска*.

Усі задані параметри пошуку враховуються одночасно. Якщо задати і вміст файлу, і його розмір, і дату створення, то знайдені файли задовольняють всім умовам одночасно.

3. Дії з об'єктами в програмі «Проводник».

Вікно програми складається з двох панелей. Ліва називається **панеллю папок**, а права - **панеллю вмісту**. Зміст двох панелей полягає в тому, що для виконання операцій копіювання чи переміщення файлів і папок треба мати два об'єкти: джерело і приймач. Об'єкти вибираються в джерелі і перетягуються в приймач.

На лівій панелі представлено файлову структуру комп'ютера. В основі файлової системи лежить папка *Робочий стол*, від якої тягнуться лінії зв'язку до інших об'єктів, значки яких присутні на *Робочем столі*. Так, зокрема, одна з гілок з'єднує *Робочий стол* і *Мой компьютер*, від якого, у свою чергу, тягнуться гілки до значків дисків і до їх файлових структур. У місцях, де гілки верхнього рівня з'єднуються з гілками нижнього рівня, утворюються вузли. Ці вузли представлені значками + чи - . Вузли можна згортати і розгорнути. Так ми можемо дістатися до будь-якої папки на лівій панелі, після чого папку можна розкрити чи закрити. При цьому вид значка міняється. Відкриття папки супроводжується зміною вмістимого правої панелі.

I. Засоби роботи з “Проводником”.

Відкриття файлу. На лівій панелі розшукайте папку, у якій зберігається потрібний файл, після чого на правій панелі відкрийте файл подвійним щикликом на його значку. Що при цьому відбудеться, залежить від типу файлу.

Копіювання файлу. На лівій панелі розшукайте папку, у якій зберігається потрібний файл, і розкрийте її. На правій панелі розшукайте значок файлу, але поки ніяких дій не починайте. На лівій панелі розгорніть папку, у яку бажаєте скопіювати файл. Якщо необхідно, використовуйте смуги прокрутки для перегляду вмісту панелей. Далі перетягніть значок файлу, що копіюється з правої панелі на значок папки-приймача на лівій панелі. Відпускати кнопку миші треба в той момент, коли папка-приймач на лівій панелі виділяється.

Переміщення файлу. Механізм цієї операції такий же, але виконується вона спеціальним, перетягуванням (при натиснутій правій кнопці).

Створення ярлика в іншій папці. Дійте так само, як і при переміщенні файлу, але по закінченні операції виберіть в контекстному меню пункт *Создать ярлык*.

Переименовання файлу. Потрібний файл на правій панелі, відкрийте контекстне меню і виберіть пункт *Переименовать*, після чого введіть нове ім'я в поле напису.

Видалення файлу. В контекстному меню до потрібного файлу на правій панелі виберіть пункт *Удалить*. Файл буде вилучений у *Корзину* після позитивної відповіді на додатковий запит. Якщо файл виділений, то видалити його можна і за допомогою клавіші **Delete**.

II. Групові операції з файлами

Виділивши цілу групу файлів, можна оперувати з нею точно так само, як з окремим файлом.

Повне виділення. **Ctrl+A.**

Виділення підряд. Для виділення потоку файлів підряд, користуються **Shift.**

Довільне виділення. Складання довільної групи файлів виконують при натиснутій клавіші **Ctrl.**

Хід роботи:

1. Завантажити програму **Калькулятор.**
2. Одержати список останніх документів, що редагувалися.
3. Відкрити папку “Мои документы”, використовуючи кнопку **Пуск.**
4. Вивести вікно настройки **Панелі задач.** Налогодити **Панель задач** так, щоб вона автоматично забиралася з екрана.
5. Додати в **Главное меню Пуск** новий пункт.
6. Відкрити **Панель управління.**
7. Змінити швидкість переміщення вказівника миші.
8. Переглянути параметри системи.
9. Одержати список встановлених шрифтів.
10. З’ясувати, чи встановлені на комп’ютері принтери та їх характеристики.
11. Одержати довідку Windows. Відшукати інформацію про пошук об’єктів в Windows.
12. Відшукати на диску C всі файли графічного типу створені протягом січня.
13. Відшукати на всіх дисках файли, що починаються з літери „А” та мають тип „Документ Microsoft Word”.
14. Відшукати в папці „Мои документы” всі файли розміром більше 50 Кбайт.
15. Завантажити „Проводник” (декілька способів).
16. Вивести вміст елементу “Мой компьютер”.
17. Вивести вміст диску C: у вигляді списку.
18. Продивитися наявність папок нижчого рівня.
19. Вивести вміст елементу “Мои документы”.
20. Створити в “Мои документы”: папку “Ім’я” та папку “Прізвище”.
21. Створити в папці “Ім’я” Документ Microsoft Word та ярлик для нього.
22. Перемістити створений ярлик на **Робочий стол.**
23. Перемістити документ у папку “Прізвище”.
24. Перейменувати файл, який створили.
25. Знищити з Робочого столу створений ярлик.
26. Знищити створену папку з “Мои документы”.
27. Очистити корзину.
28. Закрити програму „Проводник”.

Контрольні питання:

1. Яке призначення головного меню?
2. Яке призначення пункту **Програми**?
3. Яке призначення пункту **Найти**?
4. Як знайти необхідний файл?
5. Як доповнити Головне меню новим пунктом?
6. Яке призначення пункту **Настройка**?
7. Якими способами можна завантажити програму «Проводник»?
8. Для чого призначена програма «Проводник»?

Лабораторна робота № 5. Сервісні програми роботи з дисками. Архівація даних. Захист інформації від комп'ютерного вірусу.

Методичні рекомендації

1 Робота з магнітними дисками

1.1 Форматування диска

Перед першим застосуванням диск форматують.

Форматування диска — процес нанесення на його поверхню спеціального магнітного сліду, за яким здійснюється записування або читання даних.

Під час форматування диск поділяється на концентричні кола — доріжки (треки), а доріжки — на сектори. Позначаються на ньому і дефектні місця. Їх позиції заносяться в таблицю розподілу файлів **FAT (File Allocation Table)**.

Форматування диска застосовується для:

1. підготовки до використання жорсткого або логічного диска, що розміщений на жорсткому диску;
2. підготовки дискети, з якої можна завантажувати ОС (системної дискети);
3. очищення дискети від інформації та виділення дефектних ділянок.

Повторне форматування диска виконується в разі:

1. виникнення фізичних пошкоджень або дефектних місць, коли інформація не читається;
2. зараження вірусом, коли не можна вилікувати файли без втрати інформації.

Форматують диск командою **Форматировать**, яку можна вибирати так:

- відкрити папку **Мой компьютер**. Потім клацнути правою клавішею миші на значку диска, який треба відформатувати. У контекстному меню, що з'явиться на екрані, активізувати команду **Форматировать**;
- відкрити програму **Проводник**, вибрати значок диска на панелі вмісту вибраної папки і клацнути правою клавішею миші. У контекстному меню, що з'явиться на екрані, задати команду **Форматировать**.

В обох випадках відкривається діалогове вікно **Форматирование**, що містить такі параметри:

«Емкость». У списку, що розкривається, треба вибрати ємність диска (наприклад, *1,44 Мбайт*);

«Быстрое (очистка оглавления диска)» — повне стирання інформації на раніше відформатованому диску;

«Полное» — повне форматування диска з перерозміткою секторів. Використовується для форматування нових дисків (при цьому всі погані сектори на диску буде знайдено);

«Только копирование системных файлов». На відформатований диск записуються системні файли (диск стає системним);

«Метка», «Без метки». Довільна мітка вводиться до поля «Метка». Якщо мітку не вказувати, то треба ввімкнути перемикач «Без метки»;

«Вывести отчет о результатах» — на екран виводиться вікно з інформацією про форматування диска (доступна кількість байтів, кількість зіпсованих секторів та ін.);

«Скопировать на диск системные файлы» — у процесі форматування на диск записуються системні файли (диск стає системним).

Після вибору опцій треба клацнути мишею на командній кнопці **«Начать»**, щоб почалося форматування диска.

Після закінчення форматування дискети на екрані відображається вікно **«Результати форматирования»**,

Примітки

1. Не можна відформатувати диск, що містить папку з ОС **Windows** (про це на екрані з'являється повідомлення).

2. Не можна формувати мережні диски (у їхньому контекстному меню немає команди **Форматировать**).

1.2 Дискові сервісні програми

Швидкість розв'язання багатьох задач, а іноді й принципова можливість їх розв'язання, залежать від стану зовнішніх запам'ятовуючих пристроїв. Для перевірки стану дисків і поліпшення їх якості у Windows використовується кілька сервісних програм для:

- перевірки диска;
- архівації даних;
- дефрагментації диска.

1.2.1 Перевірка диска

Під час виконання операцій введення/виведення бувають випадки, коли на деяких ділянках диска (кластерах) є дані, а інформації про ці кластери немає у змісті диска, чи навпаки, у змісті вказано два посилання на один кластер. Це так звані логічні помилки на диску. Крім них, можуть траплятися фізичні помилки, коли пошкоджено магнітний шар на деяких кластерах. Тому треба періодично робити перевірку якості диска й усувати обидва види помилок за допомогою сервісної програми **ScanDisc**. При виявленні логічних помилок ця програма об'єднує загублені кластери в один

файл. Якщо в них була важлива текстова інформація, то за змістом її можна буде надалі відновити. При виявленні фізичних пошкоджень магнітного шару програма **ScanDisc** позначає кластери як непридатні для майбутнього записування в них інформації.

1.2.2 Архівація даних

Коли на диску зберігається важлива інформація і потрібно зробити її резервну копію на дискетах чи на жорсткому диску іншого комп'ютера в локальній мережі, то можна скористатися службовою програмою **Backup**.

Якщо у разі архівації виявляється, що для розміщення даних недостатньо однієї дискети, ця програма запропонує вставити другу дискету, потім третю і т. д. Відновлення копії з дискет на жорсткий диск здійснюється в тій же послідовності — спочатку в дисковод вставляють першу дискету і задають команду на відновлення, потім другу і т. д.

1.2.3 Дефрагментація диска

Якщо записувати файли на диск, то спочатку вони займають послідовні суцільні ділянки. Після видалення файлів утворюються вільні місця, куди можна записувати нові файли. Але ці ділянки часто бувають вже несуміжними. Тому новий файл, який користувач бажає зберегти, спочатку записується на першу вільну ділянку. Якщо його розмір більший за розміри останньої, то його залишок записується на другій вільній ділянці, потім на третій і т. д.

Як наслідок, один файл може розміщуватися на кількох несуміжних ділянках. Процес оброблення такого файлу сповільнюється. Якщо файлів, що фрагментами зберігаються на різних ділянках диска, виявляється багато, то і робота в цілому з таким диском стає повільною. Тому рекомендується періодично робити упорядкування файлів — дефрагментацію дисків. Ця процедура полягає у тому, що файли перезаписують у такий спосіб, щоб вони займали неперервні ділянки, а вільний простір на диску починався тільки після останнього файлу.

1.2.4 Завантаження сервісних програм

Програми для обслуговування дисків можна завантажити, вибравши у меню **Пуск** команду **Программы— Стандартные—Служебные—<Потрібна програма>**.

Якщо необхідно обробити який-небудь конкретний диск, то найчастіше слід:

- у вікні папки **Мой компьютер** клацнути правою клавішею миші на значку потрібного диска;
- у контекстному меню, що з'явилося, вибрати команду **Свойства**, а у вікні властивостей — вкладку **Сервис**;
- у вкладці **Сервис** вибрати одну з програм, натиснувши відповідну кнопку: **Проверка диска**, **Архивация данных**, **Дефрагментация диска**.

У вікні **Дефрагментация диска** виводиться детальна інформація про поточний стан диска. Щоб спостерігати за процесом дефрагментації, потрібно в цьому вікні натиснути кнопку **Сведения**. Розшифрування

умовних позначок можна прочитати у вікні *Легенда*, що викликається натисканням однойменної кнопки.

1.3 Робота з архівами на прикладі WinRAR

Програми-архіватори дають змогу заощаджувати місце на архівних дисках завдяки ущільненню інформації й об'єднують групи файлів у один архівний файл.

Архівний файл — файл або група файлів, записаних у стисненому вигляді в єдиний файл, з якого їх можна відновити.

На початку архівного файлу розташовується його зміст, який має:

- ім'я файлу;
- відомості про папку, в якій є вихідний файл;
- розмір вихідного файлу на диску й у стисненому вигляді в архіві;
- код циклічного контролю файлу для перевірки цілісності архіву.

Програми для архівації:

- записують файли в архів у стисненому вигляді та добувають їх з архіву у початковому вигляді;
- додають, відновлюють, переміщують і перейменовують файли в архіві;
- вилучають файли з архіву;
- переглядають зміст архіву та ін.

Програми-архіватори різняться форматом ущільнення, швидкістю роботи, ступенем стиснення файлів, зручністю використання тощо.

Для стиснення інформації найчастіше використовують програму-архіватор **WinRAR**. Її потрібно активізувати і виконати команду **Файл/Добавить в архив**. У вікні, що відкриється слід надати архіву ім'я та вибрати зручну для вас програму-архіватор і спосіб архівації (звичайний, максимальний, швидкий, хороший). Розархівацію файлів проводять командою **Файл/Извлечь** файли, **Файл/Извлечь в текущую папку**. Двічі клацнувши лівою клавішею миші по архіву, ви потрапите у вікно програми **WinRAR**, в якому відображена повна інформація про архів. Вибираючи файли зі списку та користуючись меню чи піктограмами ви зможете переглянути, знищити, сортувати, розархівовувати та додавати файли в архів.

1.4 Робота з антивірусним програмним забезпеченням.

Комп'ютерний вірус – це програма, яка, потрапивши разом із файлом на диск, може самостійно прописувати своє тіло (код вірусу) до інших файлів, використовуючи файлову систему диску і таким чином розмножуватись. Для виявлення та ліквідації вірусів розроблені антивірусні програми. Найпопулярнішими є **Doctor Web, AdInf, McAfee, AVP**.

Doctor Web - один із кращих антивірусів з потужним алгоритмом пошуку вірусів. Містить поліфаг, який здатний перевіряти файли в архівах, документи Word і робочі книги Excel, виявляє навіть поліморфні віруси. Аналізатор DrWeb, досліджуючи програми на наявність фрагментів коду, характерних для вірусів, дозволяє знайти майже 90% невідомих вірусів. При

завантаженні програми, у першу чергу DrWeb перевіряє самого себе на цілісність, після чого тестує оперативну пам'ять. Програма може працювати в діалоговому режимі, має зручний інтерфейс користувача, який набудовується.

Антивірус **AVP** (AntiVirus Program) відноситься до поліфагів, у процесі роботи перевіряє оперативну пам'ять, файли, у тому числі архівні, на гнучких, локальних, мережних і CD-ROM дисках, а також системні структури даних, такі як завантажувальний сектор, таблицю розділів і т.д. Програма має евристичний аналізатор, що здатний знаходити майже 80% усіх вірусів. Програма AVP є 32-розрядним додатком для роботи в середовищі операційних систем Windows 98, NT і 2000, має зручний інтерфейс, а також одну з найбільших у світі антивірусну базу. Бази антивірусів до AVP обновлюються приблизно один раз на тиждень і їх можна одержати з Internet. Ця програма здійснює пошук і вилучення різноманітних вірусів, у тому числі: поліморфних вірусів; «стелс»вірусів (вірусів-невидимок); нових вірусів для Windows; «макровірусів», що інфікують документи Word і таблиці Excel.

Крім того, програма AVP здійснює контроль файлових операцій у системі у фоновому режимі, виявляє вірус до моменту реального зараження системи, а також визначає невідомі віруси за допомогою евристичного модуля.

Хід роботи.

1. Відформатувати дискету з такими параметрами: **Полное, метка** за прізвищем студента, **Вывести отчет о результатах, Скопировать на диск системные файлы.**
2. Здійснити перевірку диску **C** за допомогою програми перевірки диска **ScanDisc**. Встановити перемикач **Автоматично правити помилки** та виконати перевірку. При виявленні помилок на диску ознайомтеся з повідомленням про помилку й виберіть серед пропонованих варіантів виправлення той, при якому помилка ігнорується й робота триває далі.
3. Ознайомитись з роботою програми **Дефрагментация диска**. Розгляньте вікно **Легенда**.
4. Заархівувати один з своїх текстових файлів різними способами, та проведіть аналіз на скільки відсотків відбулось стиснення файлу тим чи іншим способом. Додайте до цього архіву ще один файл. Розархівуйте файли у новостворену папку.
5. Завантажити антивірусну програму. Записати її назву. Продивитися і занотувати можливі задачі для програми. Перевірити вміст дискети на наявність вірусів. При необхідності провести лікування. Перевірити наявність вірусів у папці "Мої документи".

Контрольні питання

1. Що таке форматування диска. В яких випадках використовують

форматування диска?

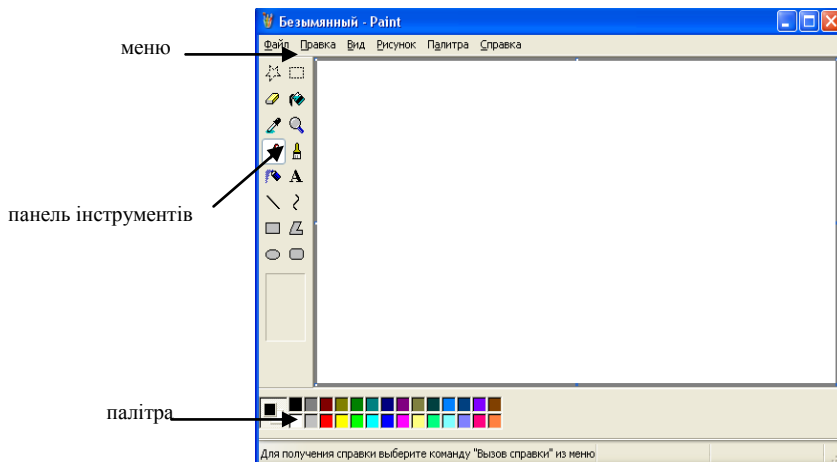
2. Для чого використовується перевірка диска?
3. Що таке дефрагментація?
4. Що таке архівація. Назвіть програми архіватори.
5. У чому полягає різниця між архівами rar та zip?
6. Що таке комп'ютерний вірус?
7. За якими ознаками класифікуються комп'ютерні віруси ?
8. На які типи поділяються антивірусні програми за методом перевірки ?
9. Вкажіть 2-3 відомих вам параметрів антивірусної програми.

РОЗДІЛ 3. ПРИКЛАДНЕ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ЗАГАЛЬНОГО ПРИЗНАЧЕННЯ

Лабораторна робота № 6. Графічний редактор Paint.

Методичні рекомендації

Операційна система Windows містить в собі програму Paint, зі допомогою якої можна вивчити основні прийоми роботи з комп'ютерною графікою. Графічний редактор Paint служить для роботи з растровими зображеннями, тобто з зображеннями, що складаються з окремих



кольорових точок (пікселів).

Запуск програми Paint здійснюється вибором команди **Пуск = Программы => Стандартные => Paint**. У вікні Paint розміщені елементи властиві саме графічним редакторам - панель інструментів і палітра. Панель інструментів розміщена зліва у вікні та містить 16 кнопок інструментів:

Карандаш, Кисть, Ластик, Распылитель, Линия Кривая, Прямоугольник, Эллипс, та інші.

Призначення кожної кнопки можна прочитати на спливаючій підказці, що з'являється при наведенні на неї вказівника миші.

Інший елемент вікна - палітра кольорів - розміщений у нижній частині екрана. Основний колір (колір переднього плану) обирається клацанням лівої клавіші миші на відповідному квадратику палітри кольорів, а колір фону - клацанням правої клавіші. Створення графічних зображень відбувається за допомогою інструментів, наприклад, інструмент **Кривая** використовують для малювання зігнутих S-подібних ліній.

Фрагменти створених зображень можна переміщати, копіювати, вирізати, вставляти. Ці операції проходять аналогічно як і з файлами. Попередньо потрібно виділити фрагмент малюнка. Для цього клацніть мишею по інструменту **Выделение**, потім клацніть мишею в полі малюнка й протягніть вказівку миші через фрагмент малюнка, який потрібно виділити. Відпустіть кнопку миші і навколо фрагмента з'явиться прямокутна пунктирна рамка, яка має вісім маркерів.

Щоб розтягнути або стиснути фрагмент, наведіть покажчик миші на один з маркерів рамки виділення. Покажчик набере форми двоспрямованої стрілки, тепер можна мишею перетягнути маркер. Щоб перемістити виділений фрагмент, клацніть по ньому мишею і перетягніть його на потрібне місце.

Ще одним типом перетворень виділених фрагментів малюнка є нахили по горизонталі і по вертикалі па різні кути. Ці операції здійснюються за допомогою команди меню **Рисунок => Растянуть/Наклонить**.

Щоб видалити невдалий фрагмент малюнка, потрібно виділити фрагмент і натиснути клавішу Delete. Очищення всього малюнка можна виконати командою **Рисунок => Очистить**.

Малюнок, створений у Paint, можна вставити в інший документ. Для цього можна скористатися можливостями буфера обміну. Виділіть фрагмент малюнка у Paint і виконайте команду **Правка => Копировать**. Потім перейдіть в цільовий документ і вставте вміст буфера обміну за допомогою відповідної команди, наприклад, **Правка ==> Вставить**.

Для збереження малюнка на диску оберіть команду **Файл => Сохранить как...** і в наступному діалоговому вікні задайте папку, в яку буде записуватися файл, а також ім'я файлу і клацніть по кнопці **Сохранить**.

Завдання для виконання.

1. У новому графічному документі намалюйте малюнок за зразком, що зображено у завданні (згідно варіанту).
2. Розфарбуйте малюнок. Дайте йому назву. Напишіть її у верхній частині екрану. У нижній правій частині екрану вкажіть свої прізвище і групу.

Лабораторна робота № 7. Word. Створення документа.

Методичні рекомендації

Текстовий редактор, наприклад Word, можна завантажити декількома способами: за допомогою ярлика програми-редактора, піктограми редактора, меню.

Командою **Файл ⇒ Відкрити** відкривають існуючий документ, а **Файл ⇒ Новий** — створюють новий. Відкриється вікно текстового редактора, що міститиме вікно документа з назвою, наприклад, Вірш (рис. 1).

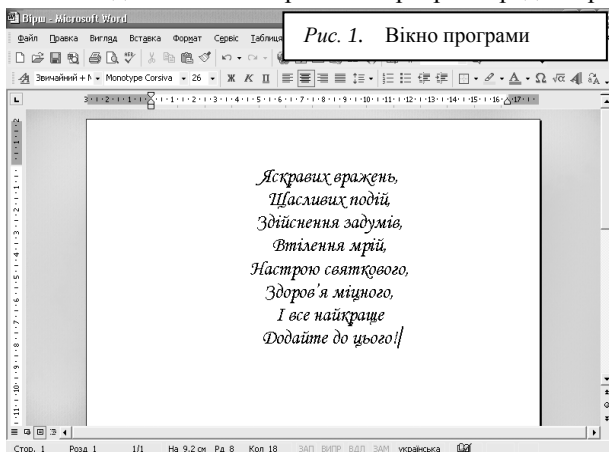


Рис. 1. Вікно програми

Вікно програми складається з:

- рядка заголовку, в якому є назва програми і документа та кнопки керування вікном програми;
- рядка командних меню з кнопками керування вікном документа;
- панелей інструментів;
- робочого поля документа зі смугами прокручування і кнопками завдання вигляду документа;
- рядка статусу.

Наявність у робочому полі I-подібного (текстового) курсору дає змогу вводити текст. Однак часто перед першим сеансом користувач готує середовище до роботи. За допомогою команд основного меню або кнопок панелей інструментів, що їх дублюють, задають чи перевіряють ввімкнення потрібних параметрів (режимів) функціонування програми – редактора. Найчастіше виконують такі підготовчі дії:

1) Задають параметри сторінки;

Для визначення параметрів сторінок необхідно виконати команду **Файл => Параметри сторони**. Діалогове вікно **Параметри сторони** містить такі опції:

Поля - для визначення відступів тексту від краю аркуша та відступу для палітурки документа та його розташування (зліва або зверху);

Орієнтація – для визначення орієнтації документа: **Книжна** - висота документа більша від ширини чи **Альбомна** - ширина документа більша від висоти;

Страницы - у цьому списку визначається розташування сторінок у документі: **обычный** - звичайного набору сторінок; **зеркальные поля** - чергування полів сторінок; **2 страницы на листе** - поділ аркуша горизонтальною лінією на 2 сторінки; **брошюра** - поділ аркуша на 2 сторінки вертикальною лінією.

На закладці **Размер бумаги** вводиться:

Размер бумаги - шляхом вибору зі списку стандартних розмірів;

Ширина, высота - безпосереднім введенням нестандартних розмірів у сантиметрах. При цьому в попередньому списку **Размер бумаги** потрібно вибрати зі списку значення **другой**;

На закладці **Источник бумаги** користувачеві надається можливість: зі списку **Начать раздел** вибрати початок поточного розділу; у полі **Страница** встановити горизонтальне вирівнювання в документі. Вирівнювання за висотою можна виконувати тільки для заповнених аркушів, інакше виконується вирівнювання за верхнім краєм; додаткові можливості оформлення документа створення колонтитулів і різноманітних меж.

За допомогою поля **Образец** можна переглянути зовнішній вигляд сторінки документа.

2) задають розміри абзацного відступу та міжрядкового інтервалу;

Для визначення цих параметрів необхідно виконати команду **Формат => Абзац => Закладка Отступы и интервалы**, яка містить такі опції: **Отступ** – для визначення абзацного відступу зліва та справа; **Интервал** – для визначення міжрядкового інтервалу всередині абзацу та інтервалу перед і після визначеного абзацу.

3) вмикають засоби автоматичного перевіряння правопису;

4) присвоюють паролі документу, якщо його треба захистити.

Для виконання останніх дій використовують діалогове вікно **Параметры: Сервис => Параметры ...** (рис.2)

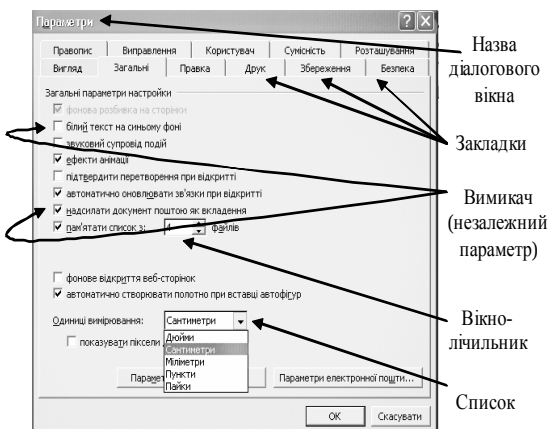


Рис. 2. Діалогове вікно

Увімкнути чи вимкнути *Панель інструментів* можна засобами пункту **Вид** і підпункту **Панель інструментов**. Найчастіше користуються такими панелями: *Стандартна, Форматування, Рисування*.

Є декілька способів *Відображення документа* на екрані для візуального огляду *звичайний, веб - документ, режим розмітки, режим структури*. Найчастіше задають вигляд в режимі розмітки (найбільш зручний).

Зручно увімкнути *лінійку* (підпункт *Лінійка* меню *Вигляд*), щоб простежити за горизонтальними і вертикальними розмірами сторінки, абзацними відступами тощо.

Увімкнувши *панелі та лінійку* (якщо вони були вимкнуті), задавши та перевіряючи параметри, можна вводити текст.

Розглянемо найважливіші дії, які можна виконати над текстовим документом. За допомогою команд з пункту **Файл** чи кнопок стандартної панелі інструментів можна:

- ◆ створити новий (**Создать...** , Ctrl + N);
- ◆ відкрити старий (**Открыть...** , Ctrl + O);
- ◆ зберегти на диску (**Сохранить** , Ctrl + S);
- ◆ зберегти з новою назвою (**Сохранить как...**);
- ◆ закрити (**Заккрыть**);
- ◆ друкувати (**Печать...** , Ctrl + P);
- ◆ задати параметри сторінки (**Параметры страницы...**).

Закінчити роботу з редактором можна або засобами пунктів меню **Файл** ⇒ **Выход**, або закривши усі вікна (Alt + F4).

Хід роботи

1. Завантажте текстовий редактор і розгляньте його меню. Запуск рекомендується виконати, використовуючи ярлик редактора або піктограму документа у форматі Word.
2. Увімкніть панелі інструментів *Стандартну* та *Форматування* і *лінійку*. Задайте вигляд документа *Звичайний*.

Вид ⇒ **Панель инструментов** ⇒ Натисніть кнопку з назвою відповідної панелі.

Вид ⇒ **Линейка. Вид** ⇒ **Обычный**.

3. Задайте параметри сторінки.

Файл ⇒ **Параметры страницы...** ⇒ У відкритій закладці **Поля** задайте усі чотири поля сторінки по 2 см ⇒ Розгорніть закладку **Размер бумаги**, клацнувши на назві, і переконайтеся, що формат сторінки є *A4*, а орієнтація *Книжкова* (вертикальна) ⇒ Закрийте діалогове вікно **Параметры страницы**, натиснувши на **ОК**.

4. Ознайомтеся з описом стилю *Звичайний*.

Формат ⇒ **Стили и форматирование...** У списку стилів знайдіть назву **Обычный**, наведіть на неї курсор і зачекайте ⇒ з'явиться опис стилю.

Закрийте вікно **Стили и форматирование**.

5. Змініть шрифт та його розмір (наприклад, задайте *Courier New* і 18). Скористайтесь відповідними кнопками панелі інструментів.

6. Переконайтеся, що вигляд шрифту – нормальний, і задайте спосіб вирівнювання тексту – з обох боків.
- Кнопки жирний (**Ж**), курсив (*К*), підкреслений (П) шрифт ненависні.
7. Задайте сантиметри як одиниці вимірювання на лінійці.
- Сервис ⇒ Параметри...** ⇒ На закладці **Общие** задайте одиниці вимірювання — *сантиметри* ⇒> Перейдіть на закладку **Правописание**.
8. Перевірте, чи задано параметри: *Автоматично перевіряти орфографію і Завжди пропонувати заміну?*
- На закладці **Правописание** задайте ці режими. ⇒ Перейдіть на закладку **Сохранение**.
9. Задайте час автозберігання документа — 23 хвилин.
- Введіть число 23 у відповідне віконце на закладці **Сохранение**. Перейдіть на закладку **Безопасность**.
- Задайте два паролі: для відкриття документа і (такий самий) для дозволу записувати зміни у файл. Пароль для всіх такий: 12345. ⇒ Натисніть на **ОК** ⇒ Підтвердіть паролі, повторно їх ввівши ⇒ Закрийте вікно **Параметры**, натиснувши на **ОК**.
10. Виберіть мову для перевірки правопису — українську.
- Сервис ⇒ Язык ⇒ Выбрать язык...** ⇒ Виберіть мову у вікні - списку — *українська* ⇒ **ОК**.
11. Задайте режим перенесення слів.
- Сервис ⇒ Язык ⇒ Расстановка переносов** ⇒ *Автоматичне розставлення переносів* ⇒ **ОК**
12. Задайте абзацний відступ 1 см.
- Пересуньте верхній трикутник на лінійці праворуч на 1 см.
13. Дайте назву створюваному документу — ваше Прізвище і слово Док1.
- Файл ⇒ Сохранить как ...** ⇒ Клацніть у діалоговому вікні текстове поле ім'я файлу і введіть текст: *Прізвище Док1* ⇒ Закрийте вікно, натиснувши на кнопку **Сохранить**.
14. Введіть текст, не форматуєючи його.
15. Збережіть текст на диску.
16. Перевірте правопис і виправте помилки.
- Сервис ⇒ Правописание**.
17. Скільки слів і символів є у набраному тексті?
- Файл ⇒ Свойства ⇒ Статистика**.
18. Збережіть текст.
19. Ліквідуйте захист пароллями і збережіть текстовий документ з попередньою назвою.
20. Закрийте всі вікна. Закінчіть роботу.

Лабораторна робота № 8. Word. Форматування текстових документів. Дії з фрагментами тексту.

Методичні рекомендації

Відформатувати текст означає надати йому такого вигляду, який потрібен користувачу. Форматування виконують за допомогою *команд відповідного меню* або *кнопок панелі форматування* (рис. 3), які дають змогу задати таке:

- ♦ стиль документа;
- ♦ шрифт;
- ♦ розмір символів у пунктах (28 пунктів = 1 см);
- ♦ стиль шрифту: товстий, курсив, підкреслений;
- ♦ вирівнювання: до лівого краю, до центру, до правого краю, двобічне;
- ♦ нумерований список;
- ♦ маркований (позначений) список;
- ♦ зсув фрагмента ліворуч;
- ♦ зсув фрагмента праворуч;
- ♦ текст у рамці тощо.

Для форматування тексту його треба спочатку *виокремити* (кажуть також: *вибрати, виділити*). Цей текст буде виділений інверсним кольором. Текст виокремлюють так: клацають на початку фрагмента і перетягують курсор над текстом, утримуючи затиснутою ліву клавішу миші.

Розглянемо швидкі способи виокремлення елементів тексту:

- ♦ щоб виділити слово, треба клацнути на ньому двічі;

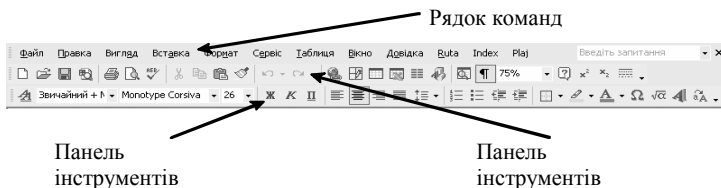


Рис. 3 Меню програми Word, панелі Стандартна і Форматування.

- ♦ щоб виокремити рядок, потрібно клацнути в лівому полі рядка;
- ♦ щоб виділити речення, треба натиснути на клавішу **Shift** і клацнути двічі в межах речення;
- ♦ щоб виділити абзац, потрібно клацнути тричі;
- ♦ щоб виокремити увесь текст (усі абзаци), треба виконати комбінацію **Ctrl+A**.

До виділеного фрагмента можна застосувати додаткові ефекти перетворення шрифтів (**Формат** ⇒ **Шрифт**), зокрема, створення верхніх чи нижніх *індексів*, *закреслень* і *підкреслень*, *контурів* і *тіней*, *інтервалів* тощо; а також *ефекти анімації*: феєрверк, неонova реклама, мурашки, мерехтіння та ін.

Елементами тексту є *символ, слово, речення, абзац*, які називатимемо фрагментом тексту. Над фрагментом тексту визначені такі основні дії: *копіювання* в буфер обміну, *забирання* (вирізання) в буфер обміну, *вилучення* (видалення) з тексту, *вирівнювання, перетворення* символів: збільшення, зменшення, завдання стилю написання (товстий, курсив, підкреслений та їхні комбінації) тощо.

Буфер обміну, призначений для зберігання фрагмента тексту з метою його копіювання чи переміщення як у межах документа, так і в інші документи.

Виокремлений (вибраний, виділений) фрагмент тексту *копіюють* у *буфер обміну* одним із трьох способів:

- командою **Копировать** з меню **Правка**;
- натисканням на кнопку Копіювати  .(стандартна панель інструментів);
- комбінацією клавіш **Ctrl + C**.

Вміст буфера обміну можна *вставити в текст* у місце, де є курсор, одним із трьох способів:

- командою **Вставить** з меню **Правка**;
- натисканням на кнопку Вставити  стандартної панелі;
- комбінацією клавіш **Ctrl + V**.

Виділений фрагмент можна *забрати з тексту* в буфер обміну також одним із трьох способів:

- командою **Вирезать** з меню **Правка**;
- натисканням на кнопку Вирізати 
- комбінацією клавіш **Ctrl + X**.

Виокремлений фрагмент *вилучають з тексту*, не заносючи в буфер обміну, натисканням на клавішу Del (Delete).

Для введення фрагментів, які часто повторюються, використовують автотексти — тексти-взірці, які спочатку заносять у бібліотеку автотекстів.

Щоб вставити автотекст, достатньо набрати на клавіатурі його перші чотири символи і натиснути на клавішу вводу.

Рисунок у текстовому документі можна побудувати, увімкнувши Панель інструментів **Рисование** (рис. 4), за допомогою кнопок з зображенням *лінії, прямокутника, еліпса* тощо. Курсор миші (а він набуде вигляду хрестика) треба розмістити в потрібному місці документа, клацнути і перемістити, щоб створити фігуру заданого розміру (автоматично може створюватися „полотно” для малювання – окреслена область з написом Створіть малюнок.).

Еліпс розтягують до утворення *кола*, перетягуючи маленькі прямокутники-маркери, що є навколо нього, або під час малювання

утримують натиснутою клавішу Shift. Аналогічно створюють *квадрат* — малюють прямокутник з затиснутою клавішею Shift.

Фігури можна *переміщувати, розфарбовувати, розтягувати, стискати, змінювати товщину ліній, накладати* одну на одну, *розміщувати* в них текст різного кольору, створювати *об'ємні ефекти* чи ефекти затінювання, *повертати*. Ці дії виконують за допомогою кнопок панелі малювання, які мають такі назви: Дії, Вибір об'єктів, Автофігури, Лінія, Стрілка, Прямокутник, Овал, Напис, Об'єкт WordArt, Додати діаграму, Додати картинку, Додати малюнок, Колір заливки, Колір ліній, Колір шрифту, Тип лінії, Тип штриха, Тип стрілки, Стиль тіні, Об'єм.

Можна скористатися з меню готових фігур (Автофігури), зокрема, для побудови блок-схем. Щоб ліквідувати невдалу фігуру, її треба вибрати (один раз клацнути на ній лівою кнопкою миші) і натиснути на клавішу Del. Нарисовані фігури є об'єктами — вони мають свої контекстні меню (які викликають звичайно — правою кнопкою миші), що полегшує роботу з ними.

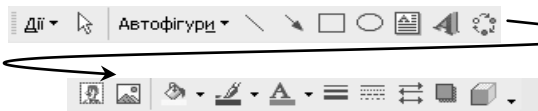


Рис. 4 Панель інструментів Малювання.

Зауваження: Щоб виконати будь-які дії над елементами тексту чи об'єктами, їх спочатку треба виокремити, а для цього — клацнути в його межах лівою клавішею миші.

Хід роботи

1. Завантажте програму-редактор і підготуйте документ до введення тексту.

Паролі не задавайте. Перевірте і задайте основні параметри і режими, а інші можете змінити пізніше.

2. Введіть перший текст — титульну сторінку звіту до лабораторної роботи. Введіть текст так:

Міністерство освіти України

Коледж Чернівецького національного університету

Основи інформатики

Лабораторна робота № 11

Тема: Редактор текстів. Форматування текстів

Варіант: № (номер комп'ютера)

Виконав: прізвище та ініціали

Група

Дата: березня 2007 р.

Викладач: прізвище та ініціали

Чернівці — 2006

3. Розташуйте текст на сторінці відповідно до зразку:

Міністерство освіти України
Коледж Чернівецького національного університету

Основи інформатики

Лабораторна робота № 11
Тема: Редактор текстів. Форматування текстів

Варіант: № (номер комп'ютера)

Виконав: прізвище та ініціали
Група

Дата: березня 200 р.

Викладач: прізвище та ініціали

Чернівці 2006

4. Перевірте, чи оптимально розташований текст на аркуші.

Застосуйте кнопку **Предварительный просмотр** на панелі інструментів або відповідну команду **Файл** ⇒

Предварительный просмотр. Закрийте попередній перегляд: кнопка **Закрити**.

5. Задайте різні перетворення шрифтів і ефекти анімації над окремими фрагментами.

Формат ⇒ **Шрифт** ⇒ **Анимация** ⇒ **Вид** (вибираємо зі списку) ⇒ **ОК**

6. Збережіть текстовий документ з новою назвою — **Прізвище2.doc**

7. Створіть новий документ

(тип звичайний), який міститиме діловий лист.

Структура ділового листа така: вверху ліворуч — адреса і дата, нижче — звертання, основний текст (6 абзаців), прощання і підпис.

8. Збережіть текстовий документ з назвою — **Прізвище3.doc**

9. Виправте граматичні помилки у листі. Форматуйте текст:

Заголовок зробіть великим шрифтом; перший абзац — товстим, другий — курсивом, третій — підкресленим курсивом; підпис виконайте одним з декоративних шрифтів (Monotype Corsiva, Comic Sans MS, Allegro BT або ін.).

10. Застосуйте до трьох абзаців тексту перетворення *Нумерований список*, а до решти трьох *Маркований список* за допомогою **Формат** ⇒ **Список...** ⇒ потрібна закладка...

11. Збережіть виправлений текст.

12. Створіть діловий лист за допомогою майстра листів.

Файл ⇒ **Создать...** ⇒ **Общие шаблоны** ⇒ закладка **Письма и факсы**: вибираємо **Мастер писем**; створити — **Документ**; **ОК**.

У меню майстра клацніть на **Создать одно письмо**. Виконайте чотири кроки (увага! — дату, привітання, звертання і т.д., що пропонуються за замовчуванням можна змінювати, клацнувши у відповідному *текстовому полі*):

➤ На закладці **Формат листа** клацніть у полі *Дата* (введіть дату); виберіть один із *шаблонів*, наприклад, Сучасний лист; виберіть *стиль*, наприклад, Звичайний. Натисніть на кнопку Далі і перейдете до наступної закладки.

➤ На закладці Одержувач введіть *ім'я одержувача й адресу*; виберіть *привітання* і діловий тип листа; натисніть на кнопку Далі.

➤ Закладку Інше можете не заповнювати ⇒ Натисніть на кнопку Далі.

➤ У закладці Відправник введіть *ім'я відправника*; адресу вводити не обов'язково; виберіть форму *процання* і натисніть на кнопку ГОТОВО.

Отримаєте майже готовий лист, куди потрібно внести основний текст — можна використати текст з пунктів 8-9 (скопіювати зміст файлу Прізвище.doc та вставити в поточний документ).

13. Збережіть документ під назвою Діловий лист.

14. Відкрийте текстовий документ Прізвище3.doc .

15. Перший абзац перемістіть у кінець тексту, використовуючи буфер обміну.

16. Другий абзац повторіть двічі.

17. Заголовок відцентруйте, зробіть товстим курсивом і задайте розмір символів 16 пунктів.

18. Перевірте правопис у тексті. Виправте помилки.

19. У другому абзаці замініть усі букви "а" в тексті на своє ім'я.

Розмістіть курсор на початку тексту. Виконайте команду **Замініть...** з меню **Правка**. У текстовому полі **Найти** введіть символ "а", у полі **Замініть на...** — своє ім'я, натисніть на кнопку **Замініть все**, закрийте вікно.

20. Відшукайте другу за порядком букву "с" у тексті і замініть її на своє прізвище.

Розмістіть курсор на початку тексту. Виконайте команду **Замініть** ⇒ Заповніть два текстові поля ⇒ Натисніть на кнопку **Найти далее** два рази ⇒ Натисніть на кнопку **Замініть** ⇒ Закрийте вікно.

21. Створіть автотекст: своє прізвище, ім'я та по-батькові і вставте його п'ять раз в кінці тексту (як різні абзаци).

Щоб створити автотекст, треба виконати такі дії з меню **Вставка: Автотекст** ⇒ **Автотекст** ⇒ Введіть фрагмент у текстове поле ⇒ **Добавить** ⇒ **ОК**.

Щоб вставити автотекст, треба набрати його перші чотири символи і натиснути на клавішу вводу (**Enter**) або виконати команди з меню: **Вставка** ⇒ **Автотекст** ⇒ **Обычный** ⇒ Клацнути на потрібному тексті у списку.

22. Відмініть останню дію — останнє вставляння і збережіть текст.

Виконайте команду **Отменить** з меню **Правка** або комбінацією клавіш **Ctrl + Z**.

23. Вставте номери сторінок у верхньому правому кутку, першу сторінку не нумеруйте.

Виконайте команду **Номера страниц...** з меню **Вставка** і виберіть у діалоговому вікні потрібне розміщення з запропонованих списків.

24. Перегляньте як розташований текст на сторінці.

25. Збережіть текст.

26. Нарисуйте в кінці тексту прямокутник, а під ним розташуйте два кола. Від прямокутника до кіл проведіть лінії-стрілки. Виберіть по черзі фігури і замалюйте їх різними кольорами. У прямокутник введіть назву групи, а в кола - своє прізвище та прізвище друга. Для введення тексту активізуйте контекстні меню фігур і виконайте команду **Добавить текст**.

27. Проекспериментуйте з об'ємними ефектами для прямокутника та тіннями для кіл. Зробіть зображення ефектним.

Виберіть (виокремте) прямокутник, клацніть на кнопки з зображенням куба і вибирайте ефект з отриманого меню. Виберіть коло, клацніть на кнопки з тінню від прямокутника і виберіть ефект.

28. Вставте автофігуру "Усміхнене обличчя".

Натисніть на кнопку **Автофігури**. Виберіть пункт **Основные фигуры** і клацніть на потрібній. Клацніть у документі. Проекспериментуйте з фігурою: замалюйте її зеленим кольором, змініть товщину контурів, поверніть — повертайте фігуру, захопивши зелений кружечок курсором миші.

29. Вставте у документ декілька будь-яких блоків блок-схеми.

Натисніть на кнопку **Автофігури** і виберіть блоки з меню пункту **Блок схема**.

30. Збережіть документ під назвою *Малювання*.

31. Продемонструйте текст і рисунки викладачеві.

Лабораторна робота № 9. **Word. Побудова таблиць та діаграм. Дії з об'єктами WordArt.**

Методичні рекомендації

Такі дані як, наприклад, списки працівників фірм з анкетними даними, інформацію про друзів чи дані виробничого характеру варто наводити у вигляді таблиць. Таблиці призначені для наочного подання інформації.

Елементами таблиці є *клітинки* (комірки), *рядки*, *стовпці*, *рамки* і *дані*, які є в клітинках (див. рис. 5).

Таблиці створюють засобами пункту **Таблица** основного меню. Є три способи створення таблиць за допомогою таких команд:

- ◆ Вставити таблицю;
- ◆ Намалювати таблицю;
- ◆ Перетворити текст в таблицю.

Розглянемо перший спосіб. Внаслідок виконання команди **Таблиця** ⇒ **Вставити** ⇒ **Таблицю...** отримаємо *Діалогове вікно*, де треба задати кількість стовпців і рядків майбутньої таблиці. Після натискання на кнопку **ОК** порожня таблиця розташується в документі там, де стояв текстовий курсор. Вводять у таблицю дані і відображають рамки.

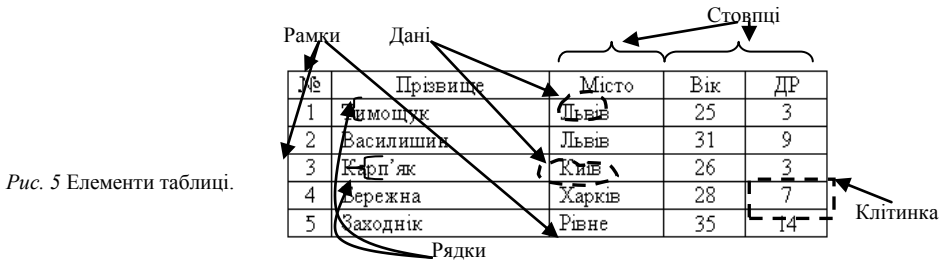


Рис. 5 Елементи таблиці.

Другий спосіб. Після виконання команди **Таблиця** ⇒ **Нарисувати таблицю**, з'явиться *панель інструментів Таблиці та межі* (рис.6).

Курсор миші набуде вигляду олівця. На цій панелі вибирають тип лінії, її товщину, колір. Розмістивши курсор-олівець у потрібному місці документа, рисують структуру таблиці: спочатку зовнішній прямокутник, а пізніше — усі потрібні лінії. Клацнувши двічі в клітинці, можна вводити дані. Щоб продовжити рисування, треба клацнути на кнопці з зображенням олівця, а щоб витерти намальоване — вибрати гумку.



Рис.6 Панель інструментів Таблиці та межі

Третій спосіб є дуже зручний: спочатку треба ввести дані, не задумуючись над виглядом таблиці. Між даними обов'язково треба набирати *розділювач* — символ, якого немає в даних, наприклад, кому, крапку чи інше. Введений текст виділяють і застосовують команду **Таблиця** ⇒ **Преобразувати** ⇒ **Текст в таблицю**.

Розглянемо основні дії, визначені над таблицями та їхніми елементами.

Щоб ввести дані в клітинку, треба спочатку клацнути на ній мишею (встановити курсор в клітинці).

Щоб переміщатися по таблиці, натискають на клавіші **Tab** (вперед), **Shift + Tab** (назад) чи на клавіші зі стрілками (клавіші управління курсором).

Щоб змінити розташування обмежувальних ліній, їх перетягують за допомогою миші. Можна вставляти чи вилучати рядки і стовпці. Клітинки можна об'єднати, виокремивши їх, застосувавши команду **Об'єднати клітинки** чи натиснувши на відповідну кнопку на панелі .

Дані в клітинках (чи у всій таблиці) *вирівнюють вертикально* відносно верхньої чи нижньої межі або по середині; а також, як звичайно, *горизонтально* по відношенні до лівого краю, правого, центру (*контекстне меню* – Вирівнювання в клітинці...).

Вигляд таблиці можна змінити засобами команди **Границя и заливка...** з пункту **Формат**. У відповідному діалоговому вікні є три закладки **Границя**, **Страница** і **Заливка**, які дають змогу зробити таке:

- вибрати тип рамки; тип, ширину і колір ліній; забрати окремі лінії.
- сторінку з таблицею взяти в рамку-рисунок.
- клітинки затінити чи замалювати їх кольором і прикрасити деяким візерунком.

Якщо таблиця містить числову інформацію, наприклад, про діяльність

фірм,
рекламу
тощо, то її
дані можна
оформити
також у
вигляді



Рис. 7 Стовпчикова діаграма.

діаграми. Щоб вставити відповідну до таблиці діаграму у текстовий документ, її треба виокремити (всю або частину) і виконати послідовність команд: **Вставка** ⇒ **Рисунок** ⇒ **Діаграма**. Через деякий час на екрані з'явиться таблиця в форматі Excel, а також відповідна діаграма (стовпчикового типу).

У цей момент ще можна робити зміни в таблиці — діаграма змінюватиметься автоматично. Тип діаграми можна вибирати з запропонованого меню, клацнувши на кнопці типу. А можна нічого не змінювати, закрити вікно таблиці і клацнути на чистому полі десь у документі. Через мить стовпчикова діаграма (рис. 7) буде там. Тепер можна змінити розміри діаграми чи її розташування.

Діаграма — це об'єкт, який активізують (вибирають, виокремлюють) клацанням мишкою один раз у його межах. Активний об'єкт буде оточений маркерами (маленькими прямокутниками), перетягуючи які можна змінювати розміри об'єкта. Щоб перемістити вибраний об'єкт, курсор миші поміщають над об'єктом, затискають ліву клавішу і перетягують об'єкт на нове місце.

Задача 1

Вибори президентів, ректорів вузів, депутатів, лідерів партій і молодіжних організацій, старост груп тощо супроводжуються опитуванням про шанси претендентів.

Створити таблицю 1 з назвою Шанси претендентів, де навести дані про шанси декількох претендентів (зі 100%) станом на вчора і сьогодні.

Прізвища чотирьох претендентів ввести довільні, наприклад, своїх друзів. Числові дані будь-які, але сума в рядку повинна дорівнювати 100.

Така таблиця буде містити п'ять стовпців з такими заголовками: Прізвища, конкретне прізвище 1, прізвище 2, прізвище 3, прізвище 4; і три рядки, останні два починаються з назв рядів даних: Вчора, Сьогодні. Створити діаграму за заданою таблицею.

Задача 2

Деяка фірма планує рекламний розпродаж трьох видів товарів і сповіщає про це у рекламному повідомленні.

Створити таблицю 2 з назвою Наші нові ціни, де буде чотири стовпці з заголовками: Наші товари, Назва 1, Назва 2, Назва 3; і два ряди даних, що мають назви Сьогодні, Завтра. Побудувати діаграму для створеної таблиці. Назви товарів і ціни ввести довільні, наприклад, так:

Наші товари	Пепсі	Фанта	Спрайт
Сьогодні	2,30	2,50	2,40
Завтра	2,00	2,00	2,00

Задача 3

Створити таблицю 3 з назвою Мої друзі, де є шість стовпців і деяка (8-10) кількість рядків з даними про ваших друзів. Заголовки стовпців: Прізвище, Ім'я, Вік, Місто, Адреса, Телефон. Дані ввести довільні. Оформіть таблицю якнайкраще.

Окрім звичайного тексту, рисунків, автофігур, таблиць і діаграм, текстовий документ може містити ще й інші об'єкти: картинки, фотографії, гіперпосилання, текстові ефекти, графічні та мультимедійні елементи тощо.

Розглянемо детальніше поняття об'єкта. Об'єкт займає один рядок у документі. Якщо він неширокий, то поруч варто розміщати інший об'єкт або вводити текст. Такий текст має обтікати (обгортати) об'єкт, як це можна побачити в газетах,

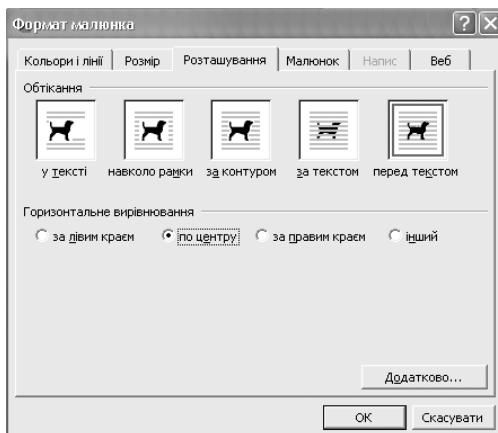


Рис.8 Стили обтікання об'єкта текстом.

журналах чи книжках.

Створені в документі різноманітні тексти і таблиці є *елементами* документа, а не об'єктами. Переміщати їх можна лише через буфер обміну.

Будь-який об'єкт має рамку, її можна робити кольоровою або невидимою. Об'єкт — це автономний елемент документа. Його можна переміщати, змінювати розміри (мишею), обгортати текстом тощо.

Є декілька стилів обтікання об'єкта текстом (рис.8). їх вибирають з меню **Формат** ⇒ **Об'єкт** ⇒ Закладка **Положення**: обтікання. За допомогою інших закладок, які тут є, задають *кольори ліній, замальовування робочого поля* тощо.

Командою **Вставка** у документ можна вставляти також **Номера страниц**, інший документ з деякого файлу (**Файл...**), поточну дату і час (**Дата и время**), примітки (**Примечание**), закладки (**Закладка**), художню букву, математичні формули та всі інші об'єкти, доступні для цього редактора у цій операційній системі.

Розглянемо, як зображення на екрані можна занести в документ як об'єкт. Якщо натиснути на клавішу **Print Screen**, вигляд екрана перенесеться в буфер обміну. потім необхідно запустити графічний редактор **Paint** і вставити вміст буфера у графічний документ. Доопрацьовуємо зображення екрана як картинку (якщо це нам потрібно) і зберігаємо його з розширенням *.bmp*. Потім лише потрібно повернутися у текстовий документ і вставити графічний файл.

Якщо розгорнути газету чи журнал, то звертається увага на оформлення тексту в рекламних оголошеннях. Їх виготовляють за допомогою редактора текстових спецефектів WordArt.



Рис.9 Панель інструментів WordArt.

Редактор спецефектів WordArt працює в двох режимах:

- створення ефекту вручну засобами меню і панелі

інструментів (рис.9);

- використання еталонних зразків ефектів (рис.10).

Вручну за допомогою меню або кнопок панелі інструментів можна підібрати таке:

- ♦ шрифт, його розмір, стиль написання;
- ♦ тип та колір ліній (Border);
- ♦ кольори букв;

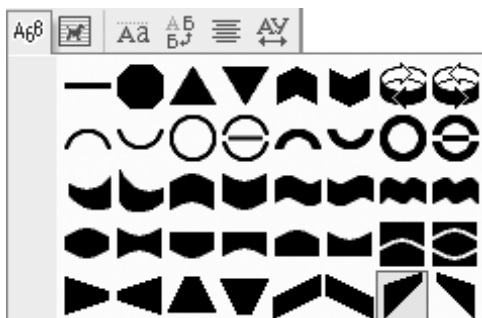


Рис.10 Еталонні зразки оформлення слів у WordArt.

- ♦ колір заливки, а також візерунки або рисунки, які можна використовувати як заливку;
- ♦ повороти та форму накреслення тексту тощо.

Гіперпосилання — це засіб для налагодження зв'язку між документами. Якщо у файл вставити гіперпосилання на інші файли, то можна мати доступ відразу до двох і більше файлів, не вставляючи їх один в одного. Гіперпосилання — це виконаний іншим кольором об'єкт (підкреслений текст чи картинка), який містить адресу іншого файлу. Клацнувши раз на гіперпосиланні, можна відкрити потрібний файл.

Хід роботи

Таблиці будуватимемо трьома різними способами.

1. Завантажте текстовий редактор.
2. У новому текстовому документі введіть заголовок таблиці: *Таблиця 1. Шанси претендентів*. Відцентруйте текст у рядку і перейдіть на новий рядок.
3. Вставте таблицю.

Таблиця ⇒ **Вставити** ⇒ **Таблицю...** ⇒ Задайте кількість стовпців і рядків ⇒ **ОК**.

4. Введіть дані у таблицю і відцентруйте її.

Ввівши дані, виокремте усю таблицю і натисніть кнопку По центру 

5. Вставте діаграму. Виберіть усю таблицю (але без заголовка) і виконайте такі команди: **Вставка** ⇒ **Рисунок** ⇒ **Діаграма**. Закрийте вікно електронної таблиці. Якщо потрібно, змініть розміри чи перемістіть діаграму.

6. Введіть заголовок таблиці: *Таблиця 2. Наші нові ціни*. Нарисуйте таблицю з потрібною кількістю стовпців і рядків. Тип лінії — подвійна, товщина — 1 мм, колір — зелений.

7. Введіть дані і відцентруйте їх. Вставте діаграму для таблиці 2.

8. Введіть заголовок третьої таблиці: *Таблиця 3. Мої друзі*. Перейдіть на новий рядок.

9. Введіть текст таблиці 3. Текст треба вводити, наприклад, так:

Прізвище, Ім'я, Вік, Місто, Адреса, Телефон
Гнатик, Богдан, 16, Чернівці, Головна 66/22, 543455
Фаст, Сашко, 17, Львів, Джерельна 23/76, 334455
Снігур, Ігор, 15, Нью-Йорк, Бродвей 34567, 88667744

10. Перетворіть текст у таблицю. Виокремте текст і виконайте відповідну команду.

11. Заголовки стовпців затініть.

Виокремте перший рядок ⇒ **Формат** ⇒ **Границы и заливка...** ⇒ Перейдіть на закладку **Заливка** ⇒ Виберіть тінь 10% (список **Тип**) ⇒ **ОК**.

12. Клітинки з даними зробіть жовтими.

Виокремте рядки з даними ⇒ **Формат** ⇒ **Границы и заливка..** ⇒ Закладку **Заливка** ⇒ Виберіть колір *жовтий* ⇒ **ОК**.

Проекспериментуйте з *візерунком* (дії аналогічні до попередніх, лише вибираємо додатково тип візерунку).

13. Змініть тип ліній на *подвійний* товщиною 1,5 пт.

Виокремте рядки ⇒ **Формат** ⇒ **Границы и заливка..** ⇒ Закладку **Граница** ⇒ Виберіть тип *подвійний*, ширина 1,5 пт ⇒ **ОК**.

14. Сторінку обведіть рамкою-рисунком — червоні яблука.

Формат ⇒ **Границы и заливка...** ⇒ Закладку **Страница** ⇒ Виберіть з меню **Рисунок: малюнок з яблуками** ⇒ **ОК**.

15. Упорядкуйте дані в таблиці 3 так, що прізвища були за алфавітом.

Виокремте дані в таблиці (всі, окрім першого рядка) і виконайте дії:

Таблиця ⇒ **Сортировка...** ⇒ У діалоговому вікні виберіть номер стовпця (Спочатку за: *стовпцем 1*) і режим **По возрастанию** (або натисніть на кнопку **Сортировка по возрастанию**).

16. Упорядкуйте дані в таблиці 3 за віковим старшинством осіб.

17. Збережіть документ у файлі з назвою *Прізвище5.doc*.

18. Створіть новий текстовий файл.

19. Введіть такий текст: *Рекламна сторінка фірми*.

20. Перейдіть на наступний рядок, де введете назву фірми:

Вставка ⇒ **Рисунок** ⇒ **Объект WordArt** ⇒ Виберіть зразок у вікні Колекція WordArt ⇒ **ОК** ⇒ У діалоговому вікні Зміна тексту WordArt в текстовому полі Текст: введіть назву будь-якої фірми ⇒ Проекспериментуйте зі *шрифтом* і *розміром* символів ⇒ Натисніть на кнопку **ОК**.

21. Заберіть (вирижте) перше речення в буфер обміну. Створіть спецефект WordArt над першим реченням: **Вставка** ⇒ **Рисунок** ⇒ **Объект WordArt** ⇒ Виберіть зі списку зразок ⇒ У текстове поле вставте текст з буфера обміну ⇒ Натисніть на кнопку **ОК**.

22. Змініть розміри і розташування спецефектів. По черзі клацайте у межах об'єктів (з першим реченням і назвою фірми) та змінюйте їх розміри до бажаного формату, перетягуючи мишкою маркери (чорні прямокутники), що їх оточують.

23. Вставте ще один довільний об'єкт WordArt. Розташуйте об'єкти на сторінці якнайкраще.

24. Перейдіть на наступну сторінку (**Ctrl + Enter**) та вставте картинку з галереї Clip Gallery.

Вставка ⇒ **Рисунок** ⇒ **Картинки...** ⇒ **Найти** ⇒ *Виберіть картинку* (клікнувши на ній один раз). Змініть розміри картини, перетягуючи маркери, що її оточують.

25. Вставте в документ ще одну картинку з галереї Clip Gallery. (обов'язково з білим фоном). Змініть фон з білого на градієнтну заливку: **Формат рисунка** ⇒ закладка **Цвета и линии** ⇒ **Заливка: колір: способи заливки** ⇒

закладка **Градиентная** ⇒ **Цвета:** два кольори (чорний та голубий) + **Тип штриховки:** *горизонтальна* (перший варіант) ⇒ **ОК**.

26. Перейдіть на наступну сторінку. Підготуйте місце для введення тексту. Задайте найзручніший для вас режим вигляду документа *Звичайний* або режим *Розмітки сторінки*.

27. Вставте текст з іншого текстового документа.

Вставка ⇒ **Файл** ⇒ Виберіть потрібний файл: *Прізвище3* ⇒ **Вставити**.

Примітка: якщо даний файл не зберігся, то створіть новий, введіть в нього невеликий довільний текст і збережіть під назвою *Прізвище3*, і вже з нього зробіть вставку.

28. Розмістіть текстовий курсор в середині тексту та вставте в дане місце довільну картинку з галереї Clip Gallery.

29. Задайте режим обтікання об'єкта текстом *За контуром* і розташуйте оптимально картинку у тексті. Активізуйте контекстне меню для малюнка ⇒ **Формат рисунка...** ⇒ Закладка **Положение** ⇒ Виберіть з меню потрібний стиль обтікання *За контуром* ⇒ **ОК**.

30. Вставте у документ рисунок з деякого графічного файлу. Пошукайте у списку файлів графічні – файли з розширенням *.bmp, .jpg, .gif*. **Вставка** ⇒ **Рисунок** ⇒ **из файла...** ⇒ Виберіть зі списку файлів довільний графічний файл ⇒ **Вставити**. Задайте обтікання текстом *За текстом + по центру*.

31. Перейдіть на наступну сторінку та введіть такий текст: *Додаткову інформацію про діяльність нашої фірми можете прочитати тут*.

32. Вставте після слова *тут* (в цьому місці повинен стояти текстовий курсор) гіперпосилання на текстовий документ з назвою *Прізвище3.doc*, чи на будь-який інший текстовий документ, який є в поточній папці.

Вставка ⇒ **Гиперссылка** ⇒ **Связать с файлом, веб-стронк...** ⇒ **Папка:** *поточна папка* ⇒ Знайдіть і виберіть файл у списку ⇒ **ОК**.

33. Перевірте, чи спрацьовує гіперпосилання. Наведіть курсор на гіперпосилання (підкреслений текст: шлях до файлу) та клацніть на ньому, затиснувши клавішу **Ctrl**. Перегляньте документ, що відкрився. Закрийте вікно документа.

34. Вставте нумерацію сторінок.

35. Збережіть документ з назвою *Прізвище6*.

Лабораторна робота № 10. Excel. Створення документа.

Методичні рекомендації

Microsoft Excel – таблицний редактор, програма для створення й обробки електронних таблиць. Ярлик Excel найчастіше має вигляд:



Excel дозволяє працювати з таблицями в двох режимах:

- ◆ Обычный – найбільш зручний для виконання більшості операцій.

◆ Разметка страницы – зручний для остаточного форматування таблиці перед друкуванням. Межі поміж сторінками у цьому режимі відображаються пунктирними лініями. Межі таблиці – суцільною лінією, пересуваючи яку змінюють розмір таблиці.

Для переходу поміж режимами Обычный і Разметка страницы використовуються відповідні пункти у меню Вид.

Під панелями інструментів Excel за звичаєм знаходиться *строка формул*, а у нижній частині вікна *строка стану* (див. рис. 11). Щоб вивести або забрати ці рядки слід у меню Вид вибирати відповідні пункти: Строка формул або Строка состояния.

Файл у Excel називається *робочою книгою*. Робоча книга складається з робочих аркушів, імена яких (Лист1, Лист2, ...) виведені на

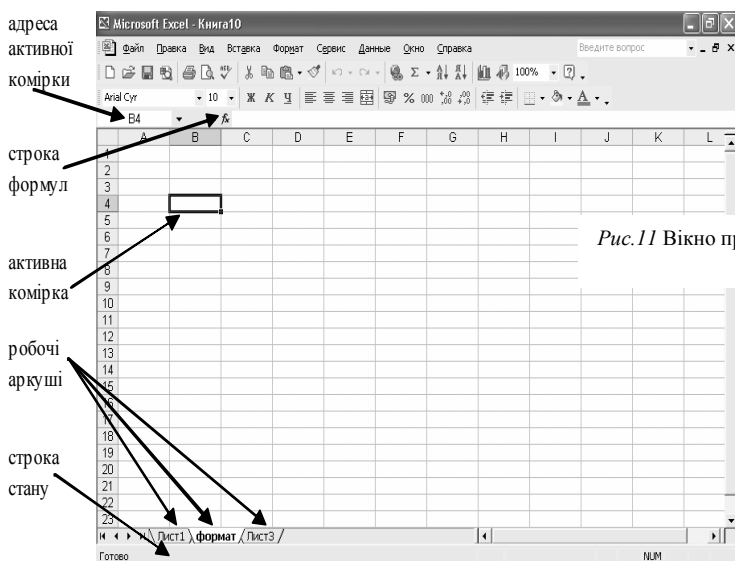


Рис.11 Вікно програми Excel.

ярликах у нижній частині вікна робочої книги (рис.11). Клікаючи по ярликах, можна переходити від аркуша до аркуша всередині робочої книги.

Робочий аркуш являє собою таблицю, що складається з 256 стовпців і 65536 рядків. Рядки позначаються цифрами, а стовпці – латинськими літерами А, В, С, ... (або – рядки та стовпці позначаються цифрами). Кожна клітинка таблиці має *адресу*, що складається з *імені рядку* й *імені стовпця*. Наприклад, якщо комірка знаходиться на перетині другого стовпця (В) і другого рядка (2), то вона має адресу В2.

Одна з комірок таблиці завжди є *активною*, активна комірка виділяється рамкою. Щоб зробити комірку активною, потрібно клікнути на ній мишею.

Для виділення декількох суміжних комірок необхідно встановити покажчик миші на одному з вічок, натиснути ліву кнопку миші і, не відпускаючи її, розтягнути виділення на всю ділянку. Для виділення декількох несуміжних груп комірок слід виділити одну групу, натиснути клавішу **Ctrl** і, не відпускаючи її, виділити інші комірки.

Щоб виділити цілий *стовпець* або *рядок* таблиці, необхідно натиснути мишею на його імені. Для виділення декількох *аркушів* необхідно натиснути **Ctrl** і, не відпускаючи її, натискати на ярликах аркушів.

Для введення даних у комірку необхідно зробити її активною (клікнути на ній) і ввести дані з клавіатури. Дані з'являться у комірці та у строці формул. Для завершення введення слід натиснути **Enter** або одну з клавіш керування курсором. Процес введення даних закінчиться й активною буде сусідня комірка.

Увага! Перед виконання будь-якої команди Microsoft Excel слід завершити роботу з коміркою, тобто вийти з режиму введення або редагування.

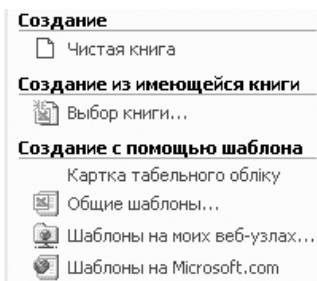
Для створення *нової робочої книги* у меню **Файл** вибрати команду **Создать..** У діалоговому вікні, що розкрилося (рис.12), вибрати шаблон, на основі якого буде створено робочу книгу. Звичайні робочі книги створюються на основі шаблону **Чистая Книга**. Для створення робочої книги на основі даного шаблону можна натиснути кнопку  або комбінацію клавіш **Ctrl + N**.

Для відкриття існуючої робочої книги необхідно в меню **Файл** вибрати команду **Открыть...** або натиснути кнопку  ⇒ розкриється діалогове вікно **Открытие документа**. У переліку слід вибрати диск, на якому знаходиться папка з потрібною робочою книгою, папку (подвійним натисканням) і саму книгу. За замовчуванням в переліку виводяться тільки файли з книгами Microsoft Excel, які мають розширення **.xls**.

Для збереження робочої книги необхідно викликати команду **Сохранить** меню **Файл**, **Ctrl + S** або натиснути. При першому збереженні з'являється діалогове вікно **Сохранение документа**. Далі слід вибрати диск та папку, в якій необхідно зберегти книгу. У полі **Тип файла** – формат, в якому буде збережено книгу – *Книга Microsoft Excel*. У полі **Имя файла** потрібно ввести ім'я книги й натиснути кнопку **Сохранить**.

Для закриття книги необхідно вибрати в меню **Файл** команду **Заккрыть** або натиснути кнопку вікна книги.

Рис.12 Створення книги в Excel.



Для завершення роботи з Excel необхідно вибрати **Файл** ⇒ **Выход** або закрити вікно програми .

1. Адресація комірок. Основні дії з комітками

Кожна комірка має свою назву A1, A2, B10 і т.д. Назву комірки називають її *відносною адресою*. Розглядають:

- > діапазон комірок, тобто набір послідовних комірок, який позначають, наприклад, A1:A10 (всі комірки A1,A2,...,A10);
- > об'єднання комірок - A1;C4:B10 (лише вказані комірки);
- > блок комірок - A2.C5 (всі комірки, які утворюють прямокутник з діагоналлю A2 -C5).

Якщо адреса комірки не повинна змінюватись, її вказують із знаком \$. Наприклад, \$A1 - не змінюється назва стовпця, B\$7 - не змінюється номер рядка, \$D\$12 - не змінюється і назва стовпця, і номер рядка. Таку адресу називають *абсолютною*. Абсолютну адресу зручно задавати, присвоюючи комірці ім'я. Присвоїти активній комірці ім'я можна за допомогою команди **Вставка⇒ Ім'я⇒ Присвоить**. Швидким способом присвоєння імені є його введення у поле адреси активної комірки в рядку формул, яке завершують натисканням клавіші **Enter**. Ім'я можна присвоїти і цілому діапазону комірок . Створити ім'я можна, використавши одну із суміжних з діапазоном комірок: для цього виділяють діапазон разом з цією коміркою, виконують команду **Вставка⇒ Ім'я⇒ Создать** і визначають, де саме буде розташоване ім'я - зверху, ліворуч, знизу чи праворуч. Відмовитись від імені можна через вікно **Вставка⇒ Ім'я⇒ Создать**, в якому виділяють надане комірці ім'я і натискають кнопку **Удалить**.

Над даними в активній комірці визначені такі основні дії:

- > заповнення ~ в більшості випадків проводять з клавіатури;
- > виділення - протягують мишею;
- > вилучення - виділяють комірку і натискають клавішу **Del** або "пробіл";
- > копіювання - курсор наводять на нижній край рамки (курсор прийме вигляд стрілки) і, утримуючи клавішу **Ctrl**, тягнуть мишею в потрібне місце, або наводять курсор на правий нижній кут рамки (курсор прийме вигляд знаку "+ ") і тягнуть мишею (ці способи копіювання використовують для сусідніх комірок);
- > переміщення - курсор наводять на нижній край рамки (курсор прийме вигляд стрілки) і тягнуть мишею в потрібне місце;
- > редагування та форматування даних в комірці - не відрізняється від аналогічних дій в редакторі Word;
- > для форматування всієї таблиці часто використовують опцію **Формат⇒Автоформат** або **Формат⇒Границы..**

Для виділення всього аркуша треба натиснути клавіші **Ctrl+A** або активізувати кнопку, розміщену у лівому верхньому куті робочого поля. Щоб виділити рядок чи стовпець, потрібно клацнути мишею по його номеру чи назві. Виділення діапазону та блоку комірок виконують протягуванням

миші, а виділення розрізнених комірок (рядків, стовпців) - клацанням по них мишею при одночасному натисканні клавіші **Ctrl**.

Для копіювання, переміщення та вставляння значень виділеної комірки застосовують такі способи:

- > команди головного меню **Правка⇒Вырезать, Правка⇒Копировать, Правка⇒Вставить;**
- > аналогічні кнопки панелі інструментів;
- > контекстне меню об'єкту.

2. Заповнення таблиці

В комірку можна вводити текст, число та формулу. Перейти з комірки в комірку можна, натиснувши клавіші $\leftarrow$, $\rightarrow$, $\uparrow$, $\downarrow$ на полі клавіатури, або клацнувши мишкою по потрібній комірці.

Щоб ввести текст необхідно зробити комірку активною і набрати потрібний текст з клавіатури. Закінчити введення можна, натиснувши **Enter** або переходом на іншу комірку.

Числа вводяться так само, як текст. Розділовим знаком в дробових числах є кома, або крапка (в залежності від налаштування).

Для кожного числа потрібно вказати його формат. Для цього треба увійти в пункт меню **Формат⇒Ячейки⇒Число**. (рис.13)

Кожне число в таблиці можна подати в різному форматі (із різною кількістю десяткових позицій, незначущих нулів та ін.).

Для зміни формату вмісту комірки необхідно:

1. виділити потрібні комірки;
2. вибрати команду **Ячейки...** з меню **Формат**;
3. вибрати закладку **Число**

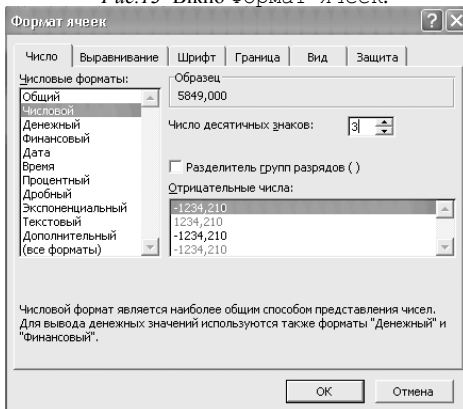
(рис.13);

4. у переліку **Числові формати** вибрати тип формату вмісту клітинок, а в полях праворуч параметри формату;
5. у полі **Образец** буде відображатися приклад вмісту комірки в обраному форматі;
6. натиснути **ОК**.

Ви можете вибрати формати для грошей, дати, часу, відсотка, і т.д. Можна також зменшити або збільшити кількість десяткових знаків числа (розрядність числа).

При роботі з великими числами вводять роздільник тисяч (для цього треба виділити потрібні комірки і клацнути по піктограмі) Всі ці опції винесені на піктограмах. Якщо число надто "довге" і не поміщається в комірку (на екрані видно набір знаків "####"), то комірку слід розширити.

Рис.13 Вікно Формат ячеек.



По замовчуванню текст в комітках вирівнюється по лівому краю комірки, а числа - по правому. При потребі (якщо текст треба розмістити в декількох комітках) декілька комірок можна об'єднати в одну. Відповідна піктограма виводиться на екран через **Сервіс** $\Rightarrow$ **Налаштування** $\Rightarrow$ **Формат**).

Хід роботи

1. Завантажте Excel двома способами.

Мишкою: через ярлик програми на Робочому столі (якщо такий є) або **Пуск** $\Rightarrow$ **Усі програми** $\Rightarrow$ Microsoft Office $\Rightarrow$ Excel.

2. Створіть три нових книги.

Зробіть це трьома різними способами: комбінацією клавіш (**Ctrl + N**), через меню (**Файл** $\Rightarrow$ **Создать...**) та за допомогою кнопки на панелі **Стандартна** .

3. Перейдіть до першої книги за допомогою пункту меню **Окно**.

Виконайте команди **Окно** $\Rightarrow$ **3 Книга1**.

4. У створеній Вами книзі додайте новий листок.

Вставка $\Rightarrow$ **Лист**.

5. Помістіть створений листок на останню позицію.

Захопіть мишкою закладку **Лист4** (див. нижню частину екрана, безпосередньо над рядком стану) і не відпускаючи клавішу перетягніть курсор у потрібне місце.

6. Видаліть **Лист3**.

«Клікніть» на відповідній закладці $\Rightarrow$ **Правка** $\Rightarrow$ **Удалить лист** або через контекстне меню.

7. Відкрийте **Лист1**. В комірку A1 введіть слово *Університет*. В комірку B1 – цифру 5.

Зауважте, як розмістилися введені дані в комітках.

	A	$\leftrightarrow$	B
1			
2			

8. За допомогою миші змініть розміри комірок.

Захопивши межу між заголовками (номерами) відповідних стовпців, перетягуванням розширюємо комірку A1, а комірку B1 звужуємо.

9. Вміст комірки B1 змініть на цифру 9, а в комірку A1 додайте 7, використовуючи строку формул.

Вибираємо комірку B1 (один раз «клікаємо» лівою кнопкою миші) і на клавіатурі натискаємо 9. Вміст комірки змінився з 5 на 9. Вибираємо комірку A1 (подвійне натискання), встановлюємо текстовий курсор після слова *Університет*. З клавіатури вводимо цифру 7. Вміст комірки тепер виглядає *Університет7*.

10. Видаліть вміст комірок A1 та B1. Вибираємо комірки $\Rightarrow$ Delete (Del).

11. В комірки A1 $\div$ G1 введіть відповідно числа 7, 5, 12, 36, 78, 35, 44.

12. Виділіть комірки B1, D1, F1 та G1. Утримуйте затиснутою клавішу

Ctrl. Скопіюйте вміст даних комірок будь-яким відомим Вам способом.

13. Вставте вміст буферу в комірки C7, D7, E7, F7. Виділіть ці комірки ⇒

Правка ⇒ Вставить.

14. Виділіть комірки A9, A10, A11, A12 та знову виконайте **Вставить**.

15. Вирижте вміст комірок A9 та B9 ⇒ вставте в E12 і F12. Видаліть вміст комірок C10 і C12.

16. Перейдіть на Лист2. Переименуйте його: *формат*.

Виконайте подвійне натискання на ярлику Лист2 і введіть нове ім'я – формат.

17. Створіть таблицю:

A1 – *Перша таблиця в лаб. роб. №10*

A2 – *коледж університету*

A4, A5, A6 – *Коледж ЧНУ, Педагогічний коледж, Медичний коледж*

B2 – *к-сть студентів*

B4, B5, B6 – *225, 230, 415*

C2 – *термін навчання*

C3 – *денна форма*

D3 – *заочна форма*

C4, C5, C6 – *3 роки*

D4, D5, D6 – *3,5 роки*

18. Відформатуємо таблицю:

Об'єднаємо комірки A1, B1, C1, D1 – виділяємо їх ⇒ натискаємо на панелі інструментів форматування .

Аналогічно об'єднуємо комірки:

A2 + A3, B2 + B3, C2 + D2

Виділяємо комірки C3 і D3 ⇒ **формат ⇒ Ячейки... ⇒ Закладка Вывравнивание ⇒ Ориентация: 90 градусов**. Аналогічно робимо для A3 і B3.

Виділяємо A4, A5, A6 ⇒ **формат ⇒ Ячейки... ⇒ Вывравнивание ⇒ Отображение:** галочка біля *Переносит по словам*.

Виділяємо всю таблицю ⇒ **формат ⇒ Ячейки... ⇒ Закладка Вывравнивание ⇒ По горизонтали і По вертикали: по центру**.

19. Перейдіть на створений Вами Лист4. Переименуйте його на *журнал відвідувань*.

Клікніть на закладці Лист4 ⇒ **формат ⇒ Лист ⇒ Переименовать** (або через контекстне меню, команда **Переименовать**) ⇒ *журнал відвідувань*.

20. Створіть на даному листі „Журнал відвідувань”.

№ п/п	Прізвище	січень	лютий	березень	квітень	Причина
1	Іванов І.І.	н				пп

2	Петров П.П.			н	н	хв
3	Сидоров С.С.		н			пп
4	Федоров І.І.				н	без пп
5	Петрощук О.О.	н	н			хв

Колір фону: **Формат** $\Rightarrow$ **Ячейки** $\Rightarrow$ **Вид** $\Rightarrow$ **Заливка ячеек**: Цвет, Узор.

Межі таблиці: **Формат** $\Rightarrow$ **Ячейки** $\Rightarrow$ **Граница** $\Rightarrow$ Тип линии.

21. Збережіть файл під назвою *Журнал відвідувань*.

Зробіть це одним з трьох способів: комбінацією клавіш (**Ctrl + S**), через меню (**Файл** $\Rightarrow$ **Сохранить**) або за допомогою кнопки панелі **Стандартна**.

22. Закрийте файл не закриваючи при цьому вікно самої програми Excel.

23. Продемонструйте кожен **Лист** з таблицями викладачеві.

Відкрийте створений Вами документ через список останніх файлів: **Файл** $\Rightarrow$ **1 Журнал відвідувань**.

Лабораторна робота № 11. Excel. Форматування та обчислення.

Методичні рекомендації

Обчислення в таблицях виконуються за допомогою формул. Формула може складатися з математичних операторів, значень, посилань на вічко й імена функцій. *Результатом виконання формули* є деяке нове значення, що міститься в комірці, де знаходиться формула. Формула починається зі знаку рівності "=".

Приклади формул:
$$=(A4 + B8) * C6 \quad =F7 * C14 + B12$$

У формулі можна використовувати числа, відносні та абсолютні адреси комірок, імена комірок, а також назви функцій. Наприклад:

$$=3+5, \quad -A\$1+A5, \quad =СУММ(A1:A10), \quad =КОРЕНЬ(H4).$$

Введення формули закінчують натисканням клавіші **Enter**. В формулах використовують знаки:

- **:** - діапазон;
- **;** - об'єднання;
- **+, -, *, /, ^, <, >, =** - знаки математичних дій. Порядок обчислень визначається звичайними математичними законами.

Потрібні у формулах функції можна знайти за допомогою **Мастера функцій**.

Функціями в Microsoft Excel називають об'єднання декількох обчислювальних операцій для розв'язання визначеної задачі. Функції в Excel являють собою *формули, що мають один або декілька аргументів*. В якості аргументів вказуються числові значення або адреси комірок.

Наприклад:

=СУММ(A5:A9) – сума клітинок A5, A6, A7, A8, A9;

=СРЗНАЧ(G4:G6) – середнє значення вічок G4, G5, G6.

Функції можуть входити одна в іншу, наприклад:

=СУММ(F1:F20)ОКРУГЛ(СРЗНАЧ(H4:H8);2);

Функції поділяють на прості та складні. До складних належать функції СЧЕТЕСЛИ, СУММЕСЛИ, ЕСЛИ. У всі ці функції входить умова, яка має вигляд порівняння, наприклад, $A10 > 3$, $B3 = C10$.

Функція СЧЕТЕСЛИ розраховує кількість заповнених комірок у вказаному діапазоні, які задовольняють поставлену умову, функція СУММЕСЛИ відшукує комірки, вміст яких відповідає умові, а функція ЕСЛИ використовується для перевірки умови стосовно значень та формул і вибирає одне із заданих в залежності від виконання чи невиконання умови.

У самій комірці міститься результат обчислення згідно формули, а формула видна лише в стрічці формул (це можна змінити через вікно **Сервіс**⇒**Параметри**).

Строка формул (*рядок формул*) має наступний вигляд:

Поле імені в рядку формул показує адресу чи ім'я активної комірки, а **поле формул** відображає вміст цієї комірки, або дію, яку в ній виконують. Знаки в рядку формул відповідно означають "вилучити, відмінити", "внести в комірку, закінчення набору", "змінити формулу". Прапорець у полі імені відкриває перелік функцій, які використовуються найбільш часто, а рядок **Другие функции** відкриває вікно **Мастер функций**. Деякі функції, що часто використовуються при роботі з таблицями, наприклад, сумування, винесені у вигляді піктограм.

Формули можна копіювати з комірки в комірку "протягуванням" мишею (як вже було описано вище). При копіюванні формул відносні адреси комірок змінюються, а абсолютні - ні, тому треба слідкувати за зміною адрес комірок і при потребі використовувати абсолютну адресацію.

Для введення функції у комірку необхідно:

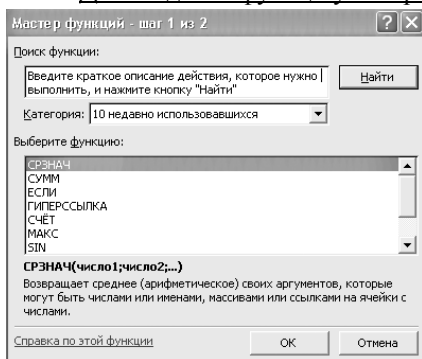


Рис. 14. Вікно Мастер функций.

1. виділити клітинку для формули (один раз клікнути на ній);
2. викликати **Мастер функций** за допомогою команди меню **Вставка** ⇒ **Функция** або кнопки  ;
3. у діалоговому вікні, що з'явилося (рис. 14), вибрати тип функції в переліку **Категория** і потім потрібну функцію в переліку **Выберите функцию:** ;
4. натиснути кнопку **ОК**;
5. у полях **Число1**, **Число2** та ін. наступного вікна (рис.15) ввести

аргументи функції (числові значення або посилання на вічка);

6. щоб не вказувати аргументи з клавіатури, можна натиснути кнопку , яка знаходиться праворуч текстового поля Число1, наприклад, і виділити мишею ділянку комірок, що містять аргументи функції;

7. натиснути **ОК**.

Вставити в комірку функцію суми СУММ можна за допомогою кнопки 

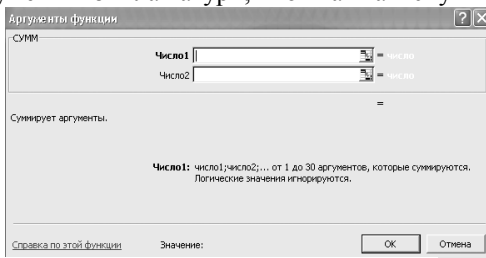


Рис.15 Вікно Аргументы функции.

Хід роботи

1. Завантажте Excel. Працюємо з аркушем Лист1.
2. Об'єднайте шість перших клітинок в першому рядку (A1:F1). В отриману комірку введіть основний заголовок: *Результати виставки-продажу автомобілів Opel по Україні у березні 2007 року*.

Якщо текст в один рядок не поміщається, то задаємо для даної комірки **Переносить по словам...** і при потребі збільшуємо висоту рядка, відцентруємо текст.

3. Перейдіть до комірок другого рядка і для стовпців зробіть такі заголовки – *Місто, Дата, Продано автомобілів, Сума, Покупців, Затрати на переїзд*. При потребі задаємо **Переносить по словам...** і збільшуємо ширину стовпців.

Відцентруємо вміст комірок по горизонталі та вертикалі.

4. У комірки стовпців під заголовком вводимо:

Стовпець А – Місто – вводимо назви 10-и міст;

Стовпець В – Дата – вводимо значення від 1 до 30;

Стовпець С – Продано автомобілів – введіть кількість проданих на Вашу думку автомобілів;

Стовпець D – Сума – введіть суму отриману за продаж автомобілів у кожному місті;

Стовпець Е – Покупців – введіть кількість покупців у кожному місті;

Стовпець F – Затрати на переїзд – введіть суми витрат на перевезення.

При потребі змінюємо ширину стовпців.

5. Копіюємо всю таблицю, відступаємо два пустих рядочки (13-ий та 14-ий), вибираємо комірку A15 і робимо вставку.

Далі працюємо з копією таблиці (друга таблиця), оригінал (перша таблиця) залишається далі без змін.

6. Застосуємо **Автоформат...**

Виділіть всю таблицю та застосуйте **Автоформат: Формат** ⇒ **Автоформат...** ⇒ зі списку виберіть **Классический 3**.

7. Копіюємо другу таблицю, до якої застосували автоформат, знову

відступаємо два порожніх рядочки (27-ий та 28-ий) і в комірку A29 робимо вставку.

Всього на аркуші в нас буде вже три таблиці.

Далі працюємо з третьою таблицею.

8. Застосуємо окремі формати.

Стовпець В – Дата – застосувати формат (виділяємо комірки з числовими значеннями під заголовком $\Rightarrow$ **Формат** $\Rightarrow$ **Ячейки...** $\Rightarrow$ закладка **Число** $\Rightarrow$ вибираємо **все формати** і в поле **тип**: вводимо: #__березня)
(між # та „березня – два нижніх тире).

Стовпець С – Продано автомобілів – формат **Числовой**, без знаків після коми.

Стовпець D – Сума – проставити знак \$ та два знаки після коми (числовий формат **Финансовый**, *обозначение* – \$ Английский (США)).

Стовпець E – Покупців – формат **Общий**.

Стовпець F – Затрати на переїзд – в гривнях та без знаків після коми (формат **Финансовый**).

9. Збережіть файл під назвою Продаж автомобілів.

10. Відкрийте Лист2. В комірки A1÷G1 введіть цифри 1, 12, 23, 34, 45, 56, 67.

Скопіюйте вміст даних комірок. Зробіть дві вставки в комірку A3 та A5 відповідно.

11. Рахуємо суму.

a. Встановлюємо курсор ,наприклад, на комірку I1 $\Rightarrow$ вводимо знак "=" $\Rightarrow$ **Вставка** $\Rightarrow$ **Функция...** $\Rightarrow$ *Выберите функцию*: СУММ $\Rightarrow$ ОК $\Rightarrow$ *Число1*: A1 : G1 $\Rightarrow$ ОК. В комірці H1 отримуємо результат сумування вмісту комірок від A1 до G1.

b. Встановлюємо курсор на комірку, наприклад H4 $\Rightarrow$ вводимо знак "=" $\Rightarrow$ почергово клікаємо (на комірці лівою кнопкою миші, на знак "+" на клавіатурі): A3 $\Rightarrow$ + $\Rightarrow$ B3 $\Rightarrow$ + $\Rightarrow$ C3 $\Rightarrow$ + $\Rightarrow$ D3 $\Rightarrow$ + $\Rightarrow$ E3 $\Rightarrow$ + $\Rightarrow$ F3 $\Rightarrow$ + $\Rightarrow$ G3 $\Rightarrow$ Enter.

c. Мишкою виділяємо комірки від A5 до H5 і на панелі інструментів клікаємо на кнопці .

У всіх трьох комірках значення суми має бути однаковим.

d. Порахуйте суму (будь-яким способом) для комірок C3, E1 та H5. Результат сумування повинен знаходитися у комірці B7.

12. Зробіть розрахунок за формулою:

$$A11 = (B3 + A5) * A3 / 2.$$

Вибираємо комірку A11, встановлюємо знак "=" і вводимо саму формулу (див. рядок формул).

13. Обчислимо середнє значення.

=CPЗНАЧ(A3:C3;E5;G1) або

=CPЗНАЧ(A3;B3;C3;E5;G1)

обчислюємо середнє значення для комірок A3, B3, C3, E5 і G1.Результат в комірці E9.

14. Продемонструйте кожен Лист з таблицями викладачеві.

Лабораторна робота № 12. Робота з матрицями та списками. Побудова діаграм.

Методичні рекомендації

1. Робота з матрицями

Числовою матрицею називають прямокутну таблицю з n стовпців m рядків, складену з чисел. Над матрицями встановлені наступні дії: множення на число, множення на вектор, множення матриць, транспонування, знаходження обернених. Всі ці дії можна виконати за допомогою електронних таблиць.

Створимо матрицю, наприклад, у блоці комірок A2:C4 і розглянемо виконання дій над нею.

1. Щоб помножити матрицю на число k .

- виділяють область для запису результату (виділяють блок комірок);
- у полі формул записують формулу =A2:C4* k ;
- натискають клавіші **Ctrl+Shift+Enter**;
- результат з'явиться у виділеній області.

2. Щоб помножити матрицю на вектор (кількість стовпців першого множника повинна дорівнювати кількості рядків другого):

- виділяють область для запису результату;
- викликають **Мастер функций** та вибирають функцію
- заносять відповідні значення блоків матриці та діапазону вектора;
- встановлюють курсор на рядок формул; натискають клавіші **Ctrl+Shift+Enter**, результат з'явиться у виділеній області. Аналогічно виконують множення матриці на матрицю.

Для знаходження оберненої матриці використовують функцію МОБР, а для транспонування - ТРАНСП. Для обчислення визначника матриці використовують функцію МОПРЕД.

2. Робота зі списками

Для того, щоб легко і зручно виконувати обчислення, проводити дії над даними, необхідно задати певну схему їх розміщення або структуру таблиці. Кількість стовпців в таблиці повинна бути однаковою, дані, які в них розташовані, - одного типу. Рядки повинні містити опис певної предметної області.

В стуктурованій таблиці виконують *сортування даних*, які в неї входять. Для цього використовують пункт меню **Данные⇒Сортировка** або відповідні піктограми. Сортування можливе як для тексту (по алфавіту),

так і для чисел. Сортування можна проводити як по одному стовпцю (рядку), так і по декількох. Це можна робити поступово, задаючи потрібні умови сортування послідовно, або поставити курсор на будь-яку комірку таблиці і відкрити пункт меню **Данные⇒Сортировка**. У вікні **Сортировка диапазона** вказують ключі (умови) сортування. Таблицю, одержану в результаті сортування, можна зберігати, копіювати тощо.

Ще однією важливою операцією є *фільтрація* (вибірка) даних, тобто, відбір даних за певними параметрами. Для вибірки даних потрібно поставити курсор на будь-яку комірку таблиці та викликати пункт меню **Данные⇒Фильтр⇒ Автофильтр**. У результаті у верхньому рядку кожного стовпця з'являться кнопки списків (прапорці). Щоб задати умову фільтрування, треба відкрити прапорець в потрібному стовпці. Якщо потрібної вам немає, відкрийте пункт **Условие**. У вікні **Пользовательский автофильтр** задайте необхідну умову. Ви одержите нову таблицю з відібраними згідно поставлених умов даними. Якщо у відкритому прапорці задати умову **Все**, то таблиця набере попереднього вигляду. Для скасування команди **Автофильтр** треба виконати команди **Данные⇒Фильтр⇒Автофильтр**.

3. Побудова діаграм

Для того щоб створити діаграму, необхідно задати для неї дані. Наприклад, виділити в таблиці комірки з потрібними вам значеннями. Побудова самої діаграми виконується за допомогою піктограми **Мастер диаграмм** або через пункт меню **Вставка**. У вікні, яке відкриється, необхідно вибрати тип діаграми і натиснути кнопку **Далее**. Далі, відкриваючи наступні вікна, уточнити діапазон та розміщення даних, задати назву діаграми та осей, визначити лінії сітки, орієнтацію, наявність та розташування легенди (*легенда* - це текстове пояснення до кожного об'єкта діаграми).

У випадках, коли автоматично встановлені програмою Excel масштаб осей та точка відліку вас не задовольняють, потрібно двічі клацнути мишею по осі Y (або викликати контекстне меню цієї осі) і вибрати пункт **Формат**. В однойменному вікні, вибираючи вкладки, можна задати потрібний масштаб на осі, колір та товщину лінії, параметри шрифту, формати чисел і т.д. Аналогічно форматують вісь X.

Створену діаграму можна переміщати та змінювати її розміри за допомогою мишки. Для редагування існуючої діаграми треба виділити її (клацнувши мишкою), викликати контекстне меню й активізувати потрібну команду. Редагування можна провести і за допомогою панелі **Диаграммы**, яка відкривається командою **Вид⇒Панели инструментов⇒ Диаграмма**.

Завдання

Варіант 1.

Переведіть грошові вклади 10 громадян в іноземну валюту згідно курсу: 1грн. - 5,25\$; 1грн. - 6,20 єуро ; 1грн. - 0,86 zł

Рекомендації

1. Зробіть комірки B2, C2, D2 іменованими.
2. Виконайте обчислення.
3. Зробіть ті ж обчислення, використовуючи абсолютні адреси комірок.
4. Побудуйте діаграму.

	A	B	C	D	E
1		\$	єуро	zł	
2		5.25	6.2	0.86	
3					
4	Прізвище	Сума в грн.	Сума в \$	Сума в єуро	Сума в zł
5	Коваль А.	2345.00			
6	Андрусів С.	13345.00			
7	Микитів О.	7890.00			
8	Шевчук Т.	45682.00			
9	Рибак Г.	3.678.00			
10	Чміль М.	4891.00			
11	Чабан К.	34567.00			
12	Кучмук О.	2789.00			
13	Миклаш П.	6789.00			
14	Стеців А.	9783.00			
15					

Варіант 2

Створіть таблицю згідно з поданим взірцем та порахуйте середній бал успішності учнів засобами табличного редактора. Виведіть значення середнього балу у вигляді діаграми.

	A	B	C	D	E	F	G
1	Список	Історія	Математика	Укр.мова	Укр.літ.	фізика	Сер.бал
2	Коваль А.І	8	4	5	8	10	
3	Андрусів С.	8	8	6	10	3	
4	Микитів О.	8	6	4	7	8	
5	Шевчук Т.	6	7	9	4	6	
6	Рибак Г.	10	10	6	6	9	
7	Чміль М.	4	5	4	7	4	
8	Чабан К.	8	8	7	7	8	

9	Кучмук О.	8	6	7	7	5	
10	Сер.бал						

Варіант 3

Нарахування зарплати викладачам проводиться погодинно згідно з їх кваліфікацією та стажем. Якщо стаж перевищує десять років, то викладач одержує доплату в розмірі 20% від нарахованої зарплати. Крім того, за класне керівництво доплачують 30 грн.(в таблиці позначено 1 - є класне керівництво, 0 - немає класного керівництва). З нарахованої суми знімають 15% податків. Обчисліть зарплату викладачів згідно приведених даних: Побудуйте діаграму.

	А	В	С	Д	Е
1	Прізвище	Педнавантаження (год.)	Оплата за годину (грн.)	Стаж роботи	Класне керівництво
2	Коваль А.М.	860	1,2	5	1
3	Яцишин М.К	920	1.6	12	0
4	Павлюк С.Т.	760	1,4	7	0
5	Шишка Л.І.	1020	1,8	20	0
6	Дуб Р.І.	836	1.9	25	1
7	Маковійчук М.О.	720	1,2	3	1
8	Марків Г.Ф.	858	1,6	13	0

Лабораторна робота № 13. Проектування бази даних у середовищі СУБД Access. Створення таблиць БД.

Методичні рекомендації

Для збереження інформації в базі даних (БД) користуються кількома таблицями, а для обробки даних створюють форми, запити, звіти тощо. Перед створенням таблиці необхідно уявляти її структуру - загальний вигляд таблиці (кількість та зміст полів таблиці). Перед роботою в Access таблиці краще зобразити на папері, це зменшить кількість виправлень у готовій таблиці. При цьому необхідно дотримуватися правил: не повторювати інформацію як у межах однієї таблиці, так і між таблицями, кожену таблицю створювати для інформації тільки на одну тему. На цьому етапі вирішують, які об'єкти будуть у БД, які між ними необхідно створити зв'язки, які поля об'явити ключовими.

Якщо за усіма даними скласти одну таблицю, вона буде перенасичена відомостями і не зручна для перегляду. Тому ці відомості краще розбити на кілька таблиць, а між ними засобами СУБД встановити зв'язки. Для зручності складемо таблицю на папері, з неї буде видно, які таблиці-об'єкти входить до БД «Україна», складові цих об'єктів та дії над ними (табл. 1).

Макет подання інформації до БД

Назва таблиці	Складові	Дії користувача щодо обробки даних
Області	Область(назва)	1
	Обл Центр	2
	Населення	3
	Площа	4
	Кількість районів	5
	Густота населення	6
	Кількість санаторіїв	7
Санаторії	Назва	Введення даних у таблицю (форму), перегляд даних, зміна даних
	Лікувальний профіль	
	Кількість місць	
	Початок сезону роботи	
	Кінець сезону роботи	
Номер машин	Область	Введення даних у таблицю (форму), перегляд даних, зміна даних
	НомЗнак	
	ТелКод	

Технологія створення БД

Створення нової бази даних, відкриття існуючої СУБД Access пропонує одразу на початку роботи з пакетом. **Мастер создания БД** допоможе на основі вже існуючих об'єктів готових баз даних створити свою БД з усіма об'єктами. Для самостійного створення своєї БД необхідно вибрати режим **Новая база данных**.

Початок роботи в середовищі СУБД Access

Створення нової БД		Робота з існуючою БД
1	Відкрити Microsoft Access (Пуск\ Программи\ [Microsoft Office] \ Microsoft Access) => на екрані вікно СУБД Access	
2	Вибрати режим <i>Новая база данных</i>	Вибрати режим <i>Открыть базу данных</i>
3	У діалоговому вікні вибрати папку, де буде збережена Ваша БД, та ввести ім'я своєї БД	У діалоговому вікні вибрати папку, де збережена Ваша БД, та за іменем вибрати БД для роботи
Якщо ви знаходитесь у вікні СУБД, створення та відкриття бази даних виконується аналогічними командами меню Файл		

Алгоритм створення та редагування об'єктів БД

	Створення	Редагування
1	У вікні Вашої бази даних у списку <i>Объекты</i> вибрати значок, необхідний для роботи об'єкта	
2	Натиснути кнопку <i>Создать</i> на панелі інструментів вікна програми	Вибрати за назвою об'єкт (виділити його)
3	Із наведеного списку вибрати режим створення об'єкта	Вибрати режим: конструктора, майстра чи режим таблиці

Створення таблиць БД

Головним об'єктом БД є таблиця. Створення нової БД починається зі створення таблиць. Для **створення нової порожньої таблиці** в Access передбачено **кілька режимів**:

Мастер таблиц;

Режим таблиц;

Режим конструктора.

Використання **Мастера таблиц** допоможе вибрати поля з уже наявних таблиць.

У **Режиме таблиц** Access пропонує порожню таблицю з такими іменами полів: «поле1», «поле2»,... До таблиці одразу вводять дані. Здійснюють автоматичну установку типу даних. Користувачу залишається замінити імена полів на свої — зайві поля (не заповнені даними) автоматично зникають.

Під час використання **Режим конструктора** користувач сам спочатку створює структуру таблиці, а потім заповнює її даними.

Структура таблиці — це загальний вигляд таблиці з певною кількістю полів та їхніми характеристиками. У створенні структури таблиці для кожного поля визначають унікальне ім'я, тип даних, якими буде надалі заповнене поле. Для деяких типів вводять розмір поля (ширину в кількості символів). Коли визначають тип даних, необхідно враховувати, які значення будуть уведені в дану колонку (не можна, наприклад, у полі числового формату зберігати текст), скільки місця необхідно для збереження значень поля, які операції виконуватимуть із значеннями (можна порахувати суму над значеннями полів типу «числове» або «грошове»). Зверніть увагу на тип поля «лічильник»: якщо значення іншого типу можуть повторюватися в межах одного поля, «лічильник» є результатом роботи арифметичної прогресії, тому його значення ніколи не дублюються. У полях типів **Числовой, Дата/время, Денежный і Логический** можна визначати формат виведення даних.

Типи даних СУБД Access

Тип даних	Види даних	Максимальний розмір даних
Текстовий	Текст - набір символів	255 символів
МЕМО	Опис	До 8 кбайтів
Числовий (Числовой)	Числа, окрім грошових позначень	8 байтів
Грошовий (Денежный)	Значення валют (використовується для уникання округлень при обчисленнях)	8 байтів: до 15 символів у цілій частині і до 4 - у дробовій
Дата/час (Дата/время)	Дата і час.	8 байтів
Лічильник (Счетчик)	Автоматична вставка чисел арифметичної прогресії з кроком 1 у внесенні запису	4 байти
Логічний (Логический)	Одне з двох значень Так/Ні, або True/False, або Вкл/Викл.	1 біт
Об'єкт OLE	Об'єкт, створений іншою програмою за протоколом OLE	До 1 Гбайта
Гіперпосилання (Гиперссылка)	Посилання на адресу	До 8 кбайтів

Таблицю *Області* створимо в режимі **Конструктор**.

Під час вибору режиму **Конструктор** і таблиці для об'єкта Access запропонує на екрані табличну структуру (не плутати з таблицею бази

даних!), кожний рядок якої містить інформацію про одне поле: ім'я, тип даних, властивості поля.

Імена полів містять не більше 64 символів, не можуть починатися з пробілу (краще пробіл взагалі не вставляти), в імені не має бути знаків «.», «!», «'», «[», «]».

Перехід між комірками автоматично зберігає введення характеристик поля у попередню комірку. Перехід здійснюється за допомогою клавіш переміщення курсору, табуляції (праворуч - клавіша Tab, ліворуч — комбінація Shift+Tab) та клавіші **Enter**.

У колонці **Тип даних** можна залишити за замовчуванням **Текстовый** або із списку типів обрати необхідний.

У колонку **Описание** вводять опис даних, які міститимуть поле. За умови додавання нових даних у поле текст опису з'явиться в рядку стану. Заповнювати опис необов'язково.

За необхідності задають значення властивостей поля в нижній частині вікна.

Таблицю зберігають інструментом **Сохранить**: у вікно діалогу необхідно ввести ім'я таблиці, наприклад, «Області» — у вікні вашої БД серед об'єктів таблиці з'явиться назва Вашої таблиці. У таблиці *Області* будуть такі поля:

- поля з іменами «Область», «ОблЦентр» (текстового типу) із довжиною 20 символів;
- поля з іменами «Населення», «Площа» будуть числового типу, з числами — довгими цілими.

Таблицю «Номер машин» створимо у режимі таблиць.

Алгоритм створення структури таблиці в Режиме таблиць

1. Вибрати **Режим таблиць**.

2. Ввести імена полів у таблицю — двічі клацнути на назві поля та ввести нові: *Область* та *НомЗнак*.

У цьому режимі можна одразу вводити дані. Тип поля за введеними даними Access визначає автоматично.

Таблицю «Санаторії» створимо у режимі **Мастер таблиць**.

Алгоритм створення структури таблиці в режимі Мастера таблиць

1. Скористатися діями з табл. 4 конспекту.

2. Вибрати режим **Мастера таблиць**.

3. У вікні **Мастера** у розділі **Таблицы** вибрати таблицю, яка за змістом та полями відповідає бажаній (Виберемо таблицю «Мероприяття»).

4. Із назв полів існуючої таблиці необхідно вибрати поля для нової таблиці:

- якщо кількість полів така сама, вибирають подвійну стрілку (зворотні стрілки дають можливість відмовитися від усіх полів або від виділеного);

- інакше вибирають за допомогою мишки поле та натискають одинарну стрілку (вибрати **НазвМероприяття**) — ім'я поля копіюється в

нову таблицю. Її можна замінити на бажане: наприклад, змінимо на **НазвСанатория**.

Аналогічно створюють наступні поля нової таблиці: **Место** (в це поле будемо вводити назву області, у якій розташований санаторій); **ОписаниеМероприятия** замінимо на **ОписСанатория** або **ЛекПрофиль**; **ИмеетсяМест**, **НачальнаяДата**, **КонечнаяДата** залишаємо без змін.

Звичайно, при виборі полів із таблиці майстра необхідно відстежувати тип поля.

5. Вибір кнопки **Далее** дасть змогу ввести ім'я таблиці.

6. На цьому можна припинити створення структури (кнопка **Готово**) або заповнювати таблицю (**Далее**).

Заповнення таблиць БД

Щоб у порожню таблицю ввести дані, необхідно виділити назву таблиці та вибрати команду **Открыть** або двічі клацнути по назві - на екрані з'явиться порожня таблиця для перегляду, введення та редагування даних.

Нова таблиця складається з одного порожнього запису. Після введення даних порожній запис переміщується в кінець таблиці.

У таблиці активний запис програма маркує трикутником, порожній — зірочкою. Номер записів у порядку зростання вказується автоматично (поле має тип лічильника). Переміщення по таблиці здійснюється такими самими клавішами, що й по структурі таблиці в режимі **Конструктор таблиць**. Перехід до іншого поля або запису є підтвердженням введення даних у попередній об'єкт таблиці.

Хід роботи

Користуючись вказівками конспекту, таблицею 1:

1. створити базу даних «Україна» (табл.2);
2. створити структуру таблиці «Області» (*Область, ОблЦентр, Населення, Площа*) у режимі **Конструктор таблиць**;
3. заповнити таблицю даними для 6 областей;
4. створити й заповнити таблицю «№ машин» (*Область, НомЗнак*);
5. створити й заповнити таблицю «Санаторії» (*НазваСанаторію, ЛікувальнийПрофіль, КількМісць*);
6. відкрити таблицю «Області». Додати до неї ще 6 записів;
7. змінити структуру таблиці «Області», додати поле «Кількість Районів»;

Таблиця 6

«Області» бази даних «Україна»

№п/п	Область	ОблЦентр	Населення	Площа
1	Вінницька	Вінниця	1832500	26500
2	Волинська	Луцьк	1063900	20200
3	Дніпропетровська	Дніпропетровськ	3745000	31900
4	Донецька	Донецьк	5007900	26500
5	Житомирська	Житомир	1445500	29900
6	Закарпатська	Ужгород	1288200	12800

7	Запорізька	Запоріжжя	2023800	27200
8	Івано-Франківська	Івано-Франківськ	1453700	13900
9	Київська	Київ	4476100	28900
10	Кіровоградська	Кіровоград	1183800	24600
11	Крим	Сімферополь	2526400	27000
12	Луганська	Луганськ	2637800	26700
13	Львівська	Львів	2728600	21800
14	Миколаївська	Миколаїв	1309900	24600
15	Одеська	Одеса	2528600	33300
16	Полтавська	Полтава	1693100	28800
17	Рівненська	Рівне	1190300	20100
18	Сумська	Суми	1354500	23800
19	Тернопільська	Тернопіль	1163900	13800
20	Харківська	Харків	2997900	31400
21	Херсонська	Херсон	1237100	28500
22	Хмельницька	Хмельницький	1474000	20600
23	Черкаська	Черкаси	1463700	20900
24	Чернівецька	Чернівці	935400	8100
25	Чернігівська	Чернігів	1302900	31900

Лабораторна робота № 14. Редагування структури таблиці й даних БД. Впорядкування, пошук та фільтрація даних.

Методичні рекомендації

Наступним етапом є **робота у базі даних**. Вона передбачає редагування даних, перегляд, розташування їх в об'єктах БД, вибірковий показ даних із застосуванням фільтрів.

Для пошуку та перегляду даних в об'єкті «таблиця», як і в інших об'єктах, можна скористатися:

- лінійкою прокрутки;
- списком порядкових номерів записів;
- діалоговим вікном пошуку;
- фільтрацією

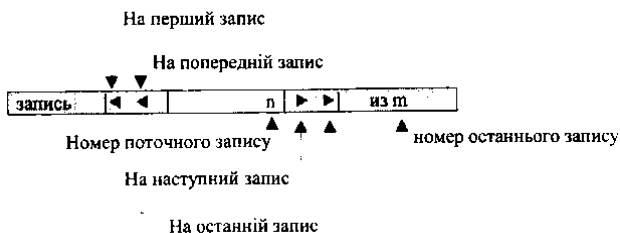
даних.

Вигляд панелі зі списком номерів записів Рис.16

Лінійкою прокрутки, якщо база містить багато записів, користуватися незручно.

Записи можна знайти за їх порядковими номерами в таблиці.

Для цього існує **панель зі списком номерів записів** (у нижній частині таблиці), в яку можна одразу (двічі клацнувши по ній мишкою) ввести



номер запису або змінити номер запису кнопками переходу по записах (рис.16).

За допомогою **вікна пошуку (Правка/ Нйти)** можна знайти конкретні записи або значення в полях. Це вікно дає змогу перейти від запису до запису. За необхідності заміни користуються вікном заміни (робота з цими вікнами виконана в однотипному режимі всіх додатків Microsoft Office).

Якщо таблиця містить велику кількість полів або якщо поля великі за розміром, на екрані з'явиться тільки частина таблиці. Для перегляду решти полів користуються лінійкою прокрутки. Але іноді необхідно, щоб деякі поля увесь час були в межах екрана, тоді необхідно ці **поля закріпити**: встановити курсор у будь-якому місці поля та скористатися командою **Формат/Закрепить столбцы**. Команда **Формат/Освободить столбцы** знімає закріплення.

Редагування вигляду таблиці

Для зміни структури таблиці необхідно скористатися **Режимом конструктора**.

Робота з полями в Режиме конструктора

1. Щоб додати поле між існуючими полями, необхідно встановити курсор на рядку з назвою поля, перед яким буде вставка, вибрати інструмент вставки рядка (табл. 2) або скористатися командою **Вставка/ Строка**. Щоб додати поле в кінець таблиці, необхідно вибрати перший порожній рядок та ввести назву.

2. Для знищення поля (або кількох полів) його спочатку виділяють, користуючись командою **Правка/ Удалить**.

3. Для переміщення одного або кількох полів виділити ці поля, клацнути мишкою на вказівник поля і, не відпускаючи мишки, перетягнути лінійку, яка з'явилася над верхнім виділеним полем, на рядок, перед яким буде вставлено виділені поля.

4. Для зміни імені поля двічі клацнути на імені мишкою та ввести нове ім'я.

5. Для зміни типу поля перейти в колонку типів даного поля та скористатися списком типів. Заміна типу поля вже заповненої таблиці може призвести до знищення даних.

6. Для зміни ширини поля необхідно змінити значення у властивостях поля.

Робота з відкритою таблицею даних

Коли відкрито таблицю з даними, є можливість редагувати як вигляд таблиці (тобто її поля), так і записи - дані. Ця робота аналогічна до роботи в режимі конструктора та роботи з електронною таблицею.

Для додавання записів користуються командами **меню Вставка: Столбец або Новая запись**. Для заміни імен полів у меню **Формат** є команда **Переименовать столбцы** (чи двічі клікають на імені і вводять нове ім'я). Послідовне вилучення полів здійснюється командою **Правка/**

Удалить, а записів - **Правка/ Удалить записи**. Розмір поля змінюється за допомогою мишки, наведеної на межу полів.

Таблиця 2

Інструменти СУБД Access

Інструменти	Призначення
	Додавання та вилучення полів таблиці.
	Впорядкування за зростанням та зменшенням
	Фільтр за виділенням; змінити фільтр; застосувати фільтр
	Встановити або зняти статус ключового поля
	Відкрити вікно схеми даних
	Активізувати вікно додавання таблиці в схему даних
	Кнопка запуску для попередньої перевірки роботи запиту

Впорядкування даних у таблиці - це розташування записів у прямому чи зворотному алфавітному порядку значень одного з полів таблиці. Говорять про впорядкування за зростанням або спаданням. Для впорядкування достатньо виділити колонку з необхідною назвою та вибрати значок упорядкування або **Записи/Сортировка/ Сортировка по возрастанию (Сортировка по убыванию)**.

Для пошуку, перегляду вибірових записів, значення яких відповідають визначеній умові, використовують **фільтри**. Фільтр дає змогу уникнути тих записів, які нецікаві на даному сеансі перегляду. За допомогою Access можна встановлювати такі фільтри:

- простий фільтр;
- фільтр за виділенням фрагментом;
- розширений фільтр.

Усі ці засоби відображені у команді **Записи/Фільтр**. Команда в **Записи/Применить фильтр** одразу активує попередньо обраний фільтр. Якщо необхідний інший засіб фільтрації, користуються командою **Записи/Фільтр/Изменить фильтр**. Відмовитися від обраного фільтру можна командою **Удалить фильтр у Записях**.

Фільтр по выделенному демонструє на екрані тільки ті записи таблиці, які містять виділений фрагмент. Працювати слід так:

- виділити дані, які обираємо за критерій фільтрації. Активізувати команду меню **Записи/Фільтр/Фільтр по выделенному** (можна скористатися й кнопкою панелі інструментів);
- якщо виділити дані й вибрати команду **Записи/Фільтр/Исключить выделенное**, то в таблиці залишаться записи, які не містять виділених даних. Виділяють і окремі символи значень.

Інколи необхідно переглянути записи, які відповідають певним умовам. З цією метою використовують фільтр **простий** або **розширений**.

Простий фільтр дає змогу ввести умову перегляду для одного поля. Він активується командою **Записи/Фільтр/ Изменить фильтр**. На екрані з'являється вікно простого фільтру для Вашої таблиці - воно містить усі назви полів таблиці. У цій таблиці необхідно встановити порядок перегляду записів. Вибрати поле для введення умов пошуку записів таблиці. Умовою може бути як конкретне значення, так і логічний вираз. Зі списку значень поля необхідно вибрати одне чи кілька бажаних значень; значення поля можна ввести і з клавіатури.

Загальний вигляд виразу: <операція порівняння> . . . <значення> . . , наприклад, населення < 100. Оскільки вираз вводиться в конкретне поле, назви поля унікаємо. Вираз можна ускладнити за допомогою логічних функцій (аналогічно вводять умовні вирази в електронних таблицях). У полях текстового типу одразу вводять значення (можна в лапках), для них також використовують шаблони з «*», «?» або іншими символами. Для підказки скористаємося шаблонами табл. 3.

Для полів типу «Дати» є стандартні функції.

Вікно розширеного фільтру викликається командою **Записи/Фільтр/Расширенный фильтр**. У цьому вікні користувач самостійно створює фільтр. Він має бланк, у якому вказує умови вибору записів. У ньому можна вказати й порядок упорядкування для одного чи кількох полів. Для зручності вікно фільтру містить об'єкт, для якого створюють фільтр, із назвами полів.

Таблиця 3

Приклади шаблонів

Символ шаблону	Зміст значення символу шаблону	Запис значення	Приклад результату пошуку
?	Визначає будь-який один символ	К?	КМ, КН, КО
*	Визначає будь-яку групу символів	О*ь	Олень, Оболонь
#	Визначає будь-яку ЦИФРУ	З#-й	35-й, 37-й, 30-й
[...]	Визначає символи, вказані в []	Середн[яі]	Середня, Середні
[!..]	Визначає всі символи, крім тих, що вказані після «!»	Середн[!яі]	Середню

Access дає змогу зберегти результат фільтрації для подальшого використання в роботі з БД — необхідно скористатися командою **Фільтр/Файл/ Сохранить как запрос**.

Хід роботи

1. Завантажити БД «Україна».
2. Розкрити таблицю «Області».
3. За допомогою списку номерів записів перейти до останнього запису.
4. Провидитися таблицею за допомогою лінійки прокрутки.
5. Провидитися запис № 6.
6. Розкрити таблицю «Санаторії».

7. Доповнити таблицю 3-ма записами.
8. Зменшити вікно таблиці, щоб у ньому було видно тільки 3 записи.
9. За допомогою вікна пошуку знайти запис, який містить 19.
10. Виділити поле зі значенням «Туберкульоз».
11. Створити й переглянути записи із застосуванням фільтру за виділенням.
12. Зняти фільтр.
13. Зробити активним вікно таблиці «Області».
14. Впорядкувати дані за збільшенням густоти населення.
15. Створити простий фільтр: *площа більша за ...* , а обласний центр починається на С.
16. Переглянути записи зі створеним фільтром.
17. Створити розширений фільтр: *площа менша за ...* , і за полем «Площа» застосувати впорядкування за зменшенням.
18. Закрити таблиці, базу.
19. Вийти із СУБД.

Контрольні питання

1. Обґрунтуйте актуальність проектування БД.
2. Вкажіть етапи проектування БД.
3. Дайте визначення структури БД.
4. Чим відрізняються поняття *таблична структура БД, таблиця БД*?
5. У якому вікні відбувається вибір об'єкта БД?
6. Перелічіть методи створення таблиць БД в Access.
7. Які відмінності мають режими створення таблиць?
8. Як зберегти структуру таблиці?
9. Які є можливості перегляду записів таблиці?
10. Коли користуються закріпленням колонок?
11. Як вилучити поле таблиці, запис?
12. Як додати до таблиці нове поле, запис?
13. Де Вам траплялися поняття «впорядкування» та «фільтрація»?
14. Які є режими впорядкування?
15. Чим відрізняється фільтр за виділенням від простого фільтру?
16. Якими розширеними можливостями характеризується розширений фільтр?

Лабораторна робота № 15. Зв'язки між таблицями. Запити

Методичні рекомендації

СУБД Access дає змогу створити об'єкти бази даних, у яких буде міститися інформація з різних таблиць. Для цього необхідно встановити зв'язок між таблицями. Під час створення зв'язку будуть об'єднані (зв'язані) записи в цих таблицях. При цьому користуються умовними термінами, говорять про базову та підлеглу таблиці. В обох таблицях повинні бути

поля, які мають однакові значення. Тоді зв'язком між таблицями буде ця пара полів (одне в базовій таблиці, друге в підлеглий). Зв'язані поля можуть мати різні імена, але тип значень цих полів має збігатися.

Поле має однозначно ідентифікувати запис таблиці. Такі поля називаються **ключовими**. Під час зв'язування записи таблиць поєднуються за однаковими значеннями в ключових полях, наприклад за значенням поля «Область» у таблицях «Області» та «Номери машин». Access забороняє дублювання значень та порожні значення в ключових полях. Коли без дубля неможливо обійтися, створюють ключ за двома полями в таблиці.

База даних, яка утримує кілька зв'язаних між собою таблиць, називається *нормалізованою БД*, тобто такою, в якій:

- а) відсутні в таблицях значення, що повторюються;
- б) кожна таблиця має ключ;
- в) усі записи однозначно ідентифікуються ключем.

У MS Access виділяють **три типи ключів**: *лічильник, простий та складений*.

1. Зручно обирати ключем поле типу **Лічильник**. Це поле автоматично змінює свої значення під час введення нового запису. Зв'язок між таблицями БД будуть здійснювати однакові числа поля.

2. Простий ключ. Якщо поле утримує такі унікальні значення, як код або інвентарний номер, його можна об'явити ключем.

Прикладом може бути зв'язок між таблицями «Області» та «№ машин». Перш ніж створити цей зв'язок, визначимо ключове поле в кожній таблиці. В обох таблицях є поля з назвою «Область», значення в цьому полі не повторюється у записах кожної з цих таблиць. Якщо поле «Область» у кожній таблиці об'явити ключем, надалі це дасть можливість поєднати таблиці, а також працювати в інших об'єктах бази даних із записами про область як із таблиці «Області», так і з таблиці «№ машин».

Для зв'язку між таблицями «Області» та «Бази відпочинку» в останню необхідно додати поле «Області». Але визначити його як ключове не можна, бо області містять по кілька баз, і в таблиці цього поля назва області повторюватиметься.

Якщо уникнути повторів зміною даних неможливо, то в таблицю додають поле типу **Лічильник**, надають йому статусу ключового або створюють складний ключ.

3. Складний ключ. Якщо під час створення простого ключа можна говорити про майбутній зв'язок загалом, то під час створення складного ключа необхідно визначити тему зв'язку таблиць (отже, і поля), проаналізувати, чи повторюються значення в полях таблиці.

Таким чином, перед створенням складного ключа необхідно визначити **тип зв'язку** - відношення кількості значень у ключових полях таблиць:

- а) «одне до багатьох» (1-n);
- б) «багато до багатьох» (m-n);

в) «одне до одного» (1-1).

Відношення «одне до багатьох» існує, коли одне з полів, за якими здійснюється зв'язок, — ключове. Тоді одному запису таблиці А відповідає кілька записів таблиці В, але запис із таблиці В не може мати більше від одного, який відповідає йому, запису в А.

Відношення «багато до багатьох» аналогічне до попереднього, коли одному запису із А відповідають кілька із В, і навпаки — одному із В можуть відповідати кілька із А. Таку схему зв'язку реалізують за допомогою третьої таблиці С. Ключ у ній складатиметься якнайменше з 2-х полів: через одне поле із С створюють зв'язок з А, через друге — з В.

Відношення «один до одного» простежується між таблицями з унікальними значеннями в ключових полях (1 запис в А відповідає 1 запису в В). Такий тип зв'язку використовують для розділення таблиць із великою кількістю полів або для відокремлення частини таблиці з метою захисту.

Алгоритм визначення або зміни ключового поля

1. Відкрити таблицю в режимі конструктора.
2. Виділити одне або кілька полів, які будуть визначені як ключові. (Для виділення одного поля клацнути кнопкою миші у рядку цього поля; кілька полів виділяють за допомогою мишки у комбінації з клавішею **Ctrl**).
3. Вибрати команду **Правка/ Ключевое поле** або скористатися кнопкою «Ключевое поле» на панелі інструментів.
4. Повторний виклик команд із п. 3 знімає з поля статус ключового. Але якщо вже був встановлений зв'язок таблиці з іншими об'єктами БД, його попередньо вилучають.

Алгоритм створення зв'язку між таблицями

(зв'язок між таблицями здійснюється у вікні схеми даних БД)

1. Відкрити вікно **Схема даних** із пункту меню **Сервіс** (або активізувати значок цього самого вікна).
2. У меню **Связи** вибрати команду **Добавить таблицу** (або скористатись

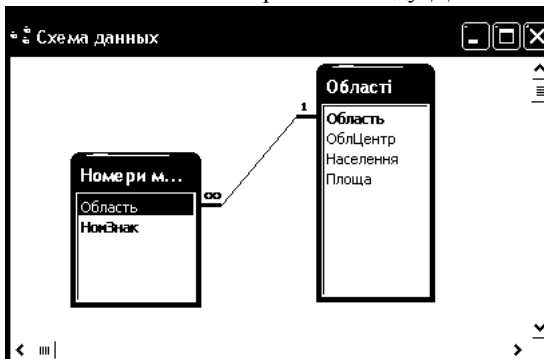


Рис.17 Зв'язок між таблицями «Області» та «Номери машин»

відповідним інструментом). На екрані з'явиться вікно схеми з вікном **Добавление таблицы**, яке містить назви вже створених об'єктів конкретної БД.

3. У вікні **Добавление таблицы** вибрати об'єкт бази **Таблицы**.
4. Для копіювання таблиць у вікно схеми даних достатньо двічі

клікнути по таблиці. Також можна послідовно виділити назви необхідних таблиць і скористатися клавішею **Добавить** (а за допомогою клавіш **Shift та Ctrl** виділяють одразу всі необхідні таблиці). У вікні схеми з'являються таблиці БД із назвами та іменами полів.

5. Вікно **Добавление таблицы** можна закрити.
6. За допомогою мишки перетягнути ключове поле з однієї таблиці в іншу — зв'язок встановлений і зображений лінією між таблицями з визначенням відношення.

За допомогою запитів фільтрують дані кількох зв'язаних об'єктів, які відповідають певним умовам, для подальшої роботи з ними. Запит-вибір дає змогу фільтрувати інформацію БД без зміни її даних, запит-змінна може переміщувати дані та змінювати їх. Запитом можна скористатися для створення нової таблиці чи форми. Перед створенням запиту з'ясовують такі питання: за якими полями фільтруватимуться дані, які значення необхідно знайти в БД, у які поля необхідно записати знайдені записи.

Основою створення нового запиту можуть бути таблиці та наявні запити. Коли викликають команду створення запиту, СУБД пропонує бланк, який необхідно заповнити відповідно до поставлених запитань щодо роботи з записами БД. Дані бланку обробляють автоматично. Результатом є новий об'єкт БД — запит у вигляді таблиці з відібраними даними та результатом обробки цих даних. Слід зауважити, що дані, отримані у запиті, не зберігаються, користувач не може їх змінити. Результати обчислень запиту оновлюються під час його виконання. Таким чином, запит відмежує поточні значення таблиць.

СУБД Access передбачає такі види запитів:

- а) запит-вибір (або простий запит);
- б) запит-дія;
- в) параметричний запит;
- г) перехресний запит.

Запит-вибір — це таблиця, яка містить окремі дані з однієї чи кількох таблиць або вже наявних запитів, відібрані за заданими умовами для подальшої роботи з ними.

Запит створюють за допомогою **Мастера** або самостійно в режимі конструктора. Як завжди, **Мастер** пропонує крок за кроком етапи створення запитів.

Таблиця 4 пропонує алгоритм створення запиту для частини завдання

1. Ця частина завдання схожа на фільтрацію. Але простий запит також використовують для обчислень. Для фіксації результату обчислень використовують поле, значення якого будуть аргументами для таких вбудованих функцій СУБД:

- а) суми значень поля (функція **Sum**);
- б) середнє значення (**Avg**);
- в) мінімальне або максимальне (**Min, Max**) значення поля;
- г) кількість значень поля (**Count**);

Алгоритм створення простого запиту (запиту-вибору) в режимі конструктора

	Дії користувача	Відображення дій на екрані
1	Серед об'єктів БД вибрати Запроси	
2	Вибрати режим створення - Конструктор	На екрані зображено вікно запиту, розділене на дві частини, та вікно Добавление таблицы . З цим діалоговим вікном працювали під час створення зв'язків між таблицями. У ньому є об'єкти БД (таблиці та запити), на основі яких створюють новий запит. Серед команд меню СУБД з'являється пункт Запрос
3	Із вікна Добавление таблицы відібрати об'єкти для нового запиту	У вікні запиту одразу з'являються визначені таблиці або вже створені запити з іменами полів
4	Після вибору таблиць вікно Добавление таблицы закрити	Якщо виникне необхідність ще в одній таблиці, це вікно відкривається командою Запрос/Добавить або кліком по відповідній кнопці інструменту
5	У бланк запиту, який розташований у нижній частині вікна запиту, вставляють поля, які необхідні для подальшої роботи (аналогічно до розширеного фільтру: використовують подвійний клік мишкою на полі; кілька полів, які перетягують у бланк) виділяють одинарним кліком з використанням клавіатури	Форма бланку запиту вже знайома з вікна створення розширеного фільтру. Верхній рядок бланку - назви обраних полів. Оскільки запит створюють на основі кількох таблиць та запитів, на відміну від фільтру, в бланк додають рядок назви відповідного об'єкта (другий рядок)
6	Вказати умови вибору значень полів (аналогічно до фільтру)	Заповнюють рядок умов бланку
7	Поставити прапорець для зображення полів нового запиту	
8	Закрити вікно запиту із збереженням створеного запиту	СУБД пропонує зберегти запит за стандартом Запит.№ , замість якого можна ввести нове ім'я

У режимі Мастера необхідно виділити поле, зі значеннями якого здійснюють обчислення, та вказати, що запит буде «підсумковим».

У режимі конструктора запитів (виконання групових обчислень) виконують вищезазначені 5 дій, далі можна ввести умову вибору і здійснювати обчислення із вибраними значеннями. Для обчислень на панелі інструментів активізують кнопку **«Групові дії»**, або користаються цією самою командою із меню **Вид**, тобто дії з групою значень. У бланку запиту одразу з'явиться рядок **Групповая операция**. У відповідному полі цього рядка вибирають необхідну операцію.

Для перевірки роботи запита користуються інструментом **(Вид)** або командою **Запуск** із меню **Запрос**, не виходячи із вікна створення запиту. Запит зберігають і закривають вікно.

Запити з параметрами дають можливість користувачу отримати відповідь на щойно поставлене запитання. Щоб вивести на екран області, які починаються на певну літеру, можна скористатися фільтром. А можна створити запит, у роботі якого кожного разу вводитимемо літеру та отримуватимемо відповідь. Для цього під час створення запиту у

відповідному полі в рядку умов бланка запиту необхідно прописати умову. Але літеру писати не потрібно, а замість неї в квадратних дужках пишуть підказку: [Введи першу літеру області]. Загальний вигляд рядка умов буде таким: Like [Введи першу літеру області] & «*»

Після запуску запиту на екрані з'являється діалогове вікно, в якому є запитання, введене користувачем у квадратні дужки, та поле для значення-відповіді. Коли відповідь ввели, на екрані з'являється таблиця запиту, як і в попередніх запитах.

Алгоритм створення запиту з параметрами

1. Відкрити вікно створення простого запиту.
2. У вікно запиту перемістити необхідні об'єкти.
3. У бланк запиту зі списку полів помістити необхідні поля.
4. У рядок умов відповідного поля ввести необхідну умову.
5. Зберегти запит.

Запит-дія (запит на зміну) використовується для створення таблиці або роботи з записами таблиці: вилучення запитів, оновлення їх або додавання нових. Зміни відбуваються одразу в кількох записах. Є такі типи запитів на зміну: *запити на вилучення, поновлення та додавання записів.*

Хід роботи

1. Відкрийте для роботи Вашу БД «Україна»;
2. Відкрийте таблицю «Області» в **Режиме конструктора**;
3. Поле «Область» зробіть ключовим;
4. Аналогічно визначте ключовим полем поле «Область» у таблиці «№ машин»;
5. Змініть структуру таблиці «Санаторії» в **Режиме конструктора**.
Додайте поля:
 - а) «№ з/п» (тип - лічильник), якщо є поле «Код», змініть його назву;
 - б) «Область» (тип - текстовий);
 - в) встановіть складний ключ для полів «№ з/п» та «Область»;
6. Відкрийте таблицю «Санаторії» та введіть назви областей у відповідне поле. У вікні Схеми даних за полями «Область» створіть зв'язки між таблицями «Області», «Санаторії», «№ машин»
7. Збережіть БД зі створеними зв'язками;
8. Активізуйте об'єкт **Запрос**.
9. Оберіть команду **Создать**.
10. Виберіть режим створення **Конструктор**.
11. Із вікна **Добавление таблицы** виберіть необхідні таблиці «Області» та «Санаторії».
12. У бланк запиту перетягніть такі поля: із таблиці Області «Площа» та «Населення», із таблиці Санаторії «Назва Санатор».
13. У бланку запиту в рядку **Условие отбора** для поля «Населення» встановіть умову >3000000.

14. Активізуйте команду **Групові дії** з меню Вид.
15. У бланку запиту з'явиться рядок із відповідною назвою. У цьому рядку в полі *НазвСанатор* із списку виберіть операцію рахування кількості **Count**.
16. Перегляньте створений запит: через інструмент або командою **Запуск** із меню **Запрос**.
17. Закрийте вікно створення запиту із збереженням своєї роботи.
18. Самостійно створіть запит в режимі **Майстра** створення запитів.
19. Збережіть створений запит.
20. Відкрийте вікно створення запитів.
21. Додайте в вікно таблицю *Області*.
22. У бланк запиту перемістіть усі поля таблиці.
23. Додайте поле *Густота*: **[Населення]/[Площа]**.
24. Збережіть запит під новим ім'ям.

Контрольні запитання

1. Що таке «ключове поле» таблиці БД? Чому таблиці БД повинні мати ключові поля?
2. Як встановити ключ таблиці?
3. Для чого полям таблиці надають статусу ключового?
4. Правила надання полям статусу ключового.
5. Як відмінити ключ? Чи завжди це можна зробити?
6. Яким вікном користуються для встановлення зв'язку між таблицями?
7. Пригадайте визначення об'єкта СУБД «запит».
8. Які є запити?
9. Чим запити відрізняються від фільтра?
10. Як зберігають запити ?
11. Що означає термін «групові дії»?

Лабораторна робота № 16. Об'єкти БД – форми та звіти. Способи створення форм.

Методичні рекомендації

Форми — це спеціальні вікна, які призначені для полегшення перегляду записів у БД або для введення та редагування даних. Форми створюють на основі таблиць чи запитів. Форма може містити вибірку інформації з об'єктів, тому її використовують у роботі кількох користувачів з однією БД, і для кожного користувача можуть бути доступні окремі форми з певною інформацією БД.

Створення нової форми аналогічне до створення будь-якого об'єкта БД.

1. У вікні бази даних вибрати для роботи об'єкт «**Форми**».
2. Активізувати команду **Создать**.

3. У вікні створення форми вибрати зі списку ім'я таблиці, для даних якої створюють форму, та режим створення форми.

Працювати з формою можна в режимі *конструктора, форми* та у режимі *таблиці*.

Режимом конструктора користуються для створення форми за своїм проектом. У цьому випадку користувач сам вирішує, які елементи міститиме форма, в якому місці форми будуть розташовані та який вигляд (розмір, колір) вони матимуть.

Із готовими шаблонами форм працюють у решті режимів. Користуючись шаблонами СУБД, можна створити форми в таких режимах:

- а) за допомогою автоформи;
- б) використовуючи **Мастер форм**;

У режимі автоформи у форму переносять усі поля вказаних таблиці або запиту. За розташуванням полів розрізняють такі автоформи:

- а) в колонку;
- б) в рядок;
- в) табличні.

В автоформі всі поля розташовані на окремому рядку, в колонку, а ліворуч від неї, в другій колонці вказано значення полів. Бланк форми на екрані містить один запис. Форми теж містять панель навігації, звідки можна переглядати всі записи.

У рядкових **автоформах** значення записів виводяться рядками, над якими є назви полів. Форма виводить на екран усі поля записів, розміщені у розмірі формі, тому значення кількох або всіх полів можуть бути скороченими.

Таблична автоформа має вигляд таблиці, у якій кожному запису відповідає рядок, а полю — колонка. Імена полів такі самі, як в об'єкті «таблиці». Таблична автоформа містить лінійку прокрутки для перегляду значень полів, що не знаходяться в розмірі форми.

Використання режиму майстра є поетапним створенням форми за визначеними СУБД вказівками.

Мастер теж пропонує вигляд форм, як і автоформа. Але якщо при виборі автоформи у форму автоматично переміщуватися усі поля початкового об'єкта, то в режимі майстра користувач сам вирішує, які поля будуть у формі. **Мастер** також дає змогу використати для форми декілька об'єктів, користувач сам вирішить, які поля з них необхідно перенести у форму.

Якщо записи таблиці, на базі якої створюють форму, були впорядковані, дані форми також будуть упорядковані автоматично. У формі зберігається і фільтрація даних таблиці для першого перегляду цих даних. У наступних сеансах роботи фільтрацію даних у формі необхідно поновити.

Підпорядкована форма — це форма, яка знаходиться всередині іншої форми. Первинна форма називається головною формою, а форма всередині форми називається підлеглою формою.

Комбінацію «форма/підпорядкована форма» часто називають ієрархічною формою. Найчастіше підпорядковані форми використовують

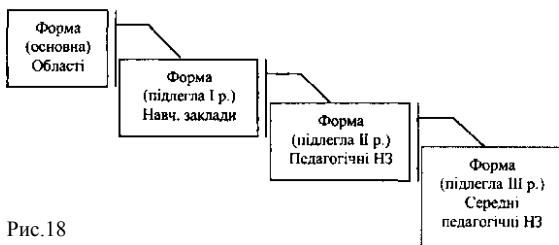


Рис.18

для зв'язку між таблицями «один до багатьох». На прикладі БД «Україна» таку комбінацію форм використаємо для таблиць «Області-» та «Санаторії», тоді головна форма

міститиме головну інформацію відомості про область, а підлегла — про лікувальні можливості області.

Головна форма завжди є простою формою з одним записом у бланку форми, підлегла може мати вигляд, вибраний за бажанням користувача.

Головна форма може мати довільну кількість підлеглих форм. Також MS Access дає змогу розробляти у формі сім рівнів підпорядкування: основна форма має підпорядковану, та в свою чергу утримує підпорядковану тощо. Прикладом може бути схема форм для нашої БД «Україна» (рис.18).

У готовій складній формі підпорядкована форма може бути представлена як власне підпорядкована, так і зв'язана. Підпорядкована форма розкрита і знаходиться на «території» головної форми, а зв'язана в головній формі подана у вигляді кнопки. Вона розкривається кліком мишки на її кнопки.

У СУБД Access **звіт** являє собою форму спеціального типу, призначену для виведення даних на друк.

Звіт створюють за необхідності під час обробки даних однієї або кількох (декількох) таблиць, запитів, для підведення підсумкових розрахунків та виведення їх на друк. Програма не тільки пропонує готовий вигляд звітів, а й дає змогу змінювати розмір та зовнішній вигляд всіх елементів звіту.

Результатом створення звіту може бути будь-який вигляд текстового документу, наприклад: поштова наклейка, діаграма як графічний засіб представлення результатів обчислень, таблична форма з результатами обчислень та графічним оформленням (емблемами, логотипами, декоративними лініями, тощо).

Звіт є об'єктом БД, призначеним для друку, збереження створеного звіту можливе в тому разі, коли комп'ютер укомплектований принтером. Якщо принтер відсутній, на екрані монітора можна переглянути макет звіту.

Створення звіту, як нового об'єкта бази даних, відбувається за стандартною схемою: у вікні бази даних вибирають об'єкт «**Отчеты**», після активізації команди **Создать** викликають конструктор або майстер.

Майстер, як завжди, пропонує крок за кроком обрати необхідні дії для створення звіту. Графічне оформлення звіту створюють у режимі конструктора.

Перед створенням звіту необхідно продумати, які початкові дані він буде утримувати та які обчислення необхідно здійснити.

Алгоритм створення звіту (пункти алгоритму вказані за вікнами майстра):

1. Відкрити свою базу даних.
2. Вибрати **«Отчеты»**.
3. Вибрати команду **Создать**.
4. Із запропонованих форм звітів обрати команду створення звіту за допомогою майстра.
5. На цьому етапі можна обрати створення звіту у вигляді діаграм, поштових наклеюнок, а також скористатися авто звітом стовпчиковим або смугастим, як у формах.
6. Виконувати вказівки майстра: вибрати із списку таблицю, на основі якої створюється звіт.
7. Із таблиці вибрати поля, дані яких будуть оброблятися звітом.
8. Якщо необхідно додати подібним чином дані з інших таблиць або звітів.
9. Обрати рівні групування.

Під час створення звіту можна виконувати групування даних. Звіт може містити кілька рівнів груп. Наприклад, база даних «Україна» містить таблицю, яка містить відомості про *області України* та всіх *сусідів* кожної області. На базі такої таблиці можна створити звіт, в якому буде зафіксовано, скільки *сусідніх областей* має дана область а також будуть перелічені для кожної області її *сусіди*. Дані в звіті будуть представлені так, що назва області та назви її *сусідів* будуть відокремленими, тобто згрупованими.

Хід роботи

Відкрити БД «Україна».

Створення форм для перегляду та введення даних

1. У вікні БД «Україна» для роботи вибрати об'єкт «*Форми*» та команду створення.
2. У вікні нової форми для створення виберіть режим **«Автоформа»** (тип автоформи визначіть за власним бажанням).
3. У цьому самому вікні виберіть таблицю, на основі якої створимо форму, — «*Області*».
4. Після підтвердження вибору на екрані з'явиться форма з іменами полів та значеннями відповідної таблиці.
5. Перейдіть за допомогою лінійки перегляду записів до останнього запису.
6. Додайте 5 нових записів.
7. Закрийте форму, присвоївши їй назву.

8. Відкрийте таблицю «*Області*», зверніть увагу, чи є в ній нові записи.
9. Створіть ще дві форми для будь-якої таблиці (виберіть типи автоформи, з якими ще не працювали).
10. Скористайтесь режимом майстра для створення форми.
11. За основний об'єкт виберіть таблицю «*Області*».
12. Виберіть такі поля для форми: *Область, ОблЦентр, Населення*.
13. Наступні кроки: вибір вигляду форми (для прикладу виберіть **Выровненный**), вибір фону форми, введення назви.
14. Виберіть команду **Готово**.
15. Відкрийте для перегляду всі форми, порівняйте їхні зміст та вигляд.
Створення складної та підпорядкованої форми.
16. У вікні БД «*Україна*» виберіть об'єкт «*Форми*» та команду створення форми в режимі майстра.
17. У форму скопіюйте такі поля:
 - а) із таблиці «*Області*» - **Область, ОблЦентр, Населення**;
 - б) із таблиці «*№ машин*» - **НомЗнак**;
 - в) із таблиці «*Санаторії*» - **НазваСанаторію та ЛікувальнийПрофіль**.
18. Подання даних щодо санаторіїв оформіть як підлеглу форму.
19. Збережіть створену форму під назвою **Лікування**, а назву підлеглого не змінюйте.
20. Закрийте форму (у вікні БД «*Україна*» залишіться у вкладниці «*Форми*»).
- Робота з даними у формах**
21. Відкрийте підпорядковану форму (вона одна у БД). Продивіться записи у підпорядкованій формі.
22. Відсортуйте записи за зростанням для поля «*Лікувальний профіль*».
23. Закрийте підпорядковану форму.
24. Відкрийте форму **Лікування**.
25. Передивіться записи основної форми та підпорядкованої за допомогою лінійок перегляду записів.
26. Закрийте форму. (Приклади форм наведено у схемах 8 та 9)
- Робота з формою в режимі конструктора.**
27. Виділіть форму **Лікування**.
28. Виберіть режим конструктора.
29. Вікно конструктора повинно мати панель інструментів та горизонтальну лінійку, які виставляють через пункт меню **Вид**.
30. Відкрийте вікно властивостей форми.
31. Перейдіть на вкладнику **Данные**.
32. У списку **Введение** встановіть значення **Нет** (воно відобразить усі записи, а значення **Да** використовують для введення даних).
33. На панелі інструментів вікна форми інструмент **Мастер** повинен бути активним.
34. Створимо свою панель навігації по записах в основній формі. На

- панелі інструментів активізуйте інструмент **Кнопка**.
35. За допомогою мишки в області форми вкажіть місце та розміри майбутньої кнопки.
 36. Крок за кроком працюємо у вікні **Создание кнопок**:
 - а) у полі **Категории** виберіть **Переходи по записям**, а в полі **Действия** — **Первая запись**;
 - б) нова кнопка буде малюнком (підтвердіть це рішення за допомогою круглих полів вікна), виберіть вигляд своєї кнопки;
 - в) вибір команди «Готово» закриває вікно **Создание кнопок**.
 37. Аналогічно створіть кнопки для переходу на попередній запис, наступний, останній та додавання запису.
 38. Відредагуйте вигляд кнопок (користайтесь мишкою та командами пункту меню **Формат**).
 39. На панелі інструментів виберіть «**Вільна рамка об'єкта**» або «**Рисунок**». Різниця між кнопками полягає у режимі роботи. Перша дає можливість зв'язати форму з файлом об'єкта, що вставляється, друга просто вставляє малюнок.
 40. У формі вкажіть місце та розміри нового об'єкта, а також необхідний файл (виберіть карту України).
 41. Завдяки інструменту підлеглої форми у свою форму додайте підлеглу із таблиці «**Бази відпочинку**».
 42. Створити звіт із групуванням на основі даних таблиць *Області* та *Номери машин*, що відображав би інформацію про зареєстровані на території кожної області автомобілі (із вказанням дати реєстрації).

Лабораторна робота № 17. Створення презентації індивідуального завдання за допомогою Power Point.

Методичні рекомендації

Області застосування презентації досить великі. За допомогою презентації можна зробити огляд проекту, навчального курсу, докласти про пророблену роботу, запропонувати нову стратегію, нові ідеї, повідомити про нововведення в педагогіці. Але, насамперед, презентація використовується для анонсування нової теми і для знайомства з нею. Головне в презентації - це тезисність (для виступаючого) і наочність (для слухача).

Щоб аудиторія в процесі показу презентації не втрачала до неї інтерес, особливу увагу потрібно звернути на дизайн презентації. Зробити презентацію привабливою допоможуть яскраві малюнки, що будять інтерес у глядача. Тому всі слайди чи презентації, або хоча б їхню частину, варто прикрасити графікою.

1. Способи створення презентації

Програма PowerPoint володіє всіма необхідними інструментами, що дозволяють навіть новачкам за короткий час створювати професійні слайди. Після запуску програми відкривається діалогове вікно PowerPoint. У цьому вікні пропонуються три різних способи створення презентації: майстер „автосодержання”, шаблон презентації і порожня презентація.

Основними елементами презентації є слайди. Для презентації одного слайда явно недостатньо. Після вибору способу створення презентації й оформлення першого слайда, вам необхідно ввести наступний. Другий слайд не з'явиться на екрані автоматично, він повинен бути вставлений у встановленому порядку. Для цього на стандартній панелі існує кнопка "Новий слайд" (не плутайте з кнопкою "Створити" - вона створює нову презентацію, а не новий слайд)

Розглянемо способи створення презентації.

Майстер автосодержання пропонує презентації по темах з готовою структурою. На екрані пропонується структура презентації доповіді. Якщо ваша тема презентації збігається з наявною в "майстра", можете скористатися запропонованим. Перехід по слайдах можна здійснити за допомогою смуги прокручування.

Можна створити презентацію на основі готового шаблону, у якому пропонується визначений стиль для презентації. Якщо в процесі роботи над презентацією, ви захочете змінити стиль - видаліть фон. Можна вибрати інший шаблон чи створити свій.

З роботою програми PowerPoint найкраще знайомитися в положенні – «порожня презентація». Інші шляхи створення презентації надають менше можливостей керування програмою, а тому менш придатні для демонстрації всіх можливостей PowerPoint.

При виборі способу "порожня презентація" пропонується 24 шаблону для створення різних типів слайдів презентації. Уважно

перегляньте запропоновані варіанти і виберіть один із шаблонів слайда. Який саме з них підійде для ваших цілей, залежить, зрозуміло, від виду презентації і того, що ви хочете представити на слайдах. Програма PowerPoint пропонує шаблони слайдів для: тексту, тексту з діаграмами, графіки й інших об'єктів. Крім того, передбачені різні варіанти розміщення згаданих елементів на слайді. Якщо вам не підходить жоден з цих шаблонів, виберіть порожній слайд, на якому можна помістити текст, графіку й інші елементи за своїм розсудом.

В одній презентації можуть бути слайди різних типів. На рис.19 зображений фрагмент розділу структури, де зазначені номер слайда - значок слайда - заголовок слайда. Активний слайд, з яким у даний момент можна працювати, позначений чорним кольором.

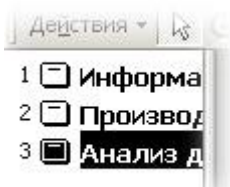


Рис.19 Фрагмент структури презентації

2. Режими роботи програми

У програмі PowerPoint використовуються кілька режимів представлення слайдів. Кожний з цих режимів має свій набір інструментів і орієнтований на виконання визначених дій. Хоча більшість операцій можна виконати в різних режимах, найбільше зручно працювати в тому режимі представлення слайдів, що призначений для розв'язання відповідної конкретної задачі.

Вибрати режим можна за допомогою пункту меню програми Вид , чи в нижньому лівому куті екрана - у вигляді іконок (рис.20).

Один з основних режимів роботи програми PowerPoint - це режим слайдів. На екрані розміщується тільки слайд. Як слідує з назви режиму, у ньому створюються і коректуються слайди, на яких можна розмістити текст, чи графіку створити діаграми.

У режимі слайдів найпростіше встановити анімаційні ефекти для окремих елементів слайда (малюнка, тексту).

Передбачено й інший режим - сортувальника слайдів, використовуваний для визначення послідовності появи слайдів на екрані. Даний режим, крім іншого, надає можливість вибору ефектів появи слайдів, тобто визначається швидкість і спосіб заміщення (по щиглику миші чи автоматично) одного слайда іншим у процесі презентації.

У режимі структури формується структура всієї презентації. Ви можете ввести заголовки для слайдів, а також текст, що повинен з'явитися на різних рівнях структури під цими заголовками. У режимі перегляду структури текст можна форматувати. Відповідно до заданого тут сценарієм PowerPoint створює слайди.

На жаль, у цьому режимі ви обмежені тільки введенням тексту, а вставка графіку не допускається.

У режимі перегляду структури на екрані видні тільки значки слайдів, поруч з якими зображені їх номери. Поруч зі значками розташований текст, що використовується на відповідних слайдах. Під заголовками слайдів з невеликим відступом вправо впливає текст, що починається із символу маркера, який на слайді з'являється в текстовому полі підзаголовка. Набраний у полях списків текст відображається з маркерами. Чим більше відступ тексту, тим більше низький рівень в ієрархії він займає. Вид режиму структури.

Програма PowerPoint у режимі структури може автоматично створити слайд, що містить заголовки виділених слайдів. Слайд змісту (підсумковий слайд) вставляється перед виділеною областю.

При збереженні презентації як Web-документа область режиму структури перетвориться в меню, при цьому заголовки слайдів автоматично

Рис.20 Вибір режимів представлення



стануть гіперпосиланнями на відповідний слайд. Перехід від слайда до слайда при перегляді презентації буде відбуватися не по щиглику миші, а з використанням меню.

Для зручності режим слайдів і режим структури сполучається в одному режимі - звичайному. Праворуч зверху розташована область, у якій відображається слайд, що редагується, під ним розташована область заміток, а ліворуч - область структури, у якій відображений порядок слайдів презентації і розміщені на них заголовки/підзаголовки.

Перетягуючи мишею границі між областями, можна змінювати їх розмір. Використовуваний за замовчуванням для відображення в області слайдів масштаб вибирається програмою автоматично, щоб слайд був видний цілком.

Замітки надають велику допомогу під час доповіді чи презентації, дозволяючи доповідачу не тримати всю потрібну інформацію в себе в пам'яті. Подивитися процес проходження презентації можна в режимі показу слайдів, коли слайд розташовується на весь екран і запускаються в дію всі анімаційні ефекти, раніше вами встановлені для кожного об'єкта слайдів. Перехід від слайда до слайда здійснюється в залежності від того, який вид переходу ви встановили для зміни слайдів - автоматично чи по щиглику миші.

Хід роботи

1. На Панелі інструментів клацніть мишкою знак “Создать”.
2. У вікні “Создание слайда” в списку авторозміток знайдіть “Титульний слайд” і двічі клацніть на ньому мишкою.
3. У полі заголовку двічі клацніть мишкою і введіть текст “Коледж ЧНУ”.
4. У полі підзаголовку введіть текст “створений у 2001 році”.
5. Виберіть пункт меню “Вставка”, “Рисунок”, “Из файла” і оберіть зображення для вставки. Розмістіть його у лівому верхньому куті.
6. Клацніть пункт меню **Вставка Новий слайд**.
7. У вікні “Создание слайда” в списку авторозміток знайдіть “Маркований список” і двічі клацніть на ньому мишкою.
8. У полі заголовку введіть текст “Спеціальності”. В полі тексту введіть назви спеціальностей нашого коледжу, натискаючи **Enter** після кожної з них.
9. Вставте ще один малюнок на цьому слайді.
10. Створіть ще 5 нових слайдів Таблиця. У кожному з них необхідно розмістити інформацію “Студенти спеціальності ”. Кожний з цих слайдів необхідно прикріпити до слайду “Спеціальності” відповідно до назви спеціальності.
11. Створіть слайд “Організаційна структура” на основі слайду Організаційна діаграма. На слайді відобразіть організаційну структуру коледжу.

12. На панелі **Команди** натисніть “**Применить шаблон оформления**”.
- Оберіть дизайн слайдів. Дизайн для кожного можна обрати різний.
13. Оберіть “**Вид**” “**Показ слайдов**”.
14. Перегляньте створену презентацію.
15. Змініть оформлення слайдів. Доповніть презентацію новими слайдами.
16. Встановіть Режим сортувальника слайдів, виділіть їх всі.
17. Виконайте команду меню “**Показ слайдов**”, “Смена слайдов”.
- Встановіть режим показу слайдів.
18. Перегляньте створену презентацію.
19. Спробуйте створити презентацію за допомогою шаблону презентацій.

Контрольні запитання

1. Область застосування презентацій.
2. Які режими представлення слайдів є в програмі Power Point?
3. Назвіть три різних способи створення презентації.
4. Як здійснюється перехід по слайдах?
5. Для чого використовуються шаблони слайдів? Які бувають шаблони?

РОЗДІЛ 4. КОМУНІКАЦІЙНІ ЗАСОБИ ЕОМ

Лабораторна робота №18. Робота в локальній мережі Windows.

Методичні рекомендації

1. Однією з форм реалізації передачі інформації між комп'ютерами є мережа. Комп'ютерна мережа - це група комп'ютерів, які з'єднані між собою за допомогою спеціального мережного устаткування та спеціального програмного забезпечення. Комп'ютерні мережі призначені для:

- спільного використання апаратних та програмних засобів комп'ютерів;

- забезпечення спільного доступу до ресурсів даних.

Комп'ютерні мережі прийнято ділити на 2 класи:

- локальні (LAN - Local Area Network);

- глобальні (WAN - Wide Area Network).

Локальна мережа - це набір комп'ютерів, периферійних пристроїв, комунікаційних пристроїв, що з'єднані кабелями та відповідного програмного забезпечення, яке міститься у Windows. У мережі є багато комп'ютерів і для однозначної ідентифікації кожен комп'ютер у мережі має своє ім'я, під яким його "бачать" інші комп'ютери.

Комп'ютери в мережі можуть бути рівноправними й можуть користуватися ресурсами один одного. Такі мережі, що переважно базуються на використанні операційної системи Windows, називаються одноранговими. Якщо в мережі є спеціальний комп'ютер, виділений для спільного використання учасниками мережі, то він називається сервером.

Цей комп'ютер виконує задачі управління мережею. Отже, серверна мережа - це мережа, в якій комп'ютери мають різні функції - одні з них виступають в ролі серверів, що поставляють ресурси та послуги, інші комп'ютери відіграють роль клієнтів, що користуються цими ресурсами та послугами. Комп'ютери в мережі можуть об'єднуватися в робочі групи. Кожен комп'ютер належить до якоїсь робочої групи.

Для встановлення параметрів роботи локальної мережі можна скористатися програмою **Сеть** (Пуск=>Настройка=> Панель управління=>Сеть). У вікні цієї програми можна визначити конфігурацію локальної мережі (тип мережної плати, встановленої на комп'ютері, мережний протокол, службу доступу до файлів і принтерів у мережі, визначити (задати) мережне ім'я комп'ютера, визначити робочу групу, до якої він належатиме та ін.). Після завершення налаштування конфігурації, комп'ютер користувача матиме доступ до ресурсів мережі. На робочому столі Windows з'явиться піктограма **Сетевое окружение**. Відкривши вікно піктограми Сетевое окружение, ми побачимо комп'ютери зі своєї групи, а також значок **Вся сеть**. Усі комп'ютери мережі доступні через папку Вся сеть, при цьому вони розбиті на робочі групи. Мережне ім'я комп'ютера можна змінити через контекстне меню папки Сетевое окружение, вибравши пункт Свойства.

Спільні ресурси - це компонент іншого комп'ютера, наприклад, папка, файл або принтер, до якого був відкритий доступ через мережу.

Призначення спільної папки. Потрібно вибрати папку за допомогою програми Мой компьютер або Проводник. Потім із контекстного меню папки вибрати команду Свойства=>вкладка **Доступ** і задати параметр **Общий ресурс**. Можна встановити доступ тільки для читання інформації, повний доступ (для читання та запису) і доступ з паролем. Завершують операцію натисненням кнопки ОК. На піктограмі ресурсів загального призначення на комп'ютері користувача з'являється зображення руки, на долоні якої знаходиться значок відповідного ресурсу.

Відкриття спільної папки на іншому комп'ютері. За допомогою програми **Сетевое окружение** відкривають ресурси необхідного комп'ютера, потім відкривають потрібну папку. Операції з файлами і папками здійснюються аналогічно, як при звичайній роботі в системі Windows.

У локальній мережі для друкування можна використовувати один принтер. Для цього виконують такі дії: відкривають папку **Принтеры** (Пуск=>Настройка=> Принтеры) на комп'ютері, до якого приєднують принтер; у контекстному меню принтера вибирають команду **Свойства**, вкладку **Доступ** і задають параметр **Общий ресурс**.

Для управління доступом на рівні користувачів на вкладці **Доступ** натискають кнопку **Добавить** для визначення користувачів, яким буде дозволено використовувати принтер.

Для друку на мережному принтері потрібно підключитись до мережного принтера. Для цього в папці **Принтеры** натискають **Установить принтер**. З'явиться вікно майстра встановлення принтера, який може допомогти встановити драйвер потрібного принтера. Якщо принтер підключений до іншого комп'ютера, то потрібно вибрати кнопку "Сетевой принтер" і задати мережний шлях до комп'ютера за допомогою кнопки **Обзор**. У папці **Принтеры** з'явиться піктограма мережного принтера (принтер, під'єднаний до кабелю).

Після встановлення мережного принтера друк здійснюється як на принтері, приєднаному до цього комп'ютера.

Хід роботи.

1. Ознайомитися з параметрами конфігурації локальної мережі.
2. З'ясувати мережне ім'я вашого комп'ютера та робочої групи, до якої він належить.
3. Визначити, які ресурси вашого комп'ютера доступні для роботи в мережі.
4. Відкрити папку **Сетевое окружение**.
 - а) з'ясувати, скільки комп'ютерів вашої робочої групи на даний момент працюють у локальній мережі;
 - б) за допомогою піктограми **Вся сеть** продивитись структуру локальної мережі.
5. Створити папку Net1 і до неї скопіювати з вашого жорсткого диска декілька файлів.
6. Встановити доступ до цієї папки як доступ для читання. Використати контекстне меню цієї папки, відкрити пункт **Свойства** і на вкладинці **Доступ** задати тип доступу **Только чтение**.
7. Скопіювати файли з папки Net1 іншого комп'ютера в папку Net1 на своєму комп'ютері. - Для цього необхідно використати звичайні засоби роботи з файловою системою у Windows, тобто спочатку відкрити папку ProbaNet на іншому комп'ютері, виконати команду **Копировать**, потім відкрити папку ProbaNet на своєму комп'ютері і задати команду **Вставить**.
8. Передати текстове повідомлення про здачу іспиту з інформатики зі свого комп'ютера на інший активний комп'ютер мережі, отримати відповідь. Для цього необхідно запустити на виконання програму winporcup.exe (папка Windows).
9. Підключитись до мережного принтера.

Контрольні питання

1. Чому виникла необхідність створення комп'ютерних мереж?
2. Що таке комп'ютерна мережа?
3. Які ви знаєте апаратні засоби для створення комп'ютерної мережі?
4. Для чого призначені комп'ютерні мережі?
5. Що таке локальна мережа?

6. Яка мережа називається одноранговою?
7. Як побачити ім'я комп'ютерів, які складають локальну мережу?
8. Що таке сервер?
9. Яка мережа називається серверною мережею?
10. Що таке робоча група?
11. Як встановити (перевірити) параметри локальної мережі?
12. Для чого призначена програма Сетевое окружение?
13. Як можна змінити мережне ім'я комп'ютера?
14. Що таке спільні ресурси?
15. Як зробити папку спільною?
16. Які типи доступу до ресурсу ви знаєте?
17. Як відкрити папку на іншому комп'ютері?
18. Як здійснити копіювання файлів з одного комп'ютера на інший?
19. Як визначити свій принтер для спільного користування в локальній мережі?
20. Як підключитися до мережного принтера?

**Лабораторна робота №19. Робота в Інтернет з допомогою браузера
Internet Explorer. Дії з повідомленнями в Outlook Express.**

Методичні рекомендації

Internet - це всесвітня інформаційна глобальна комп'ютерна мережа, яка об'єднує багато комп'ютерних мереж для обміну даними і доступу до інформаційних ресурсів. Internet складається з багатьох комп'ютерів, з'єднаних між собою лініями зв'язку і встановлених на цих комп'ютерах відповідних програм.

Для передачі інформації в Internet використовується протокол TCP/IP. Протокол це сукупність правил взаємодії комп'ютерів. Мережний протокол розписує комп'ютерам правила роботи (це є мова спілкування комп'ютерів). В Internet є декілька рівнів протоколів, які взаємодіють один із одним. На нижньому рівні використовують два основних протоколи **TCP** (Протокол управління передачею даних) та **IP** (Інтернет протокол). Часто ці два протоколи об'єднуються в один - TCP/IP.

Протокол TCP- це протокол транспортного рівня. Він керує передачею інформації. При цьому дані, що відправляються, розподіляються на порції. Та нумеруються таким чином, щоб при отриманні їх можна було б правильно "зібрати" на комп'ютері одержувача.

Суть адресного протоколу IP полягає в тому, що в кожного учасника Всесвітньої мережі повинна бути своя унікальна цифрова IP-адреса. Без цієї адреси не можна говорити про точну доставку TCP-пакетів на відповідний комп'ютер. Цифрова адреса виражається чотирма байтами, наприклад, 126.37.215.61, і зручна для обробки на комп'ютері. Кожен комп'ютер

мережі, через який проходить TCP-пакет по чотирьох байтах, може визначити, якому наступному комп'ютеру треба переслати пакет, щоб він виявився "ближче" до одержувача. Розв'язанням питань, який комп'ютер вважати "ближчим", займаються спеціальні засоби маршрутизатори - спеціалізовані програми, що працюють на вузловому сервері мережі.

Користувачу не зручно працювати з числовою IP-адресою, тому для користувача кожен комп'ютер мережі має ще й доменне ім'я. У доменній адресі є різні рівні, що називаються доменами, які відділяються крапкою. В доменній системі імен використовується принцип послідовного уточнення. Домен верхнього рівня розміщений справа і позначає графічний регіон - країну (ua - Україна, ru - Росія, de - Німеччина і т.д.). Домени нижнього рівня розміщені зліва (їх може бути декілька). Наприклад, адреса www.chnu.cv.ua вказує на Web-сторінку Чернівецького національного університету.

Доменне ім'я легко запам'ятовується, тому що, як правило, воно вказує на зміст Internet-ресурсу. Значить, користувач працює з доменною адресою, а автоматична робота серверів організована з використанням чотирьохбайтової числової адресації, тому необхідно зіставляти доменні імена зі зв'язаними з ними IP-адресами. Цим і займаються сервери служби імен доменів(DSM). Наш запит на одержання деякого файлу з серверу спочатку обробляється сервером DSM, а далі він пересилається згідно з відповідною IP-адресою.

В Internet діють декілька служб (різних видів послуг). У залежності від задач, користувачі Internet використовують ті чи інші служби.

Різні служби мають різні протоколи, їхнє дотримання забезпечується роботою спеціальних програм. Таким чином, щоб скористатися якоюсь із служб Інтернету, необхідно встановити на комп'ютері програму, яка може працювати з протоколом даної служби. Такі програми називаються клієнтами.

Так, наприклад, для передачі файлів в Інтернеті використовується спеціальний прикладний протокол FTP (File Transfer Protocol). Таким чином, щоб одержати з Інтернету файл, необхідно: мати на комп'ютері програму FTP-клієнт; встановити зв'язок із сервером, що надає послуги FTP (FTP-сервер). Найпопулярнішою службою сучасного Інтернету є служба WWW (World Wide Web). WWW являє собою єдиний інформаційний простір, що складається з величезної кількості взаємозв'язаних електронних документів, які зберігаються на Web-серверах. Окремі документи, що складають Web- простір, називають Web-сторінками. Групи тематично об'єднаних Web-сторінок називають Web-вузлами або Web-сайтами. Один фізичний Web-сервер може містити багато Web-вузлів, кожному з яких, як правило, відводиться окремий каталог на жорсткому диску сервера.

Користувачам служби WWW надається інформація у вигляді гіпертекстових документів, які у свою чергу теж можуть мати гіперпосилання на інші документи. Гіпертекстовими посиланнями можуть бути слова, сполучення, зображення, які виділені кольором, підкресленням

та рамкою. Сукупність великої кількості гіпертекстових документів, що зберігаються на серверах WWW, утворює своєрідний гіперпростір документів, між якими можливе переміщення. Користувач, почавши навігацію з деякого Web-документа, може переходити до нових документів як в межах цього ж Web-сервера, так і на інших серверах і здійснювати перегляд документів.

Перегляд документів починається, як правило з домашньої сторінки. Програми для перегляду Web-сторінок називаються браузерами. Документи, що є на Web-серверах, формуються за допомогою спеціальних правил **HTML** (Hyper Text Markup Language - мова розмітки гіпертекстів). Відображення документа на екрані відбувається згідно з командою мови розмітки гіпертексту, і ці команди називаються тегами. Наприклад, за допомогою тегів створюються гіпертекстові посилання. Такі документи ще називають документами в HTML форматі. При відображенні HTML документу на екрані бачимо тільки текст документа (теги не відображаються). Клацнувши лівою кнопкою миші на гіперпосиланні, тим самим відправляємо запит на відкриття нового документа.

Такий зв'язок між документами існує, тому що кожна адреса файлу в Інтернеті визначається уніфікованим вказівником ресурсів - **URL - адреси**. Найчастіше URL адреса має такий формат протоколу: //доменна адреса комп'ютера, на якому зберігається даний ресурс/ім'я каталогу/ім'я підкаталогу/ім'я файлу.

Протокол http - протокол передачі гіпертекстів для служби WWW, ftp - протокол пересилання файлів довільного типу, nntp - протокол передачі новин.

Служба передачі файлів (FTP) використовується при передачі файлів-програм, при пересиланні великих документів (книг), при передачі великих архівних файлів. Служба FTP має свої сервери у світовій мережі, на яких зберігаються архіви даних. У користувача для роботи із серверами FTP може бути встановлене спеціальне програмне забезпечення, хоча і браузери мають вбудовані можливості для роботи з протоколом FTP, але спеціалізовані FTP-клієнти забезпечують більш стійке з'єднання й володіють зручнішим інтерфейсом для користувача.

Типовим прикладом браузера є програма Internet Explorer, яка входить до складу Windows. Вона призначена для перегляду Web-документів і для доступу до FTP-архівів.

Запуск браузера відбувається через значок на  робочому столі або на панелі швидкого запуску через операцію **Пуск=>Программи=>Internet Explorer**, через документ у форматі HTML.

Перегляд Web-сторінок. Як тільки ви запустили Web браузер, він автоматично завантажить, як правило, початкову сторінку компанії, що випустила цей браузер. Після цього для відображення у вікні браузера необхідної Web-сторінки потрібно:

- клацнути в рядку введення адреси (рядок Адрес, який

знаходиться у верхній частині вікна);

- надрукувати URL-адресу необхідного Інтернет-ресурсу й натиснути клавішу **Enter** (стара адреса повинна зникнути). Можна вибрати адресу із списку, що розкривається;
- через декілька секунд нова сторінка завантажиться на ваш комп'ютер.

При перегляді Web-сторінки зустрічаються гіперпосилання на інші Web-документи. Гіперпосилання в тексті виділяються кольором і підкресленням.

При наведенні курсору миші на гіперпосилання курсор набуває форми людської руки з витягнутим вказівним пальцем, а URL-адреса нової Web-сторінки з'являється в рядку статусу. Клацнувши мишею на гіперпосиланні, ми відкриємо на екрані нову Web-сторінку, яка замінить поточну (гіперпосилання відіграє роль ярликів для відшукування Web-сторінок). Щоб переглянути великі Web-сторінки використовують смугу прокрутки. Web-документи можна зберегти на своєму комп'ютері – команда **Файл=>Сохранить как**, можна роздрукувати – команда **Файл=>Печать**.

При подорожуванні у Web-просторі можна зустріти Web-сторінки, які ділять робоче поле браузера на декілька частин **фреймів**. Як правило, в лівому фреймі подано зміст Web-вузла або посилання на його Web-сторінки. Вибравши мишею посилання, розташоване в лівому фреймі, в правому фреймі отримують відображення відповідної сторінки. Щоб активізувати фрейм, треба клацнути мишею по ньому. Для того, щоб запустити зразу ж декілька Web-сторінок, потрібно завантажити ще раз браузер Internet Explorer, клацнувши мишею на його значку і в цьому новому вікні в адресному рядку ввести URL-адресу необхідної Web-сторінки, або скористатися командою **Файл=>Создать=>Окно** або вибрати команду **Открыть в новом окне** з контекстного меню необхідного гіперпосилання, або клацнути мишею на гіперпосиланні при натиснутій клавіші **Shift**. Використовуючи панель задач, можна переключатись між різними задачами.

Панель інструментів Internet Explorer. Для управління роботою браузера зручно використовувати панель інструментів, яка включається командою **Вид=>Панель инструментов=>Обычные кнопки**. Опишемо дії, які виконуються за допомогою цих кнопок:

Назва кнопки	Дії, які здійснюються при натисканні кнопки
<i>Назад</i>	Повернутися до попередньої Web-сторінки
<i>Вперед</i>	Повернутися до наступної вже відкритої Web-сторінки
<i>Остановить</i>	Зупинити завантаження Web-сторінки
<i>Обновить</i>	Заново завантажити Web-сторінку (якщо її завантаження було перервано)
<i>Домой</i>	Перейти на домашню Web-сторінку
<i>Поиск</i>	Здійснити пошук Web-сторінки за допомогою пошукових служб
<i>Избранное</i>	Додати поточну Web-сторінку до списку сторінок, які регулярно відвідуються
<i>Журнал</i>	Перегляд протоколу Web-сторінок, які відкривалися в попередні дні, тижні

<i>Почта</i>	Відбувається завантаження програми для роботи з поштою (MS Outlook Express)
<i>Размер</i>	Зміна розміру шрифту поточної Web— сторінки

Для встановлення параметрів ефективної роботи програми Internet Explorer використовують команду головного меню **Сервіс=>Свойства обозревателя**. У вікні **Свойства обозревателя** є шість закладок, на яких можна встановлювати необхідні параметри.

Проблема читання Web-документів. Існують таблиці кодів для символічної інформації. Internet Explorer у більшості випадків автоматично вибирає правильну відповідність між кодами документа при його формуванні та його відображенні. Але іноді текст у вікні броузера неможливо прочитати. Вибрати потрібне кодування символів можна за допомогою команди головного меню **Вид=>Вид кодировки**.

Якщо це не допомагає можна спробувати знайти на Web документі гіперпосилання, яке дозволяє прочитати документ.

Єдиним ефективним засобом пошуку інформації є пошукові сервери. Пошукові системи можна розділити на два класи класифікаційні і словникові. У класифікаторах пошук здійснюється шляхом послідовного уточнення тематичних рубрик, поступово звужуючи пошук інтересів. Словникові пошукові системи постійно сканують Інтернет і заносять у свої довідники інформацію про те, що є в Інтернеті. Словникові пошукові системи здійснюють пошук інформації згідно із запитом користувача. Запит здійснюється шляхом завдання одного чи кількох ключових слів. Якщо ключовий елемент повинен бути присутній на Web-сторінках, то перед ним ставлять "+", якщо обов'язково відсутній то символ "-". Наприклад, + Windows-95 - означає пошук Web-сторінок, що обов'язково містять Windows, але не містять 95. Опис правил складання запитів є у кожній пошуковій системі в розділі Help.

Наведемо огляд пошукових машин, якими часто користуються:

а) еталонні американські системи: Yahoo (<http://www.yahoo.com>), Lycos (<http://www.lycos.com>) HotBot (<http://www.hotbot.com>)

б) російські системи: Рамблер (<http://www.rambler.ru>), Лист (<http://www.list.com>) Гугл (<http://www.google.com>)

в) українські системи: Мета (<http://meta-ukraine.com>), Пошук (<http://poshuk.dnepr.ua>).

5. Робота з FTP-сервером. Як відомо, Web-сторінки можуть містити посилання не тільки на HTML-документи, GIF-, JPEG-файли, які браузер Internet Explorer відкриває самостійно, але і файли інших типів - архіви, програмні файли, які браузер не може відкрити. У цьому випадку браузер починає процес завантаження файлів на робочий комп'ютер; при цьому відкривається вікно майстра завантаження файлів, в якому вказуємо на необхідність зберегти цей файл на диску і далі побачимо хід завантаження цього файлу.

Завантаження файлів можна здійснювати через введення в адресний рядок браузера адреси FTP-сервера, наприклад <ftp://ftp.mcd.cv.ua>. Щоб здійснити пошук FTP-серверів, що містять даний файл у браузер завантажують Web-сторінку спеціалізованої пошукової системи (наприклад <http://www.files.ru>).

У полі запити вводять ім'я шуканого файлу й натискають кнопку пошуку. Через деякий час у вікно браузера завантажиться Web сторінка з результатами пошуку (FTP-сервери, що містять потрібний файл).

6. Додаток Microsoft Outlook Express забезпечує всі необхідні засоби для роботи з електронною поштою і групами новин.

Після того як програма Outlook Express завантажить повідомлення чи після натискання кнопки **Доставить почту** на панелі інструментів, повідомлення можна читати або в області перегляду, або в окремому вікні.

1. Клацніть значок папки **Входящие** на панелі Outlook чи в списку папок.
2. Для перегляду повідомлення в області перегляду клацніть його в списку повідомлень.
3. Для перегляду повідомлення в окремому вікні двічі клацніть його в списку повідомлень.

Для перегляду всіх відомостей про повідомлення з групи новин (автор, дата відправлення і т.п.) виберіть команду **Свойства** в меню **Файл**.

Для збереження повідомлення на локальному диску виберіть команду **Сохранить как** і вкажіть місце і формат (поштове повідомлення, простий текст, гіпертекст).

Для відправлення повідомлення по електронній пошті:

1. На панелі інструментів натисніть кнопку **Создать сообщение**.
2. У поле **Кому и/или Копия** введіть адреси електронної пошти всіх одержувачів, розділяючи їх комами чи крапками з комою (;). Щоб вибрати одержувачів з адресної книги, клацніть значок із книгою, що знаходиться поруч з полями **Кому**, **Копия** у вікні створюваного повідомлення, і виберіть потрібні адреси.
3. У поле **Тема** вкажіть тему повідомлення.
4. Введіть текст повідомлення і натисніть кнопку **Отправить**.

Якщо на комп'ютері налаштовано кілька облікових записів, і потрібно використовувати обліковий запис, який не обирається за замовчуванням, клацніть поле **Вид** і виберіть потрібний обліковий запис.

При роботі в автономному режимі нове повідомлення зберігається в папці **Входящие**. При першому ж підключенні до мережі воно буде відправлено.

Щоб зберегти незавершене повідомлення, виберіть команду **Сохранить** в меню **Файл**. Можна також вибрати **Сохранить как**, щоб вказати формат, у якому треба зберегти повідомлення: поштове повідомлення (.eml), простий текст (.txt) чи гіпертекст (.htm).

Використовуючи бланки, що є в програмі Outlook Express, можна створювати привабливі поштові повідомлення і повідомлення в групі новин.

Бланк - це шаблон, до складу якого може входити фоновий малюнок, оригінальні кольори для тексту повідомлення, розмір полів і т.п. Щоб використати бланк до всіх вихідних повідомлень:

1. Виберіть **Параметры** в меню **Сервис** і відкрийте вкладку **Создание сообщения**.
2. У групі **Бланк** встановіть прапорець **Почта** чи **Новости** (або обидва ці прапорці), а потім натисніть кнопку **Выбрать**.

Щоб використовувати бланк до окремого повідомлення, у меню **Сообщения** вкажіть на пункт **Создание с использованием**, а потім виберіть бланк із необхідним стилем.

Хід роботи

1. Запустити програму Internet Explorer.
2. Зайти на домашню сторінку Чернівецького національного університету. - В адресному рядку ввести URL-адресу <http://www.chnu.cv.ua>.
3. Використовуючи гіперпосилання, перегляньте інші Web-сторінки цього сайту.
4. Додайте посилання на Web-сторінку про коледж до папки **Избранное**.
5. Збережіть Web-сторінку про коледж в окремому файлі. - Використайте команду **Файл=>Сохранить как**.
6. Визначте властивості Web-сторінки. - Виконайте команду **Свойства** з контекстного меню фона Web-сторінки.
7. Перегляньте властивості рисунка з Web -сторінки. – Команда **Свойства** з контекстного меню рисунка.
8. Збережіть на диску в окремому файлі тільки малюнок з Web сторінки. -- Виконайте команду **Сохранить рисунок как** з контекстного меню рисунка. (Можна зберегти і фоновий малюнок Web-сторінки за допомогою відповідної команди контекстного меню фонового малюнка).
9. Зробіть на своєму комп'ютері початковою сторінкою Web сторінку нашого університету. - Для цього завантажте домашню сторінку Web-сайту www.chnu.cv.ua, а потім користайтесь командою **Сервис=>Свойства обозревателя=>вкладка Общие=>С текущей**.
10. Ознайомтесь з параметрами програми Internet Explorer (команда меню **Сервис=>Свойства обозревателя**).
11. Користуючись пошуковою системою Internet, знайдіть таку інформацію:
 - Відомості про Ніклауса Вірта;
 - Відомості про м.Чернівці;
 - Реферат на тему “Топологія та класифікація комп’ютерних мереж”
12. Відшукайте в Internet URL - адреси сайтів навчальних закладів м.Чернівці.
13. Запустити програму Outlook Express.
14. Створити новий обліковий запис, ввівши своє ім'я, електронну адресу,

параметри серверів, пароль.

15. Створити повідомлення, де повідомлялося б про термінові збори акціонерів фірми.

16. Надати повідомленню високої важливості.

17. Відправити повідомлення списку акціонерів фірми, а копію - начальнику фірми.

18. Створити текстове повідомлення – привітання зі святом 8 березня.

Зберегти його в папці **Черновики**.

19. Змінити повідомлення з п.18, доповнивши його малюнками та оформленням. Адресувати його 3-м дівчатам.

20. Відправити всі повідомлення.

21. Перевірити пошту (одержати нові повідомлення).

22. Відсортувати повідомлення в папці **Входящие** за датою прийому.

23. Знищити деякі повідомлення.

24. Очистити папку **Удаленные**.

25. Закрити програму Outlook Express.

Контрольні питання

1. Що таке Internet?

2. Що таке протокол?

3. Які функції виконує протокол TCP/IP?

4. Яка структура IP-адреси?

5. Яка структура доменної адреси?

6. Які види послуг надає мережа Internet?

7. Що таке WWW, FTP?

8. Що таке Web-сторінка, Web-сервер?

9. Що таке броузер? Які броузери Ви знаєте?

10. Що таке гіперпосилання, який вони мають вигляд?

11. Як здійснюється перегляд Web-сторінок?

12. Яке призначення мають кнопки з панелі інструментів програми Internet Explorer?

13. Як встановити параметри програми Internet Explorer?

14. Яку роль виконують пошукові сервери? Які ви знаєте пошукові сервери?

15. Як здійснити пошук файлів в Internet?

16. Що таке обліковий запис?

17. Які можливості налагодження середовища програми Outlook Express вам відомі?

18. Які додаткові можливості (крім роботи з електронною поштою) має програма Outlook Express?

19. Які програми використовуються для роботи з електронною поштою ?

РОЗДІЛ 5. ОСНОВИ АЛГОРИТМІЗАЦІЇ ТА ПРОГРАМУВАННЯ

Лабораторна робота № 20. Тема: Робота в системі Turbo Pascal Робота з екраном

Методичні рекомендації

1. Як почати роботу в системі Turbo Pascal.

Система Turbo Pascal на твердому диску зазвичай міститься в каталозі з ім'ям TP (чи PAS, TURBOPAS, PASCAL і т.п.). Для виклику системи Turbo Pascal необхідно відшукати в деревоподібній структурі каталогів ПК цей каталог і в ньому файл TURBO.EXE. Цей файл містить готову до роботи діалогову систему програмування Turbo Pascal. У нього входять мінімально необхідні частини Turbo Pascal (текстовий редактор, компілятор, компоновщик, завантажник). Для нормальної роботи в діалоговому середовищі знадобляться також основна бібліотека, що розташована у файлі TURBO.TPL, і довідкова служба (файл TURBO.HLP). Цих файлів досить для написання, компіляції і виконання прикладів.

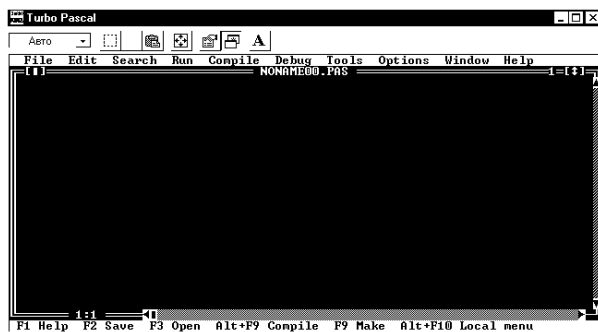


Рис. 21. Вид екрана після виклику системи Turbo Pascal

Після успішного виклику системи екран ПК здобуває вигляд, показаний на рис. 1. Для виходу з системи варто натиснути **Alt+X**.

Верхній рядок містить "меню" можливих режимів роботи Turbo Pascal, нижня - коротку довідку про

призначення основних функціональних клавіш. Вся інша частина екрана належить вікну редактора, обкресленому подвійною рамкою і призначеному для введення і корекції текстів програм. У його верхньому рядку приводяться ім'я того дискового файлу, звідки був прочитаний текст програми (новому файлу привласнюється ім'я *NONAME00.PAS*), два спеціальних поля, які використовуються при роботі з маніпулятором "миша" (ці поля виділені квадратними дужками), і цифра 1 - номер вікна. У Turbo Pascal можна працювати одночасно з декількома програмами (чи частинами однієї великої програми), кожна з яких може розташовуватися в

окремому вікні редактора. Середовище дозволяє використовувати до 9-ти вікон редактора одночасно.

Крім вікна (вікон) редактора в Turbo Pascal використовуються також вікна режиму відладки, виводу результатів роботи програми, довідкової служби, стека, регістрів. За бажанням вони можуть викликатися на екран по черзі чи бути присутніми на ньому одночасно.

2. Призначення функціональних клавіш в системі Turbo Pascal.

Функціональні клавіші використовуються для керування середовищем Turbo Pascal. З кожної з цих клавіш зв'язується деяка команда меню. Дія майже усіх функціональних клавіш можна модифікувати трьома особливими клавішами: **Alt**, **Ctrl** і **Shift**. Нижче приводяться команди, що передаються середовищу Turbo Pascal функціональними клавішами і деякими їх комбінаціями з клавішами **Ctrl** і **Alt**:

F1 - звернутися за довідкою до вбудованої довідкової служби (Help-допомога);

F2 - записати текст, що редагується, у дисковий файл;

F3 - прочитати текст із дискового файлу у вікно редактора;

F4 - використовується в режимі відладки: почати чи продовжити виконання програми і зупинитися перед виконанням того її рядка, на якому стоїть курсор;

F5 - розгорнути активне вікно на весь екран;

F6 - зробити активним наступне вікно;

F7 - використовується в режимі відладки: виконати наступний рядок програми; якщо в рядку є звертання до процедури (функції), ввійти в цю процедуру і зупинитися перед виконанням першого її оператора;

F8 - використовується в режимі відладки: виконати наступний рядок програми; якщо в рядку є звертання до процедури (функції), виконати її і не просліджувати її роботу;

F9 - компілювати програму, але не виконувати її;

F10 - перейти до діалогового вибору режиму роботи за допомогою головного меню;

Ctrl+F9 - виконати прогін програми: компілювати програму, що знаходиться в редакторі, завантажити її в оперативну пам'ять і виконати, після чого повернутися в середовище Turbo Pascal.

Alt+F5 - змінити вікно редактора на вікно виводу результатів роботи (прогону) програми.

3. Текстовий редактор середовища Turbo Pascal.

Текстовий редактор Turbo Pascal надає користувачу зручні засоби створення і редагування текстів програм. Ознакою того, що середовище знаходиться в стані редагування, є наявність у вікні редактора курсору - невеликого миготливого прямокутника. Якщо середовище знаходиться в стані вибору з меню, курсор зникає, а в рядку меню з'являється кольоровий покажчик-прямокутник, який виділяє одне з кодових слів (опцій меню). Для переходу від стану вибору режиму з головного меню в стан редагування

потрібно натиснути клавішу **Esc**, а для переходу до вибору з головного меню - **F10**.

Для створення тексту програми потрібно ввести цей текст за допомогою клавіатури ПК. Після заповнення чергового рядка варто натиснути на клавішу **Enter**, щоб перевести курсор на наступну рядок (курсор завжди показує те місце на екрані, куди буде поміщений черговий символ програми, що вводиться.).

Вікно редактора імітує довгий і досить широкий лист папера, фрагмент якого видний у вікні. Якщо курсор досяг нижнього краю, здійснюється прокручування вікна редактора: його вміст зміщується нагору на один рядок і знизу з'являється новий рядок листа. Якщо курсор досяг правої границі екрана, вікно починає в міру введення символів зміщатися вправо, показуючи правий край листа. Розміри листа по горизонталі і вертикалі обмежуються тільки загальним числом символів у файлі, яких не повинно бути більше 64535, однак компілятор Turbo Pascal сприймає рядка програми довжиною не більш 126 символів.

Вікно можна зміщувати щодо листа за допомогою наступних клавіш:

PageUp - на сторінку нагору;

PageDown - на сторінку вниз;

Home - у початок поточної рядка;

End - у кінець поточного рядка;

Ctrl+PageUp -у початок тексту;

Ctrl+PageDown - у кінець тексту.

Якщо Ви помилилися при введенні чергового символу, його можна стерти за допомогою клавіші зі стрілкою (чи написом **Backspace**), розташованої над клавішею **Enter**. Клавіша **Delete** стирає символ, на який у даний момент вказує курсор, а команда **Ctrl+Y**- весь рядок, на якому розташований курсор.

Редактор Turbo Pascal вставляє наприкінці кожного рядка невидимий символ-роздільник. Цей символ вставляється клавішею **Enter**, а стирається клавішами **Backspace** чи **Delete**.

Нормальний режим роботи редактора - режим вставки, у якому кожний символ, що знову вводиться, як би "розсовує" текст на екрані, зміщуючи вправо залишок рядка. Редактор може також працювати в режимі накладення нових символів на існуючий старий текст: у цьому режимі новий символ заміняє собою той символ, на який вказує курсор, а залишок рядка не зміщується вправо. Для переходу до режиму накладення потрібно натиснути клавішу **Insert**, а якщо натиснути цю клавішу ще раз, знову встановлюється режим вставки. Ознакою того, у якому режимі працює редактор, є форма курсору: у режимі вставки він схожий на миготливий символ підкреслення, а в режимі накладення він являє собою великий миготливий прямокутник.

4. Робота з файлами

Основною формою збереження текстів програм поза середовищем є файли. Після завершення роботи з Turbo Pascal можна зберегти текст нової програми в дисковому файлі для того, щоб використовувати його наступного разу. Для обміну даними між дисковими файлами і редактором середовища призначені клавіші **F2** (запис у файл) і **F3** (читання з файлу). Якщо Ви створюєте нову програму, то середовище привласнює їй стандартне ім'я *NONAME00.PAS*. Для збереження тексту програми у файлі потрібно натиснути **F2**. У цей момент середовище перевірить ім'я програми і, якщо це стандартне ім'я *NONAME*, запитатиме, чи потрібно його змінювати: на екрані з'явиться невелике вікно запиту з написом (рис.22):

Нижче наведено напису розташовується поле для введення імені файлу, у якому можна написати потрібне ім'я і натиснути **Enter** - текст буде збережений у файлі. Якщо в імені опущено розширення, середовище привласнить файлу стандартне розширення *PAS*. Якщо Ви захочете завершити роботу з системою, а в редакторі залишився не

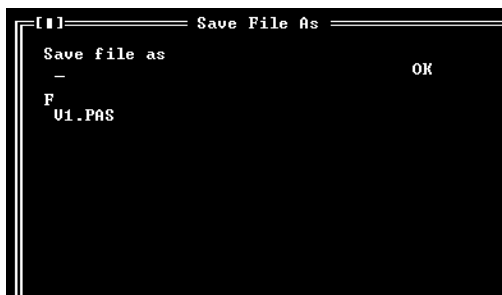


Рис.22. Запит на збереження файлу з

збережений у файлі текст, на екрані з'явиться вікно з запитом на збереження модифікованого (зміненого) файлу. У відповідь варто натиснути **Y**(Yes - так), якщо необхідно зберегти текст у файлі, чи **N**(No - немає), якщо зберегти текст не потрібно.

5. “Прогон” та відладка програми

Після підготовки тексту програми можна спробувати виконати її, тобто відкомпілювати програму, зв'язати її (якщо це необхідно) з бібліотекою стандартних процедур і функцій, завантажити в оперативну пам'ять і передати їй керування. Уся ця послідовність дій називається “прогоном” програми і реалізується командою **Ctrl+F9**.

Якщо в програмі немає синтаксичних помилок, то всі дії виконуються послідовно одна за другою, при цьому в невеликому вікні повідомляється про кількість відкомпільованих рядків і обсяг доступної оперативної пам'яті. Перед передачею керування завантаженої програмі середовище виводить на екран вікно прогону програми, а після завершення роботи програми знову бере керування комп'ютером на себе і відновлює на екрані вікно редактора.

Якщо на якому-небудь етапі середовище знайде помилку, воно припиняє подальші дії, відновлює вікно редактора і поміщає курсор на той рядок програми, при компіляції чи виконанні якого виявлена помилка. При цьому у верхньому рядку редактора з'являється діагностичне повідомлення про причину помилки. Усе це дозволяє дуже швидко налагодити програму,

тобто усунути в ній синтаксичні помилки і переконатися в правильності її роботи. У деяких ситуаціях зазвичай використовують покрокове виконання програми за допомогою команд, зв'язаних із клавішами **F4**, **F7** і **F8**.

6. Вивід та розміщення даних на екрані.

У мові Паскаль виведення даних на екран комп'ютера здійснюється за допомогою вбудованої процедури *Write*, яка має такий формат:

```
Write(u, v, z);
```

де аргументи *u*, *v*, *z* - це ті змінні, значення яких потрібно вивести.

Якщо потрібно дані вивести на принтер, то перед списком аргументів в операторі *Write* записують слово *List*. Наприклад, оператор

```
Write(List, 'Сума = ');
```

виведе на принтер текст *Сума =* . Для виведення кожної змінної в новий рядок використовують оператор *Writeln*. Керувати виведенням змінних можна ще і використовуючи формати виведення.

7. Використання додаткових модулів.

Для забезпечення управління текстовим режимом роботи екрану використовують стандартний модуль *CRT*. В ньому розміщені процедури та функції, що керують переміщенням курсору в довільну позицію на екран, змінюють колір символів, які виводяться та фону, створюють вікна. Крім того, в модуль включені також процедури «сліпого» читання клавіатури та управління звуком. Підключення модулю здійснюється в розділі *Uses* наступною командою:

```
Uses Crt;
```

В програмі можна використовувати наступні процедури:

```
ClrScr; {очищає вікно екрану}
```

```
GoToXY(x,y: byte); {переміщує курсор в точку екрану з координатами (x,y)}
```

```
Procedure TextBackground (Color : Byte); {задає колір фону екрану}
```

```
Procedure TextColor (Color : Byte); {задає колір символів повідомлення, що виводиться на екран}
```

Константи фонового та основного кольорів

Black	= 0;	{ чорний }
Blue	= 1;	{ синій }
Green	= 2;	{ зелений }
Cyan	= 3;	{ бірюзовий }
Red	= 4;	{ червоний }
Magenta	= 5;	{ малиновий }
Brown	= 6;	{ коричневий }
LightGray	= 7;	{ світло-сірий }
DarkGray	= 8;	{ темно-сірий }
LightBlue	= 9;	{ світло-блакитний }
LightGreen	= 10;	{ світло-зелений }
LightCyan	= 11;	{ світло-бірюзовий }
LightRed	= 12;	{ світло-червоний }
LightMagenta	= 13;	{ світло-малиновий }
Yellow	= 14;	{ жовтий }

```
White          = 15;    { білий }  
Blink          = 128;   { миготіння }
```

Завдання 1.

Перевірте виконання наступної програми:

```
Program P1;  
Uses CRT;  
Begin  
  ClrScr;  
  TextBackGround (7);  
  TextColor (14);  
  GotoXY (32,10) ;  
  Write ('*****');  
  GotoXY (32,11);  
  TextColor (9+Blink);  
  Write ('* Вітаю !!! *');  
  GotoXY (32,12);  
  TextColor (3);  
  Write (' * з першою програмою *');  
  GotoXY (32,13);  
  Write (' * на мові Паскаль !!! *');  
  GotoXY (32,14);  
  TextColor (15);  
  Write ('*****');  
End.
```

Завдання 2.

Скласти програму, яка б виводила на екран різними кольорами Ваше прізвище, ім'я та по батькові.

Лабораторна робота № 21-22. Робота в системі Turbo Pascal Типи даних.

Методичні рекомендації

1. Типи даних та їх описи.

Будь-який елемент даних (змінну) необхідно в програмі описати типом змінних. Тип визначає ту множину значень, які можуть приймати змінні і операції, які можна застосовувати до цих змінних. Крім цього, тип визначає формат представлення даних в пам'яті комп'ютера.

Pascal характеризується розгалуженою структурою типів даних, до яких належать:

- 1) прості (цілі, дійсні, символьні, перераховуванні, діапазон):
- 2) структуровані (масиви, записи, файли та ін.).

Прості типи даних

Цілі дані представляються цілими числами зі знаком і без знаку. В залежності від діапазону значень цілі дані можуть бути таких типів:

integer - представляє значення цілих в діапазоні -32768 до 32767 (2 байти),

byte - числа в діапазоні від 0 до 255 (1 байт),

word - числа в діапазоні від 0 до 65535 (2 байти).

Наведемо приклад опису цілочислових змінних:

```
var H1, H2: integer;
```

Над цілими операндами можна виконувати наступні арифметичні операції: додавання, віднімання, множення, ділення. Знаки цих операцій: +, -, *, div.

Результат арифметичних операцій над цілими є величиною цілого типу. Результат виконання операції ділення цілих величин є ціла частина частки, наприклад, $17 \text{ div } 2 = 8$, $3 \text{ div } 5 = 0$.

До цілочислових даних можна застосувати і стандартні (вбудовані) функції Паскаля: *SIN(x)*, *COS(x)*, *LN(x)*, *SQRT(x)*, але результат уже буде дійсним числом.

Дійсні типи. В арифметичних виразах звичайно використовуються змінні, які набувають дійсних значень. Всі дійсні числа можуть зображатися у формі з фіксованою крапкою (наприклад, 0.13, 641.27), із плаваючою крапкою (наприклад, $0.13 \cdot 10^{-7}$). Значення з плаваючою крапкою у мові Pascal записуються у форматі <мантиса>E<порядок>, наприклад, 0.13E-7.

Для представлення дійсних значень у Pascal частіше всього використовуються типи *real* та ін. Ці типи розрізняються діапазоном допустимих значень і обсягом необхідної пам'яті:

real - числа від 2.9E-39 до 1.7E38 з мантисою 11-12 десяткових знаків; надається 6 байт пам'яті під змінну цього типу. Формат опису дійсних типів має такий вигляд

```
var X: real;
```

Символьний тип. Змінні, які набувають символьних значень з таблиці ASCII належать до символьного типу *char*.

Формат опису символьних змінних має вигляд

```
var WX: char;
```

тоді значення символьних змінних з програми беруть в лапки: *WX* = 'B'.

Логічні дані приймають два значення: true (істинно), false (хибно) і можуть бути типів *boolean* (1 байт) та ін. Опис цих величин у програмі має вигляд:

```
var P: boolean;
```

Перелічуваний тип даних визначається шляхом перерахування елементів за такою схемою:

```
type <ім 'я типу'> = <(список імен)>
```

Наприклад,

```
type Notation = (do, re, mi, fa, sol, la, si);
```

Тоді при описі змінних типу MyNota використовується тип Notation:

```
var MyNota: Notation;
```

і в програмі змінна MyNota може набувати лише значень з переліченого ряду.

Тип діапазон. У деяких випадках зручніше не називати всі значення змінної, а просто зазначити межі інтервалу, у якому вони містяться. Для цього застосовується інтервальний тип даних (тип діапазон), в описі якого вказується інтервал від найменшого до найбільшого значення, що розділені двома крапками, наприклад,

```
type E1 = 100..200;
```

Тоді опис змінних цього типу має вигляд `var N1, N2: E1;`

Структуровані типи даних

Усі розглянуті вище типи (цілий, дійсний, символьний, логічний, перелічуваний, тип діапазон) називаються простими типами. Поряд з простими типами у Паскалі передбачені структуровані типи, в яких дані складаються з компонентів. До структурованих типів належать масиви, записи, файли, рядки та ін.

2. Читання в Паскалі. Процедури читання Read, Readln

Для введення даних в програму передбачена процедура *Read*, формат якої: *Read(A, B, C);*

Дані набираються на клавіатурі, хоча б через один пробіл, в кінці набору натискається Enter. Дані, що вводяться, мають відповідати визначеному для них типу. Якщо ця відповідність порушується, то буде виведено повідомлення про помилку. Дані для оператора *Read* набираються в одному рядку.

Однак у Паскалі передбачене зчитування даних з окремих рядків за допомогою процедури *Readln*. Кожен оператор *Readln* починає зчитування з нового рядка на відміну від операторів *Read*, дані для яких набираються в одному рядку.

3. Складання найпростіших алгоритмів виконання математичних операцій

Приклад:

```
Program Prima;
Uses crt;
Var a,b,c:Integer;
    y:Real;
Begin
  ClrScr;
  a:=5; b:=1; c:=1;
  y=(b+c)/((c-a)*b);
  Write (y:2:5, ' = (' , b, ' + ' , c, ') / ((' , c, '- ' , a,
    ' ) * ' , b, ' ) ' );
End.
```

Підключення стандартного модуля процедур керування

Блок опису типу змінних

Блок присвоєння значень змінним та виконання математичних обчислень

Виведення результатів на екран

Процедури обчислення математичних функцій

Запис	Функція
Abs (x) ;	Обчислення модуля числа
Arctan (x) ;	Обчислення арктангенса числа
Cos (x) ;	Обчислення косинуса числа
A div B;	Цілочисельне ділення
Exp (x) ;	Обчислення експоненти числа
Ln (x) ;	Обчислення натурального логарифму числа
A mod B;	Залишок від ділення чисел
Sin (x) ;	Обчислення синусу числа
Sqr (x) ;	Обчислення квадрату числа
Sqrt (x) ;	Обчислення кореня квадратного числа

Завдання 1.

Скласти програму, яка б додавала, віднімала, множила та ділила два числа. Передбачити вивід результату на екран.

Завдання 2.

Скласти програму, яка у діалоговому режимі запропонувала Вам обчислити задану математичну функцію.

1. $y = \ln ab + c \sin^2 d + 0.5$ при $a=6.8$; $b=0.75$; $c=12$; $d=342$;
 2. $y = a^2 + b^2 \cos^2 c + d$ при $a=14$; $b=7.75$; $c=0.8$; $d=-3.65$;
 3. $y = b^2 + c^2 + 1 + 2.5 \cdot \ln ad$ при $a=15.7$; $b=2$; $c=0.85$; $d=7.75$;
 4. $y = a^2 + b^2 + c + 3d$ при $a=6$; $b=1.5$; $c=0.85$; $d=0.75$;
 5. $y = (a^2 + c^2)(\operatorname{tg} d + 3.3b)$ при $a=1.2$; $b=18$; $c=0.725$; $d=15.4$;
 6. $y = (a^2 - 1)/(b+1) - (c-2)/\cos(d+1.5)$ при $a=3.4$; $b=25$; $c=70.8$; $d=0.57$;
 7. $y = \operatorname{arctg}(a/d) + (b + d)^2$ при $a=75$; $b=4.5$; $c=6.2$; $d=5$;
 8. $y = \sin a \cdot \ln dc + (b - 1)^2 / 3$ при $a=5.5$; $b=-3.6$; $c=63$; $d=0.54$;
 9. $y = b^2 + \ln c + (d^2 - 1)/2a$ при $a=-0.05$; $b=3.7$; $c=48$;
- $d=15$;
10. $y = \cos(b/a) + 2ad + (c - 1)^3$ при $a=0.75$; $b=7.7$; $c=-2.83$; $d=45$;
 11. $y = 3d^2 + b \cdot \sin(2/(c-d))$ при $a=2.89$; $b=18$; $c=3.75$; $d=-0.29$;
 12. $y = (2a + b)^2 + c^3/(d - 2)$ при $a=0.83$; $b=4.2$; $c=-6.7$; $d=-1.5$

Завдання 3.

1. Складіть програму для обчислення довжини гіпотенузи прямокутного трикутника, якщо відомі довжини його катетів.

2. В одному акваріумі А рибок, а в іншому - на D рибок більше. Складіть програму для обчислення кількості рибок у двох акваріумах разом.

3. Два пішоходи вийшли одночасно в одному напрямку з пунктів А і Б. Перший - зі швидкістю V_1 км/год, а другий V_2 км/год. Складіть програму для визначення відстані між пішоходами через три години, якщо відстань між пунктами А і Б - 50 км.

4. В одному будинку D квартир, а в іншому - у N разів менше. Скласти програму для визначення кількості квартир у двох будинках разом.

5. Два велосипедисти виїхали одночасно в різних напрямках з пункту N. Перший - зі швидкістю A км/год, а другий - B км/год. Складіть програму для визначення відстані між ними через три години, якщо другий велосипедист останні 20 хв. їхав у зворотному напрямку.

6. В одному класі M хлопчиків, а в іншому - у D разів більше. Скласти програму для знаходження кількості хлопчиків у двох класах разом.

7. Два автомобілі виїхали одночасно в різних напрямках з пункту A. Перший - зі швидкістю V_1 км/год., а другий - V_2 км/год. Складіть програму для визначення відстані між ними через дві години, якщо перший зробив дві зупинки по 20 хв., а другий - одну десятихвилинну зупинку.

8. В одній роті R солдатів, а в двох разом - S солдатів. Скласти програму для визначення, на скільки солдатів у другій роті більше, ніж у першій.

9. Два автобуси виїхали одночасно в одному напрямку з пункту N. Перший - зі швидкістю A_1 км/год., а другий - A_2 км/год. Складіть програму для визначення відстані між ними через п'ять годин, якщо перший зробив дві зупинки по 15 хв., а другий шість зупинок по 10 хв.

10. Куплено 10 мішків капусти. В одному – M_1 кг, у другому - M_2 кг M_3 г, а у всіх інших - по M_4 кг M_5 г. Складіть програму, що визначає, скільки куплено тонн капусти.

11. У бензобаку автомобіля було V літрів бензину. Після того, як автомобіль тричі з'їздив у пункт A і назад, у бензобаку залишилося M см^3 бензину. Складіть програму, яка б визначала відстань до пункту A, якщо витрата бензину складає U літрів на кожні 100 км.

12. Хлопчик поклав у коробку стільки кубиків, скільки їх там вже було, і ще D штук. Після цього в коробці виявилось N кубиків. У скільки разів більше кубиків поклав хлопчик у коробку другого разу, ніж першого?

Завдання 4.

1. Підприємець виготовив K_1 одиниць товару першого виду і K_2 одиниць товару другого виду собівартістю по D_1 і D_2 доларів за тисячу одиниць. Від реалізації виготовленого товару він одержав відповідно по P_1 і P_2 гривень за кожну одиницю. Скласти програму для визначення прибутку підприємця, якщо відомо, що U одиниць товару першого виду виявилися бракованими, а курс долара в ці дні складав G гривень S коп.

2. У продавця N ящиків, у кожному з яких по B кг бананів. Визначити, який штраф він сплатить комітету із захисту прав споживачів, якщо він продає банани по C гривень за кілограм і при цьому обважує покупців на G г при зважуванні кожного кілограма, а сума штрафу дорівнює п'ятикратному розміру прибутку, отриманого за рахунок обважування покупців. Складіть програму для визначення результату задачі.

3. Магазин одержав 30 рулонів тканини. 5 рулонів довжиною по T м, 3 рулони - по $T1$ м 20 см, а всі інші - по $T2$ м $T3$ дм. Складіть програму, що визначає, скільки усього метрів тканини одержав магазин.

4. Після того, як автобус виконав сім рейсів по своєму маршруту туди і назад, у нього у бензобаку залишилося M літрів бензину. Визначити, скільки бензину було в бензобаку перед початком першого рейсу, якщо довжина маршруту від однієї кінцевої зупинки до іншої R км, а витрата бензину складає U дм^3 на кожні 10 км. Складіть програму для розв'язування цієї задачі.

5. На грядці росли тюльпани. Спочатку зрізали K тюльпанів, а потім п'яту частину всіх, що залишилися. Складіть програму для визначення, скільки тюльпанів росло на грядці, якщо залишилося рости T штук.

6. Продавець одержав макарони за оптовою ціною C гривень за кілограм і повинен продати їх за роздрібною ціною R гривень за кілограм. Макарони були отримані в N коробках. У кожній коробці по U упакувань з макаронами. Вага одного упакування V кг. Однак виявилось, що під час транспортування з кожної коробки зникли по Z упакувань. Визначити, по якій мінімальній ціні (за кілограм) продавець повинен реалізувати макарони, щоб не допустити збитку. Складіть програму для визначення результату задачі.

7. Турист пройшов 8 маршрутів. Три маршрути довжиною по R км, K маршрутів - по $R1$ км 800 м, а інші по $R3$ км $R2$ м. Складіть програму за допомогою якого можна визначити, яку відстань пройшов турист по всіх маршрутах разом.

8. Автомобіль виконав чотири рейси з овочевої бази в магазин. При цьому він перевозив щоразу по M мішків з картоплею. У результаті в магазині виявилось T тонн картоплі. Визначити місткість одного мішка (у кілограмах), якщо спочатку в магазині залишалося K кг картоплі, а за час перевезень магазин продав P кг. Складіть програму для розв'язання цієї задачі.

9. У зоопарк привезли B нових мавп. Після того, як через 2 дні третю частину всіх мавп зоопарку перевели в літні вольєри, виявилось, що в літніх вольєрах мавп у K разів більше, ніж їх було у всьому зоопарку до надходження нових. Складіть програму для визначення, скільки в зоопарку було мавп до надходження нових.

10. Студентська їдальня ранком спекла A тістечок. Після того, як на другій перерві продали сьому частину всіх тістечок, що залишилися від продажу на першій перерві, виявилось, що це в N разів більше, ніж було продано на першій перерві. Складіть програму для визначення, скільки тістечок було продано на першій перерві.

Лабораторна робота № 23-24. Програми з розгалуженою структурою

Методичні рекомендації

1. Умовний оператор IF.

Оператор (інструкцію) if можна описати двома способами:

Варіант 1

```
if Умова
then
    begin
        {ці інструкції виконуються,
        якщо Умова істинна}
    end
else
    begin
        {ці інструкції виконуються,
        якщо Умова хибна}
    end;
```

Варіант 2

```
if Умова
then
    begin
        {ці інструкції виконуються,
        якщо Умова істинна}
    end;
```

Наприклад:

```
if a>b then writeln ('a більше b')
    else writeln ('a менше або рівне b');
```

або

```
if (a>=0) and (a<10) then writeln ('число a -
однозначне');
```

2. Оператор CASE.

Інструкція case має вигляд:

```
case вираз-селектор of
    СписокКонстант1: begin
        {інструкції 1}
    end;
    СписокКонстант2: begin
        {інструкції 2}
```

```

end;
.....
СписокКонстантN: begin
    {інструкції N}
end
else
    begin
        {інструкції}
    end;
end;

```

Наприклад:

```

case i of
1..10: writeln ('Число ', i:4, ' в діапазоні 1 -
10');
11..20: writeln ('Число ', i:4, ' в діапазоні 11 -
20');
21..30: writeln ('Число ', i:4, ' в діапазоні 21 -
30')
else writeln ('Число поза діапазоном')
end;

```

Завдання 1.

1. Скласти програму, яка б визначала стан води (лід, вода, пара) в залежності від її температури.
2. Скласти програму, яка б серед чисел x, y, z знаходила максимальне.
3. Скласти програму, яка б визначала скільки цифр міститься -у введеному Вами числі.
4. Скласти програму, яка б виводила на екран день тижня за вказаним Вами його порядковим номером.
5. Присвоїти змінній X число 230 у тому випадку, якщо сума значень змінних X і B менше їхнього добутку, і присвоїти зазначеній змінній число 1, якщо це не так.
6. Присвоїти змінній B число 3 у тому випадку, якщо значення змінної C менше 121.
7. Присвоїти змінній R число 20 у тому випадку, якщо різниця значень змінних K і A менше їхньої частки, і присвоїти зазначеній змінній число 0, якщо це не так.
8. Присвоїти змінній N число 100 у тому випадку, якщо значення змінної A в 2 рази більше значення змінної C.
9. Присвоїти змінній X суму чисел A і B у тому випадку, якщо значення змінної X не менше значення змінної A, і присвоїти тій же змінній різницю зазначених чисел, якщо це не так.

10. Присвоїти змінній F число 22 у тому випадку, якщо значення змінної F дорівнює 11.
11. Присвоїти змінній Q число 25 у тому випадку, якщо добуток значень змінних Y і Z менше їхньої різниці, і присвоїти зазначеній змінній число 250, якщо це не так.
12. Скласти програму, яка б за температурою визначала фізичний стан води.

Завдання 2.

Дані дійсні (тип Real) числа x_1 , x_2 , x_3 , x та y . Обчислити $F(x)$, якщо

$$1. \quad F(x) = \begin{cases} x_1 - x_2, & \text{при } x_3 < 5, \\ x_1 + x_2, & \text{при } x_3 = 5, \\ x_3, & \text{при } x_3 > 5; \end{cases}$$

$$2. \quad F(x) = \begin{cases} x_1 + 4x_2 - x_3, & \text{при } x_1 < 3, \\ x_1^2 + x_3, & \text{при } x_1 = 3, \\ x_2 + 8x_3, & \text{при } x_1 > 3; \end{cases}$$

$$3. \quad F(x) = \begin{cases} x^2 - x, & \text{при } x < 0, \\ 0, & \text{при } x = 0, \\ x^2 + 8x, & \text{при } x > 0; \end{cases}$$

$$4. \quad F(x) = \begin{cases} x - y, & \text{при } x < y, \\ 2x + \frac{y}{2}, & \text{при } x = y, \\ y - x + 1, & \text{при } x > y; \end{cases}$$

$$5. \quad F(x) = \begin{cases} x^2 + 4x + 5, & \text{при } x \leq 2, \\ \frac{1}{x^2 + 4x + 5}, & \text{в інших випадках,} \end{cases}$$

$$6. \quad F(x) = \begin{cases} 2x^2 - 3x + 1, & \text{при } x \leq 5, \\ \frac{1}{9x^2 + 7x - 4}, & \text{в інших випадках;} \end{cases}$$

$$7. \quad F(x) = \begin{cases} x^2 + 3x - 1, & \text{при } x \leq 3, \\ \frac{1 + 2x}{3 - 5x}, & \text{в інших випадках;} \end{cases}$$

$$8. \quad F(x) = \begin{cases} \frac{x^2 - 3x}{1 + 4x^2}, & \text{при } x \leq -2, \\ \frac{5 - 2x}{1 + x^2}, & \text{в інших випадках;} \end{cases}$$

$$9. \quad F(x) = \begin{cases} \frac{x^2}{2x-1}, & \text{при } x \leq -1, \\ \frac{3-4x}{x^2}, & \text{в інших випадках;} \end{cases}$$

$$10. \quad F(x) = \begin{cases} \frac{x^2-9}{x+3}, & \text{при } x = 2, \\ \frac{5-2x}{2+x}, & \text{в інших випадках;} \end{cases}$$

$$11. \quad F(x) = \begin{cases} \frac{x^2+3x+2}{x-3}, & \text{при } x \leq 4, \\ \frac{x^2-3x}{x^2-2x+7}, & \text{в інших випадках;} \end{cases}$$

$$12. \quad F(x) = \begin{cases} \sin x, & \text{при } x \leq 0, \\ \cos x, & \text{при } 0 < x \leq 1, \\ \operatorname{tg} x, & \text{в інших випадках;} \end{cases}$$

$$13. \quad F(x) = \begin{cases} \sin^2 x, & \text{при } x \leq 0, \\ \ln x - \sin x, & \text{при } 0 < x \leq 3, \\ \cos x^2, & \text{в інших випадках;} \end{cases}$$

$$14. \quad F(x) = \begin{cases} \sin 4x + \sin^2 2x, & \text{при } x \leq 90, \\ \sin 2x + 4 \cos x^2, & \text{при } 90 < x \leq 180, \\ \sin 3x - \cos x, & \text{в інших випадках;} \end{cases}$$

$$15. \quad F(x) = \begin{cases} \frac{3x-2}{5x+8}, & \text{при } x \leq 0, \\ (2x-3)(1-x^3), & \text{при } 0 < x \leq 1, \\ \frac{x^2}{2} + \frac{3}{x^3} + 1, & \text{в інших випадках;} \end{cases}$$

Завдання 3.

1. Скласти програму, яка б виконувала арифметичні операції над двома числами. Введення чисел та вибір математичної операції виконати в діалоговому режимі. У програмі передбачити уникнення помилок, наприклад ділення на нуль.
2. Скласти алгоритм визначення поведінки ракети в залежності від швидкості її запуску (якщо швидкість $V < 7.8$ км/с - ракета впаде на землю; якщо $V < 11.2$ км/с - ракета стане супутником Землі; якщо $V < 16.4$ км/с - ракета стане супутником Сонця; якщо $V > 16.4$ км/с - ракета вилетить за межі Сонячної системи).

3. Скласти програму яка б за координатами x та y , визначала якому квадранту належить точка $M(x;y)$.
4. Скласти програму яка б пропонувала Вам ввести число і потім виводила на екран кількість комп'ютерів (введене число + слово "комп'ютер" у відповідному відмінку).
5. Дані дійсні додатні числа a, b, c, d . Визначити чи можна прямокутник зі сторонами a і b розмістити всередині прямокутника c і d так, щоб кожна зі сторін одного прямокутника була паралельна або перпендикулярна кожній стороні іншого прямокутника.
6. Дані дійсні додатні числа a, b, c, x, y . Визначити, чи пройде цеглина з ребрами a, b, c в прямокутний отвір зі сторонами x і y таким чином, щоб кожне з її ребер було паралельне або перпендикулярне кожній з сторін отвору.
7. Задано довжини 3-х відрізків. Визначити чи можна з даних відрізків скласти трикутник. Якщо так, то визначити тип даного трикутника (звичайний, рівносторонній, рівнобедрений, прямокутний).
8. Задано довжини 3-х відрізків. Визначити чи можна з даних відрізків скласти трикутник. Якщо так, то визначити за формулою Герона його площу.

$$\text{Формула Герона: } S = \sqrt{p(p-a)(p-b)(p-c)}, \quad \text{де } p = \frac{a+b+c}{2}.$$

9. Задано довжини 3-х відрізків. Визначити чи можна з даних відрізків скласти трикутник. Якщо так, то визначити чи цей трикутник прямокутний (обчислити його площу. Теорема Піфагора: $c^2 = a^2 + b^2$,

Площа прямокутного трикутника: $S = \frac{1}{2} a \cdot b$ де a, b - катети прямокутного

трикутника, c -гіпотенуза.

10. Задано довжини 3-х відрізків. Визначити чи можна з даних відрізків скласти трикутник. Якщо так, то перевірити чи він рівнобедрений і обчислити його площу і висоту. Висота рівнобедреного трикутника:

$$h = \sqrt{b^2 - \left(\frac{a}{2}\right)^2}, \quad \text{Площа рівнобедреного трикутника: } S = \frac{1}{2} a \cdot h \quad \text{де } a, h -$$

відповідно основа та висота рівнобедреного трикутника;

11. Задані дійсні числа $x_1, y_1, x_2, y_2, x_3, y_3$ - Визначити чи належить початок координат трикутнику з вершинами $(x_1, y_1), (x_2, y_2), (x_3, y_3)$

12. Задані точки $A(x_1, y_1), B(x_2, y_2), C(x_3, y_3)$ і $O(x_4, y_4)$. Визначити, чи є точка O центром кола, яке проходить через точки A, B, C . Довжина відрізка обраховується за формулою: $AB = \sqrt{(X_B - X_A)^2 + (Y_B - Y_A)^2}$.

Завдання 4.

1. Задано координати клітини шахівниці $(x; y)$ - цілі числа. З'ясувати, чи збігається колір цієї клітини з кольором клітини з координатами $(1;1)$.

2. Територія, що охороняється, має вид правильного багатокутника зі стороною A метрів, у центрі якого прив'язана собака. Уся територія абсолютно порожня і собака може по ній вільно пересуватися. Довжина ланцюга C метрів, відстань від ошейника до передніх зубів собаки Z см. Складіть програму для визначення, чи є на цій території ділянки, що залишилися без охорони.
3. Задано координати клітини шахівниці $(x; y)$ - цілі числа. З'ясувати, чи перетинає дану клітку пряма, що проходить через центри клітин з координатами $(1; 1)$ і $(7; 1)$.
4. Відомий номер автомобіля (ціле п'ятицифрове число). Складіть програму для визначення, чи починається цей номер з трійки або з дев'ятки.
5. Задано координати клітини шахівниці $(x; y)$ - цілі числа. З'ясувати, чи перетинає дану клітку пряма, що проходить через центри клітин з координатами $(2; 1)$ і $(8; 7)$.
6. Відомий номер автомобіля (ціле п'ятицифрове число). Визначте, чи починається цей номер з нуля або вісімки.
7. Задано координати клітини шахівниці $(x; y)$ - цілі числа. З'ясувати, чи може шахова тура за один хід перейти з цієї клітини в клітину з координатами $(4; 6)$.
8. Відомий номер автомобіля (ціле п'ятицифрове число). Визначте, чи починається цей номер двійкою або п'ятіркою.
9. Задано координати клітини шахівниці $(x; y)$ - цілі числа. З'ясувати, чи може шаховий ферзь за один хід перейти з цієї клітини в клітину з координатами $(5; 7)$.
10. Напишіть програму для розв'язання квадратного рівняння.
11. Напишіть програму для калькулятора із застосуванням CASE.
12. Складіть програму для порівняння віку трьох дітей.

Лабораторна робота № 25-26. Організація циклів.

Методичні рекомендації

У програмах, що зв'язані з обробкою даних чи обчисленнями, часто виконуються циклічно повторювані дії. Цикл - це послідовність операторів, що виконується неодноразово. У мові програмування Turbo Pascal є три різновиди циклу - *цикл з післяумовою* (інструкція `repeat`), *цикл з передумовою* (інструкція `while`) і *цикл із лічильником* (інструкція `for`).

Оператор REPEAT.

Оператор повтору `repeat` складається з *заголовка* (`repeat`), *тіла* і *умови закінчення* (`until`). Ключові слова `repeat`, `until` позначають "повторюй" і "поки" відповідно.

repeat

```
{ Інструкції }
```

```
until Умова виходу з циклу;
```

Спочатку виконується тіло циклу - інструкції, що знаходяться між repeat і until, потім перевіряється значення Умови виходу з циклу. У тому випадку, якщо воно дорівнює false (хибне), тобто не виконується - інструкції циклу повторюються ще раз. Так продовжується доти, поки умова не стане true (істина). На рис.23 приведена блок-схема оператора повтору repeat.

Варто знати::

- число повторень операторів (інструкцій) циклу repeat визначається в ході роботи програми і у багатьох випадках заздалегідь невідомо;
- інструкції циклу repeat будуть виконуватися, поки умова, що стоїть після until, буде залишатися хибною;
- після слова until записується умова завершення циклу;
- умова - це вираз логічного типу: простий вираз відношення чи складний логічний вираз;
- для успішного завершення циклу repeat у його тілі обов'язково повинні бути інструкції, виконання яких впливає на умову завершення циклу, інакше цикл буде виконуватися нескінченно - програма зациклиться. Іншими словами, змінна, котра бере участь в умові виходу з циклу, обов'язково повинна змінюватися в тілі циклу.
- цикл repeat - це цикл з післяумовою (умова перевіряється після виконання тіла циклу), тобто інструкції тіла циклу будуть виконані хоча б один раз;
- цикл repeat зручно використовувати в тих випадках, коли тіло циклу гарантовано повинно виконуватися хоча б один раз;
- нижня границя операторів тіла циклу чітко позначена словом until, тому немає необхідності вкладати ці оператори в операторні дужки begin і end. У той же час наявність операторних дужок не буде помилкою.

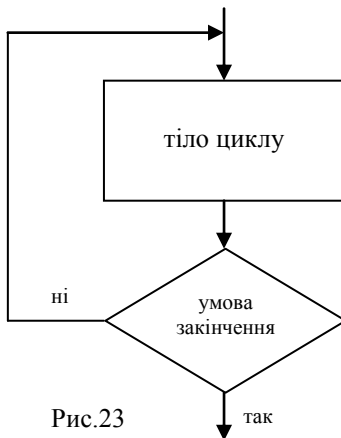


Рис.23

Приклад Гра "Вгадай число".

```
var
    comp: integer;           { число, яке "задумав" комп'ютер }
    igrok: integer;         { варіант гравця }
begin
```

```

    randomize;                                { ініціалізація датчика випадкових
чисел }
    comp:=random(1000);                        { комп'ютер задумав число }
    repeat
    write('Введіть число: '); readln(igrok);
    if igrok>comp then writeln('Занадто велике...')
    else if igrok<comp then writeln('Занадто мале...')
        else writeln ('Ви вгадали!');
    until igrok=comp;
end.

```

Оператор WHILE

Оператор повтору while складається з *заголовка* і *тіла циклу*. Ключові слова while і do позначають "доти, поки" і "виконуй" відповідно.

while Умова виконання циклу **do**
begin

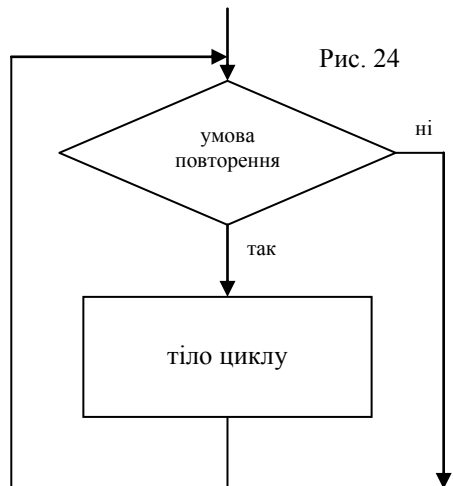
{ Інструкції }

end;

Оператор while аналогічний оператору repeat, але перевірка Умови виконання циклу відбувається на самому початку оператора - якщо значення умови дорівнює true (істина), то виконуються інструкції циклу, що знаходяться між begin та end і знову обчислюється вираз Умова виконання циклу. Так продовжується доти, поки значення Умова виконання циклу не стане рівною false (хибна). На рис.24 приведена блок-схема оператора повтору while.

Варто знати:

- *число повторень* операторів (інструкцій) циклу while визначається в ході роботи програми і, як правило, заздалегідь невідомо;
- після слова while записується умова продовження виконання інструкцій циклу, у цьому відмінність циклу while від циклу repeat;
- умова - це вираз логічного типу: простий вираз відношення чи складний вираз відношення, що може приймати одне з двох значень: true чи false;



- для успішного завершення циклу `while` у його тілі обов'язково повинні бути присутні інструкції, що *впливають* на умову виконання інструкцій циклу.
- цикл `while` - це цикл із *передумовою*, тобто інструкції тіла циклу взагалі можуть бути не виконані, якщо умова, що перевіряється, хибна з самого початку;
- цикл `while` зручно використовувати в тих випадках, коли можливі ситуації невиконання циклу;
- в операторі циклу `while` крапка з комою ніколи не ставиться після зарезервованого слова `do`.

Приклад Пошук суми чисел від 1 до 10.

```
var
    s, i: integer;
begin
    s:= 0; i:= 1;
    while i<=10 do      { знаходимо суму чисел від 1 до 10 }
        begin
            s:=s+i;
            i:=i+1;      { зміна змінної керування циклом }
        end;
    writeln ('Сума = ', s:4);
end.
```

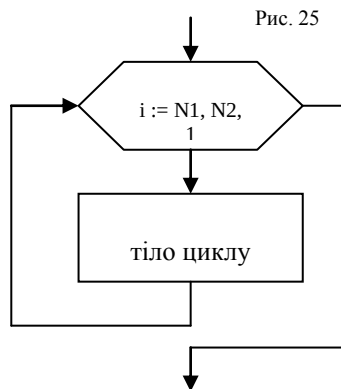
Оператор FOR

Цей вид оператора циклу називають *циклом з лічильником* чи *циклом з параметром*. У ньому важливу роль відіграє *змінна-параметр*, що на кожному кроці циклу автоматично змінює своє значення на одиницю - тому її і називають лічильником. Інструкцію `for` можна реалізувати двома способами.

Варіант 1 (зі збільшенням лічильника).

```
for Лічильник :=
ПочатковеЗначення to
КінцевеЗначення do
    begin
        { Інструкції }
    end;
```

Ключові слова `for`, `do` позначають "для", "виконуй" відповідно. Рядок, який містить `for...do`, називається *заголовком циклу*, оператор, що стоїть після `do` утворює його *тіло*. Дуже часто тіло циклу - складений оператор. Якщо тіло циклу представлене одним



оператором, то `begin` і `end` не пишуться. На блок-схемі алгоритму цикл із параметром позначають в спосіб зображений на рис. 25.

Інструкції між `begin` і `end` виконуються стільки разів, скільки визначає вираз $[(\text{КінцевеЗначення} - \text{ПочатковеЗначення}) + 1]$. Це відповідає всім значенням лічильника від початкового до кінцевого включно. Якщо `ПочатковеЗначення` більше, ніж `КінцевеЗначення`, то інструкції між `begin` і `end` не виконуються жодного разу.

Варіант 2 (із зменшенням лічильника).

for Лічильник := ПочатковеЗначення **downto**
КінцевеЗначення **do**

begin
 { Інструкції }
end;

Інструкції між `begin` і `end` виконуються стільки разів, скільки визначає вираз $[(\text{ПочатковеЗначення} - \text{КінцевеЗначення}) + 1]$. Якщо `ПочатковеЗначення` менше, ніж `КінцевеЗначення`, то інструкції між `begin` і `end` не виконуються жодного разу.

Приклад. Обчислення факторіалу числа n

```
uses crt;  
var k, n, f: integer;  
begin  
  clrscr; { очищення екрана }  
  writeln('Введіть число ?');  
  readln(n);  
  f:= 1;  
  for k:=1 to n do  
    begin  
      f:=f*k;  
    end;  
  writeln('факторіал числа ',n:4, ' рівний ', f);  
end.
```

Завдання 1.

1. 10 разів виконати таку дію: вивести на екран число 2
2. N разів виконати такі дії: значення змінної E додати до значення змінної S і результат присвоїти змінній E
3. Дванадцять разів виконати такі дії: вивести на екран число M.
4. S разів виконати такі дії: значення змінної E додати до значення змінної S і результат присвоїти змінній S.
5. Сім разів виконати таку дію: змінній S присвоїти результат додавання значення змінної S і числа 2.
6. D разів виконати таку дію: вивести на екран суму значень змінних A і B.

7. Двадцять разів виконати таку дію: змінній T присвоїти результат додавання змінної T і числа 2.
8. K разів виконати такі дії: присвоїти змінній результат віднімання числа 5 від змінної C, вивести на екран значення змінної C.
9. 15 разів виконати таку дію: вивести на екран число 5.
10. N разів виконати такі дії: значення змінної E помножити на значення змінної C і результат присвоїти змінній E
11. 10 разів виконати такі дії: вивести на екран число (- M).
12. N разів виконати такі дії: значення змінної E відняти від значення змінної C і результат присвоїти змінній C.

Завдання 2.

Дано натуральні числа a, b, c. Необхідно вивести таблицю відповідності:

1. маси в кілограмах і фунтах для значень фунтів від a до b з кроком c фунтів, знаючи що 1 фунт = 0,4 кг;
2. віддалі в дюймах і сантиметрах для значень дюймів від a до b з кроком c дюймів, знаючи що 1 дюйм = 2,54 см;
3. віддалі в футах і сантиметрах для значень фунтів від a до b з кроком c футів, знаючи що 1 фут = 30,48 см;
4. віддалі в милях і кілометрах для значень миль від a до b з кроком c миль, знаючи що 1 миля = 1,6 км;
5. вартості порції масла масою від a до b кілограм з кроком c кілограм, знаючи що 1 кг масла коштує 11,7 грн.
6. вартості молока об'ємом від a до b літрів з кроком c літрів, знаючи що 1 літр молока коштує 1,65 грн.
7. вартості бензину об'ємом від a до b літрів з кроком c літрів, знаючи що 1 літр бензину коштує 3,95 грн.
8. об'єму пам'яті в кБ та байтах від a до b кБ з кроком c кБ, знаючи, що 1 кБ = 1024 байт.
9. маси в грамах та каратах для значень грамів від a до b з кроком c грамів, знаючи що 1 карат = 0,2 г;
10. маси в кілограмах і пудах для значень пудів від a до b з кроком c пудів, знаючи що 1 пуд = 16,38 кг;
11. віддалі в кабельтов і метрах для значень кабельтов від a до b з кроком c кабельтов, знаючи що 1 кабельтов = 185,2 м;
12. віддалі в морських милях і кілометрах для значень миль від a до b з кроком c миль, знаючи що 1 морська миля = 1,852 км;

Завдання 3.

Скласти і вивести на екран таблицю значень функції для заданих значень аргументу від a до b з кроком c. Результат вивести у вигляді таблиці:

1. $y = \sqrt{2x^2 - 2}$
2. $y = \sqrt{x^2 + 4}$
3. $y = x^3 / 2 + 2;$

$$4. y = (x - 1) / x \quad 5. y = x^2 + 2x + 3 \quad 6. y = 3 - x^2$$

Дано натуральне число n .

7. Обчислити 2^n (без використання формули піднесення до степеня);
8. Обчислити $1/n!$ ($1 \leq n \leq 10$);
9. Обчислити суму $s = 1 + 1/2 + 1/3 + \dots + 1/n$;
10. Обчислити суму $s = 1/2 + 2/3 + 3/4 + \dots + (n - 1)/n$ ($2 \leq n \leq 30$);
11. Обчислити добуток $d = 2/1 \cdot 3/2 \cdot 4/3 \cdot \dots \cdot n/(n-1)$ ($2 \leq n \leq 15$);
12. Обчислити добуток $d = (1 + 1/1^2) \cdot (1 + 1/2^2) \cdot \dots \cdot (1 + 1/n^2)$

Завдання 4.

1. У магазин надійшло N видів сиру. Про кожен вид сиру відомо: кількість голівок сиру, вага однієї голівки сиру (y кг), ціна 1 кг сиру (y грн.). Складіть програму для визначення вартості всього сиру, що надійшов.
2. Відомі річні оцінки учня з усіх навчальних предметів. Скласти програму для визначення, чи є цей учень «хорошистом» («хорошист» - це той учень, у якого всі оцінки не нижче «7» і не вище «9»).
3. На фермі працюють T тракторів. Про кожен трактор відомо: скільки днів він працював на полі і скільки гектарів обробляв щодня. Складіть програму для визначення площі, обробленої всіма тракторами разом.
4. Відомо, скільки балів набрав учасник олімпіади на кожному з N турів, а також максимально можлива кількість балів за кожен тур. Складіть програму для визначення, чи може цей учасник стати призером олімпіади, якщо для цього необхідно набрати не менше, ніж 30 % максимально можливої кількості балів, а також ні за який тур не одержати 0 балів.
5. У басейн наливається вода з N труб. Про кожну трубу відомо: скільки літрів води в хвилину наливається з труби в басейн, скільки хвилин була відкрита труба. Складіть програму для визначення, скільки літрів води було наливо в басейн.
6. Відомо, скільки балів набрав учасник олімпіади на кожному з N турів, а також максимально можлива кількість балів за кожен тур. Скласти програму для визначення, чи може цей учасник стати призером олімпіади, якщо для цього необхідно за кожен тур набрати не менше, ніж 25 % максимально можливої кількості балів, або за усі тури разом не менше, ніж 40 % балів.
7. У міській естафеті «Забіг на 10 км» брали участь N учнів школи. Про кожного учня відомо: відстань, що він пробіг, а також його середня швидкість. Складіть програму для визначення командного часу забігу.
8. Відомі річні оцінки учня з усіх навчальних предметів. Складіть програму для визначення, чи є цей учень «трієчником». «Трієчник» - це той учень, у якого всі оцінки не нижче «4», однак він не є «хороши-стом» чи «відмінником» (всі оцінки не нижче «7»).
9. Дано числа від 1 до 100. Визначити скільки з цих чисел діляться на 3.
10. Ви поклали гроші в банк під 10% у місяць. Скласти програму, яка б визначала яка сума буде знаходитись на Вашому рахунку через N місяців.

11. Визначити яким повинно бути N , щоб значення суми $S = 1 + 2 + 3 + \dots + N$ було рівне або більше 1000.

12. У басейн наливається вода з N труб. Про кожну трубу відомо: скільки літрів води в хвилину наливається з труби в басейн, скільки хвилин була відкрита труба. Складіть програму для визначення, скільки часу потрібно для наповнення басейну ємністю L .

Лабораторна робота № 27-28. Масиви. Алгоритми обробки таблиць.

Методичні рекомендації

Якщо робота програм пов'язана із збереженням та обробкою великого об'єму однотипних змінних, для їх представлення в програмі можна використовувати масиви. *Масив* – це сукупність даних *одного типу* із загальним для всіх елементів *іменем*. Елементи масиву *пронумеровані*, і звернутися до кожного з них можна за номером (або декількома номерами – наприклад, для елемента таблиці задається номер рядка та стовпця). Номери елементів масиву інакше називаються *індексами*, а самі елементи масиву – *змінними з індексами* (*індексованими змінними*).

Характеристики масиву:

- Тип – *спільний* тип всіх елементів масиву;
- Розмірність (ранг) – *кількість індексів* масиву;
- Діапазон зміни індексу (індексів) – *визначає кількість елементів в масиві*.

Вектор (одномірний масив) – це приклад масиву, в якому елементи нумеруються одним індексом. Якщо в масиві зберігається таблиця значень (матриця), то такий масив називається *двомірним*, його елементи нумеруються *двома* індексами – номером *рядка* та *стовпця* відповідно. В якості номеру (індекса) елемента масиву, в загальному випадку, використовується вираз порядкового типу. Найчастіше індекс – це ціла константа або змінна типу `integer`, зрідка - типу `char` або `boolean`. При звертанні до елемента масиву індекс вказується в квадратних дужках після імені масиву. Наприклад, `a[3]`, `b[1,2]`.

Опис масиву в розділі VAR.

Для одномірного масиву:

```
var Ім'яМасиву: array [НижняМежа .. ВерхняМежа]  
      of ТипЕлементів;
```

Приклад.

```
var  
a: array [1..100] of integer;      { 100 елементів -  
цілі числа }  
b: array [0..50] of char;        { 51 елемент - символи }  
c: array [-3..4] of boolean;     { 8 елементів - логічні  
значення }
```

Для двовірного масиву:

```
var      Ім'яМасиву:      array      [НижняМежаІндекс1 ..  
ВерхняМежаІндекс1, НижняМежаІндекс2 .. ВерхняМежаІндекс2]  
of ТипЕлементів;
```

Приклад.

```
var  
  a: array [1..3, 1..4] of integer; { матриця з трьох  
рядків і 4-х стовпців, елементи - цілі числа}
```

Заповнення масиву даними

- Заповнення вектору з n елементів:

```
for i:=1 to n do readln(a[i]);
```

- Заповнення матриці розміром $m \times n$ елементів (всього потрібно ввести $m \times n$ чисел):

```
for i:=1 to m do  
  for j:=1 to n do  
    readln(a[i,j]);
```

- Заповнення вектора з n елементів випадковими числами в діапазоні від 0 до 99 включно:

```
randomize {ініціалізація датчика  
випадкових чисел}  
for i:=1 to n do a[i]:=random(100);
```

Вивід масиву

- В стовпець:

```
for i:=1 to n do writeln(a[i]);
```

- Матриці розміром $m \times n$ елементів в стандартній формі запису – по рядкам і стовпцям:

```
for i:=1 to m do  
  begin  
    for j:=1 to n do  
      write(a[i,j:4]);  
    writeln;  
  end;
```

Обробка масиву

Дії з одномірними масивами (вектор, що містить n елементів)

- Обчислення суми елементів:

```
S:=0;  
for i:=1 to n do s:=s+a[i]; {звичайне накопичення  
суми в s}
```

- Обчислення добутку елементів:

```
p:=1;  
for i:=1 to n do p:=p*a[i]; {звичайне накопичення  
добутку в p}
```

- Підрахунок кількості елементів, що задовольняють певній умові (наприклад, кількість парних чисел в цілочисельному масиві):

```
k:=0;
for i:=1 to n do
    if a[i] mod 2=0 then k:=k+1;      {збільшуємо на 1
лічильник парних чисел, якщо число ділиться на 2}
```

- Пошук елемента із заданим значенням та вивід його номера в масиві:

```
i:=0;      {номер елементів масиву}
repeat
    i:=i+1;
until (a[i]=0){знайшли} or (i=n){масив закінчився};
if a[i]=0 then writeln('Номер першого нульового
елемента = ', i)
    else writeln('Таких елементів немає');
```

- Пошук максимального елемента та його номера в масиві:

```
max:=a[1]; k:=1; {змінна max зберігає значення
максимуму, k- його номер в масиві}
for i:=1 to n do {перебираємо елементи, починаючи з
другого}
    if a[i]>max then
        begin
            max:=a[i]; k:=i; {запам'ятовуємо значення і номер
елемента, який більше всіх попередніх}
        end;
```

Дії з двомірними масивами (n рядків та m стовпців)

- Обчислення суми елементів кожного рядка:

```
for i:=1 to n do
    begin
        S:=0;
        for j:=1 to m do s:=s+a[i,j];
        d[i]:=s;      {результатом є вектор d, що
складається з n сум елементів рядків}
    end;
```

- Пошук мінімального елемента всієї матриці:

```
min:=a[1,1]; k:=1; l:=1; {змінна min зберігає значення мінімуму, k
і l- номер рядка та стовпця, де він знаходиться}
for i:=1 to n do
    for j:=1 to m do
        if a[i,j]<min then
            begin
                min:=a[i,j]; k:=i; l:=j;
            end;
```

- Множення матриці a на вектор x, в результаті отримуємо новий вектор y:

```

for i:=1 to n do
  begin
    s:=0;
    for j:=1 to m do
      s:=s+a[i,j]*x[j];
    y[i]:=s;
  end;

```

Завдання 1.

Ввести одномірний масив А з N елементів та виконати наступні дії:

1. 5-му елементу присвоїти значення останнього елемента;
2. 7-му елементу присвоїти значення суми попереднього та наступного елементів;
3. різницю значень останнього та першого елементів присвоїти другому елементу;
4. 5-му елементу присвоїти результат множення першого та останнього елементів;
5. суму значень першого та другого елементів присвоїти останньому елементу;
6. змінити знак першого елемента на протилежний;
7. змінити знак останнього елемента на протилежний;
8. першому елементу масиву присвоїти його подвоєне значення;
9. останньому елементу масиву присвоїти його подвоєне значення;
10. першому елементу масиву присвоїти значення добутку останнього та передостаннього елементів;
11. останньому елементу масиву присвоїти значення добутку першого та другого елементів;
12. перший та останній елементи поміняти місцями.

Результуючий масив вивести на екран.

Завдання 2.

Ввести одномірний масив А з N елементів та виконати наступні дії:

1. Знайти мінімальний елемент та присвоїти йому значення 0.
2. Знайти максимальний елемент та присвоїти йому значення 100.
3. Знайти максимальний за модулем елемент та поміняти його місцями з останнім елементом масиву.
4. Знайти мінімальний за модулем елемент та поміняти його місцями з першим елементом масиву.
5. Підрахувати кількість від'ємних елементів масиву та присвоїти їм значення 0.
6. Підрахувати кількість додатніх елементів масиву та присвоїти їм значення 100.
7. Підрахувати кількість елементів кратних 3-м та присвоїти їм значення 33.
8. Підрахувати кількість елементів кратних 5-ти та присвоїти їм значення 55.

9. Знайти максимальний за модулем елемент, його індекс та присвоїти йому частку від ділення цих даних.
10. Знайти мінімальний за модулем елемент, його індекс та присвоїти йому значення добутку цих даних.
11. Підрахувати різницю між від'ємними та додатніми елементами масиву.
12. Всім нульовим елементам масиву присвоїти значення 100.

Результат вивести на екран.

Завдання 3.

Ввести двовірний масив А з $M \times N$ елементів та виконати наступні дії:

1. Знайти суму модулів всіх від'ємних елементів матриці.
2. Знайти суму всіх додатніх елементів матриці.
3. Знайти мінімальний елемент і присвоїти йому значення 0.
4. Знайти максимальний елемент присвоїти йому значення 1000.
5. Всім парним елементам присвоїти значення 2;
6. Всім непарним елементам присвоїти значення 1;
7. Порахувати кількість від'ємних елементів;
8. Порахувати кількість додатніх елементів;
9. Обчислити різницю між макисмальним та мінімальним елементами матриці;
10. Обчислити кількість парних елементів матриці.
11. Обчислити кількість непарних елементів матриці.
12. Знайти кількість елементів матриці, значення яких більше значення першого елемента.

Результат вивести на екран.

Завдання 4.

Ввести двовірний масив А з $M \times N$ елементів та виконати наступні дії:

1. Знайти суму всіх елементів, що лежать в діапазоні $(-\infty; 1)$ і її значення присвоїти елементам, що знаходяться в першому рядку матриці.
2. Знайти суму всіх елементів, що лежать в діапазоні $(4; +\infty)$ і її значення присвоїти елементам, що знаходяться в останньому рядку матриці.
3. Знайти мінімальний елемент і його значення присвоїти елементам, що знаходяться в першому рядку матриці.
4. Знайти максимальний елемент і його значення присвоїти елементам, що знаходяться в останньому рядку матриці.
5. Матриця $M=N$. Елементам головної діагоналі присвоїти значення 0;
6. Матриця $M=N$. Елементам, що знаходяться нижче головної діагоналі присвоїти значення 1;
7. Матриця $M=N$. Елементам, що знаходяться вище головної діагоналі присвоїти значення 100;
8. Матриця $M=N$. Знайти мінімальний елемент, та його значення присвоїти елементам головної діагоналі;
9. Матриця $M=N$. Знайти максимальний елемент, та його значення присвоїти елементам головної діагоналі;

10. Знайти мінімальний елемент і рядок, де він знаходиться поміняти з першим рядком матриці.
 11. Знайти максимальний елемент і рядок, де він знаходиться поміняти з останнім рядком матриці..
 12. Знайти мінімальний та максимальний елемент та поміняти їх місцями.
- Результат вивести на екран.

Лабораторна робота № 29-30. Масиви. Методи сортування.

Методичні рекомендації

Сортування - це процес упорядкування набору даних одного типу по зростанню чи спаданню значення якої-небудь ознаки. При сортуванні елементи масиву міняються місцями таким чином, що їхні значення виявляються упорядкованими чи по зростанню, чи по спаданню. Якщо в масиві є однакові елементи, то говорять про сортування чи по неспаданню чи по незростанню. У більшості випадків мова йде про сортування одномірного масиву.

Звичайна постановка задачі по сортуванню масиву виглядає в такий спосіб: програма повинна вивести несортований масив цілих чисел на екран, виконати його сортування, а потім вивести відсортований масив.

Розглянемо два методи сортування по незростанню для того самого масиву *m* з 20 цілих чисел. Для спостереження поточного стану масиву після кожної перестановки елементів будемо виводити його на екран. Для зручності налагодження і тестування масив задається у вигляді типізованої константи.

Лінійне сортування (сортування добором)

Ідея лінійного сортування по незростанню полягає в тім, щоб, послідовно переглядаючи весь масив, відшукати найбільше число і поміняти його місцями з першим елементом. Потім проглядаються елементи масиву, починаючи з другого, знову знаходиться найбільший, котрий міняється місцями з другим і т.д.

```
const count=20;
```

```
m: array [1..count] of byte =  
(9,11,12,3,19,1,5,17,10,18,3,19,17,9,12,20,20,19,2,5);  
var i,j,buf,l: byte; a: integer;  
begin  
  writeln('Вихідний масив:');  
  for i:=1 to count do write(' ', m[i]);  
  writeln; readln;  
  {змінa розміру невідсортованої частини масиву}  
  for i:=1 to count-1 do  
    {порівнюємо по чергово i-й елемент невідсортованої частини  
    масиву з усіма від i+1-го до кінця}
```

```

        for j:=i+1 to count do
        begin
            if m[i]<m[j] then
                { якщо в невідсортованій частині масиву знайшли елемент,
                більший, ніж i-й, то міняємо їх місцями}
                begin
                    buf:=m[i];    { buf – буфер обміну }
                    m[i]:=m[j] ;
                    m[j]:=buf;
                end;
            for l:=1 to count do write(' ', m[l]);
        {після сортування}
        end;
    end.

```

Сортування методом "бульбашки"

“Бульбашковий” метод сортування – заснований на тому, що в процесі виконання алгоритму більш “легкі” елементи масиву поступово “спливають”. Особливістю даного методу є порівняння, а потім, якщо потрібно, і перестановка сусідніх елементів

```

const count=20;
    m: array [1..count] of byte =
        (9,11,12,3,19,1,5,17,10,18,3,19,17,9,12,20,20,19,2,5);
var i,j,buf,l: byte;
begin
    writeln('Вихідний
    масив:');
    for i:=1 to count do
        write(' ',m[i]);
    writeln;
    readln;
    for i:=2 to
        count do
        begin
            for j:=count
                downto i do
                begin
                    if m[j-1]<m[j] then { якщо елемент праворуч
                    більше елемента}
                    { ліворуч, то "витісняємо" його вліво -
                    бульбашка "спливає"}
                    begin
                        buf:=m[j-1];    {
                        перестановка елементів }
                        m[j-1]:=m[j];
                        m[j]:=buf;
                    end;
                for l:=1 to count do write(' ',m[l]);

```

```
        end;  
    end;  
end.
```

Завдання 1.

Заповнити матрицю розмірності 20х20 елементів випадковими числами. Рядок, номер якого відповідає Вашому варіанту відсортувати одним з методів за неспаданням. Результат вивести на екран.

Завдання 2.

Заповнити матрицю розмірності 20х20 елементів випадковими числами. Елементи головної діагоналі відсортувати за незростанням.

Лабораторна робота № 31-33. Процедури та функції.

Методичні рекомендації

Якщо в програмі виникає необхідність частого звернення до деякої групи операторів, які виконують дії або обчислюють вираз, то раціонально згрупувати ці оператори в блок, до якого можна звертатися по імені. Такі розроблені програмістом самостійні програмні блоки називаються *підпрограмами користувача*. Використання підпрограм дозволяє, зосереджуючи в них детальний опис деяких операцій, в основній програмі вказувати тільки *імена підпрограм*, щоб виконати ці операції. Такі звернення до підпрограм можливі багаторазові та з різних ділянок основної програми, причому при виклику підпрограми можна *передати* інформацію (різну в різних зверненнях), щоб одна і та ж підпрограма виконувала розв'язання підзадачі для різних випадків.

Процедура – це незалежна іменована частина програми, яку після *одноразового* опису можна *багаторазово* викликати по *імені* з наступних частин програми для виконання певних дій. Процедура не може виступати як операнд у виразі.

Структура процедури повторює структуру програми, вона також представлена заголовком та тілом. На відміну від програми, для процедур та функцій заголовок є обов'язковим. *Заголовок* складається із зарезервованого слова `procedure`, ідентифікатора (*імені*) процедури та необов'язкового, заключеного в круглі дужки, списку *формальних параметрів* із вказанням *типу* кожного параметра:

Procedure Ім'яПроцедури (ФормальніПараметри);

Ім'я процедури повинно бути унікальним, тобто його не можна використовувати повторно в програмі для найменування інших процедур. *Тіло процедури* за своєю структурою аналогічне звичайній програмі:

Procedure Ім'яПроцедури (ФормальніПараметри);

```

    {Описова частина процедури}
begin
    {Інструкції виконуваної частини процедури}
end;

```

Для звернення до процедури використовується *оператор виклику процедури*. Він складається з *імені* процедури та списку *фактичних параметрів*, що відділені один від одного комами та заключені в круглі дужки. Список параметрів *відсутній*, якщо процедурі *не передаються* жодні значення.

Ім'яПроцедури (ФактичніПараметри);

Параметри забезпечують механізм *заміни*, що дозволяє виконувати процедуру з *різними початковими даними*. Між фактичними параметрами в операторі виклику процедури і формальними параметрів у заголовку опису процедури встановлюється *взаємно однозначна відповідність* в результаті їх перебору зліва праворуч. Кількість і тип формальних параметрів рівні кількості і типу фактичних параметрів. Відповідні один одному параметри не обов'язково повинні однаково позначатися.

Якщо процедура повертає в програму якісь значення, відповідні змінні повинні бути описані як *параметри* - змінні з використанням слова *var*.

При виклику процедури робота головної програми *припиняється* і починає виконуватися *викликана процедура*. Коли процедура виконає свою задачу, програма продовжиться з оператора, що слідує за оператором виклику процедури.

Опис міток, констант, типів і т.п. *дійсний тільки в межах даної користувальницької процедури*. Для примусового виходу з процедури використовується оператор завершення *exit*, що забезпечує вихід у зовнішній блок (зазвичай - основну програму).

Приклад . Запишемо процедуру, що виконує зведення до цілого ненегативного степеня будь-яке числа. За допомогою даної процедури обчислимо, скільки байтів міститься в кілобайті, мегабайті і гігабайті, використовуючи відомі співвідношення: 1Кбайт = 210 байт, 1 Мбайт =220 байт, 1 Гбайт =230 байт

```

procedure degree(x: real; n: byte; var res: real);
{ процедура приведення числа x до цілого невід'ємного степеня n,
}
{ результатом є параметр-змінна res }
var i: integer;
begin
    res:=1;
    for i:=1 to n do res:=res*x;
end;
{ основна програма }
var kb,mb,gb: real;
begin
    degree (2,10,kb);

```

```

degree (2,20,mb);
degree (2,30,gb);
writeln('1 Kb = ', kb:4:0, ' byte');
writeln('1 Mb = ', mb:7:0, ' byte');
writeln('1 Gb = ', gb:10:0, ' byte');
end.

```

Функції користувача

Якщо результатом підпрограми є тільки одне значення, то має сенс оформити таку підпрограму не у вигляді процедури, а у вигляді функції. *Функція користувача* аналогічна процедурі, але є дві відмінності.

- Функція передає в програму результат своєї роботи - єдине значення, носієм якого є ім'я самої функції.
- Ім'я функції може входити у вираз як операнд. Функція повертає результат у точку свого виклику.

Функція, визначена користувачем, складається з заголовка і тіла функції. *Заголовок* містить зарезервоване слово *function*, *ім'я* функції, обмежений круглими дужками *необов'язковий* список *формальних* параметрів і, на відміну від процедури, *тип* значення, що повертається функцією:

function Ім'яФункції (ФормальніПараметри) :

ТипРезультату;

Наприклад:

```

function fibo(n: integer): integer;
function instep(a,b: real): real;
function normrandom : double;

```

Ім'я функції унікальне в межах програми. *Тіло* функції за своєю структурою аналогічне звичайній програмі:

```

function Ім'яФункції (ФормальніПараметри) : ТипРезультату;
{ Описова частина функції }
begin
{ Інструкції виконуваної частини функції }
Ім'яФункції := Результат;
end;

```

У розділі операторів функції повинен знаходитися принаймні один оператор, що присвоює її імені значення, яке повертається як *результат роботи функції*. Якщо таких присвоєвань *декілька*, то результатом виконання функції буде значення *останнього* оператора присвоєння. Якщо ж такий оператор відсутній чи не був виконаний, то значення, яке повертається функцією, *не визначено*.

На відміну від процедури, виклик функції не оформляється у вигляді окремого оператора. Звертання до функції здійснюється шляхом використання вказівника функції в якості операнда в деякому виразі. *Вказівник функції* являє собою ім'я функції з *необов'язковим* списком

аргументів - фактичних параметрів. Вимоги до них такі ж, як і у випадку процедури.

Для допитливих !!!

Приклад.

У 1202 р. італійський математик Леонард Пізанський, відомий під ім'ям Фібоначчі, запропонував наступну задачу. Пара кроликів кожен місяць дає приплід - двох кроликів (самця і самку), від яких через два місяці уже виходить новий приплід. Скільки пар кроликів буде через рік, якщо на початку року ми мали одну пару тільки що народжених кроликів? Звертається увага на те, що числа, які відповідають кількості пар кроликів по місяцях, складають послідовність

1, 1, 2, 3, 5, 8, 13, 21, 34, ...

Кожний із членів цієї послідовності, починаючи з третього, дорівнює сумі двох попередніх членів. Ця послідовність називається рядом Фібоначчі, а її члени - числами Фібоначчі. Числа Фібоначчі мають багато цікавих властивостей. Ряд Фібоначчі визначають так:

$$F_0 = F_1 = 1; \dots F_n = F_{n-1} + F_{n-2}.$$

Програма обчислення членів ряду Фібоначчі

```
function fibo(n: integer): integer;
{ функція обчислення чисел Фібоначчі }
var fn, fn1, fn2, k: integer;
begin
    fn1:=1;
    fn:=0;
    for k:=1 to n do
        begin
            fn2:=fn1;
            fn1:=fn;
            fn:= fn1+fn2;
        end;
    fibo:= fn; { імені функції присвоюється значення, яке
повертається }
end;
{ основна програма }
var i, n: integer;
begin
    write('Введіть число членів ряду Фібоначчі: '); readln(n);
    for i:= 1 to n do writeln(fibo(i));    { викликв функції }
end.
```

Завдання 1.

Виконати завдання з використанням умови завдання 2 лабораторної роботи 28 із використанням процедури для оформлення результатів у вигляді таблиці. Горизонтальні лінії якої виділити різним кольором (за допомогою звернення до процедури).

Завдання 2.

Скласти програму з використанням процедури для обчислення виразу:

1. $S = 1 + 3 + 5 + 7 + \dots + (2n - 1);$

2. $S = 1 + 9 + 25 + \dots + (2n - 1)^2;$

3. $S = 1 + 27 + 125 + \dots + (2n - 1)^3;$

4. $S = \sum_{i=1}^n \frac{i(i-1)}{2};$

5. $P = \prod_{i=1}^n \frac{i(i-1)}{2}$

6. $S = \sum_{i=1}^n \frac{i^2(i-1)^2}{2}$

7. $S = \sum_{i=1}^n \frac{i^2}{2^i}$

8. $S = 2 + 4 + 6 + 8 + \dots + 2n ;$

9. $S = 2 + 4 + 16 + \dots + (2n)^2;$

10. $S = 8 + 64 + 216 + \dots + (2n)^3$

11. $S = \sum_{i=1}^n (1 + \frac{1}{i^2});$

12. $S = \sum_{i=1}^n \frac{i^3}{3^i}$

Лабораторна робота № 34-35. Обробка рядків.

Методичні рекомендації

Тип STRING (рядок) у Паскалі широко використовується для обробки текстів. Він багато в чому схожий на одномірний масив символів ARRAY[0..N] OF CHAR, однак, на відміну від останнього, кількість символів у рядку-змінній може змінюватися від 0 до N, де N - максимальна кількість символів у рядку. Значення N визначається оголошенням типу STRING [N] і може бути будь-як константою порядкового типу, але не більше 255. Рядок трактується як ланцюжок символів. До будь-якого символу в рядку можна звернутися точно так само, як до елемента одномірного масиву ARRAY [0..N] OF CHAR, наприклад:

```
var
st : String;
begin

.....

if st[5] = 'A' then...
```

end.

Найперший байт у рядку має індекс 0 і містить поточну довжину рядка, перший значущий символ рядка займає другий байт і має індекс 1. Над довжиною рядка здійснювати необхідні дії і такий спосіб змінювати довжину. Наприклад, видалити з рядка усі відомі пробіли можна в такий спосіб:

```
var
st : String;
i : Byte;
begin
i := ord(st [0] ) ; {i - поточна довжина рядка}
while (i <> 0) and (st[i] = ' ') do
begin
dec(i);
st[0] := chr(i)
end;
.....
end.
```

Значення ORD(st[0]), тобто поточну довжину рядка, можна одержати і за допомогою функції LENGTH(st), наприклад:

```
while (Length(st)<>0) and (st[Length(st)]= ' ') do
st[0] := chr(Length(st)-1)
```

До рядків можна застосовувати операцію "+" - зчеплення, наприклад:

```
st := 'a1' + 'b';
st := st + 'c'; {st містить "abc"}
```

Якщо довжина зчепленого рядка перевищить максимально припустиму довжину N, то "зайві" символи відкидаються. Наступна програма, наприклад, надрукує символ 1:

```
var
st: String [1] ;
begin
st:='123';
WriteLn(st)
end.
```

Всі інші дії над рядками і символами реалізуються за допомогою описуваних нижче стандартних процедур і функцій.

CONCAT(S1 [,S2, ... , SN]) - функція типу STRING; повертає рядок, що є результатом зчеплення рядків-параметрів S1, S2, ..., SN.

COPY(ST, INDEX, COUNT) - функція типу STRING; копіює з рядка ST COUNT символів, починаючи із символу з номером INDEX.

DELETE (ST, INDEX, COUNT) - процедура; видаляє COUNT символів з рядка ST, починаючи із символу з номером INDEX.

INSERT (SUBST, ST, INDEX) - процедура; вставляє підрядок SUBST у рядок ST, починаючи із символу з номером INDEX.

LENGTH (ST) - функція типу INTEGER; повертає довжину рядка ST.

POS (SUBST, ST) - функція типу INTEGER; відшукує в рядку ST перше входження підрядка SUBST і повертає номер позиції, з якої він починається; якщо підрядок не знайдено, повертається нуль.

STR(X [, WIDTH [: DECIMALS]], ST) - процедура; перетворить число X будь-якого дійсного чи цілого типів у рядок символів ST; параметри WIDTH і DECIMALS, якщо вони присутні, задають формат перетворення: WIDTH визначає загальну ширину поля, виділеного під відповідне символічне представлення дійсного чи цілого числа X, а DECIMALS - кількість символів у дробовій частині (цей параметр має сенс тільки в тому випадку, коли X- дійсне число).

VAL(ST, X, CODE) - процедура; перетворить рядок символів ST у внутрішнє представлення цілої чи дійсної змінної X, що визначається типом цієї змінної; параметр CODE містить нуль, якщо перетворення пройшло успішно, і тоді в X міститься результат перетворення, у іншому випадку він містить номер позиції в рядку ST, де виявлено помилковий символ, і в цьому випадку вміст X не змінюється.

Наприклад:

```
var
x : Real;
y : Integer;
st,st1: String;
begin
    st := concat('12','345'); {рядок st містить
12345}
    st1 := copy(st,3,Length(st)-2); {st1 містить
345}
    insert('-',st1,2); {рядок st1 містить 3-45}
    delete(st,pos('2',at),3); {рядок st містить 15}
    str(pi:6:2,st); {рядок st містить 3.14}
    val('3,1415' ,x,y) ; {y містить 2, x залишився
без зміни}
end.
```

Завдання 1.

Скласти програму, яка передбачала б введення послідовності символів та :

1. Порахувати скільки разів у цій послідовності зустрічається буква «х».
2. Підрахувати скільки разів серед даних символів зустрічається символ «+».
3. Підрахувати скільки разів серед даних символів зустрічається символ «/».
4. Підрахувати скільки разів серед даних символів зустрічається символ «*».

5. Підрахувати загальне число символів «+», «-», «у» у даній послідовності.
6. Визначити порядковий номер останньої коми у цій послідовності.
7. Замінити у даній послідовності всі знаки оклику крапками.
8. Визначити чи є в даній послідовності комбінація символів ", - ".
9. Підрахувати найбільшу кількість пробілів, які йдуть підряд.
10. Визначити кількість цифр у даній послідовності.
11. Визначити чи є у даній послідовності символи відмінні від цифр.
12. Знайти довжину самої довгої послідовності цифр, які йдуть підряд.
13. Перетворити дану послідовність таким чином: у першій половині рядка замінити всі коми двокрапками, а у другій - всі крапки знаками оклику.
14. Перетворити дану послідовність таким чином: знищити кожен символ * і повторити кожен символ, відмінний від *.
15. Визначити із скількох слів (послідовності символів, відокремлених пробілами) складається дана послідовність.

Завдання 2.

Дано послідовність символів:

1. Визначити слово, яке містить найменшу кількість символів.
2. Визначити кількість букв в останньому слові.
3. Вивести на екран кожне слово у окремому рядку.
4. Замінити у даній послідовності всі символи "." словом "крапка".
5. Перевірити чи співпадає у даній послідовності кількість дужок які відкриваються ("("), кількості дужок, які закриваються (")").
6. Знищити з даної послідовності всі символи, які розміщені між дужками ("(...)"). Самі дужки також повинні бути знищені. Передбачається, що всередині кожної пари дужок інших дужок немає.
7. Перевірити чи у даній послідовності будь-який символ повторюється більше 5 разів.
8. Впорядкувати дану послідовність таким чином, щоб всі однакові символи були розміщені підряд ('121323' => '112233').
9. Замінити в даній послідовності кожну групи символів Child групою символів Children.
10. Замінити кожну крапку послідовності трьома крапками.
11. Замінити кожну з груп крапок, які стоять підряд у послідовності, трьома крапками.
12. Послідовність містить Ваше прізвище, ім'я, по батькові. Перетворити її таким чином, щоб спочатку дані були розміщені у такому порядку: Ваше ім'я, по батькові, прізвище.
13. Послідовність містить Ваше прізвище, ім'я, по батькові. Визначити Ваші ініціали (перші букви прізвища, імені, по батькові).
14. Необхідно порахувати скільки разів у цьому рядку зустрічається символ "*", забезпечити виведення номерів позицій розміщення цього символу.
15. Замінити у рядку всі знаки питання знаками оклику.