

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
МІСЬКОГО ГОСПОДАРСТВА імені О. М. БЕКЕТОВА**

**МЕТОДИЧНІ ВКАЗІВКИ
ДО ЛАБОРАТОРНИХ, ПРАКТИЧНИХ, САМОСТІЙНИХ
ТА КОНТРОЛЬНИХ РОБІТ
З ДИСЦИПЛІНИ**

ІНФОРМАТИКА ТА КОМП'ЮТЕРНА ТЕХНІКА

*(для студентів 1 курсу денної та заочної форм навчання за напрямом
підготовки 6.170202 «Охорона праці»)*



Харків – ХНУМГ – 2013

Методичні вказівки до лабораторних, практичних, самостійних та контрольних робіт з дисципліни «Інформатика та комп'ютерна техніка» (для студентів 1 курсу денної та заочної форм навчання за напрямом підготовки 6.170202 «Охорона праці») / Харк. нац. ун-т міськ. госп-ва ім. О. М. Бекетова; уклад.: С. В. Дядюн, М. П. Пан, Г. В. Білогурова. – Х.: ХНУМГ, 2013. – 113 с.

Укладачі: С. В. Дядюн
М. П. Пан
Г. В. Білогурова

Лабораторні, практичні, самостійні та контрольні роботи для студентів 1 курсу денної та заочної форм навчання за напрямом підготовки 6.170202 «Охорона праці» розроблені за вимогами кредитно-модульної системи організації навчального процесу, на підставі робочих програм з дисципліни, і містять такі розділи: лабораторні роботи та практичні заняття, завдання для самостійної роботи, контрольні роботи, а також рекомендації з виконання робіт.

Затверджено на засіданні кафедри прикладної математики і інформаційних технологій, протокол №6 від 15 грудня 2010 р.

ЗМІСТ

Вступ.....	4
1. ЛАБОРАТОРНІ РОБОТИ ТА ПРАКТИЧНІ ЗАНЯТТЯ	4
1.1. ОПЕРАЦІЙНА СИСТЕМА MICROSOFT WINDOWS	4
1.1.1. Лабораторна робота та практичне заняття №1. Знайомство з WindowsXP. Елементи робочого стола, робота з вікнами.....	4
1.1.2. Лабораторна робота та практичне заняття №2. Програма «Провідник». Робота з файловою системою.....	5
1.1.3. Лабораторна робота та практичне заняття №3. Стандартні програми WINDOWS.....	7
1.2. ТЕКСТОВИЙ ПРОЦЕСОР MICROSOFT WORD.....	8
1.2.1. Лабораторна робота та практичне заняття № 1. Створення та збереження документу у текстовому процесорі Word. Введення, редагування і форматування тексту.....	8
1.2.2. Лабораторна робота та практичне заняття № 2. Пошук і заміна тексту. Автокорекція та автотекст.....	9
1.2.3. Лабораторна робота та практичне заняття №3. Підготовка та виведення документа на друк.....	10
1.2.4. Лабораторна робота та практичне заняття №4. Текстові колонки.....	10
1.2.5. Лабораторна робота та практичне заняття №5. Робота з таблицями	12
1.2.6. Лабораторна робота та практичне заняття №6. Робота з графічними об'єктами.....	13
1.3. СТВОРЕННЯ ПРЕЗЕНТАЦІЙ ЗА ДОПОМОГОЮ POWER POINT	14
1.3.1. Лабораторна робота та практичне заняття №1. Знайомство з презентаціями.	14
1.3.2. Лабораторна робота та практичне заняття №2. Керування об'єктами на слайді.	14
1.4. ОСНОВИ ЕЛЕКТРОННИХ ТАБЛИЦЬ EXCEL.	15
1.4.1. Лабораторна робота та практичне заняття №1. Введення, редагування і форматування даних і формул.....	15
1.4.2. Лабораторна робота та практичне заняття №2. Використання функцій.Робота з масивами даних.....	17
1.4.3. Лабораторна робота та практичне заняття №3. Побудова і редагування діаграм.....	19
1.4.4. Лабораторна робота та практичне заняття №4. Організація та документування робочих книг Виведення на друк робочих аркушів і діаграм.....	20
1.5. МОВА ПРОГРАМУВАННЯ VBA – Visual Basic for Applications.....	22
1.5.1. Лабораторна робота та практичне заняття №1 Створення і виконання макросів.....	22
1.5.2. Лабораторна робота та практичне заняття №2 Лінійний обчислювальний процес.....	30
1.5.3. Лабораторна робота та практичне заняття №3 Обчислювальний процес, що розгалужується.....	41
1.5.4. Лабораторна робота та практичне заняття №4 Циклічний обчислювальний процес.....	47
1.5.5. Лабораторна робота та практичне заняття №5 Масиви.....	56
1.5.6. Лабораторна робота та практичне заняття №6 Двовимірні числові масиви.....	65
1.5.7. Лабораторна робота та практичне заняття №7 Процедури і функції.....	72
2. САМОСТІЙНІ РОБОТИ	75
2.1. ОПЕРАЦІЙНА СИСТЕМА MICROSOFT WINDOWS	75
2.1.1. Самостійна робота №1.....	75
2.1.2. Самостійна робота №2.....	75
2.2. ТЕКСТОВИЙ ПРОЦЕСОР MICROSOFT WORD.....	75
2.2.1. Самостійна робота №1. Знайомство з текстовим редактором Microsoft Word.....	75
2.2.2. Самостійна робота №2. Форматування документів Microsoft Word.....	75
2.2.3. Самостійна робота №3. Створення маркованих і нумерованих списків.....	76
2.2.4. Самостійна робота №4. Прийоми роботи з таблицями. Вставка символів та буквиці.....	77
2.2.5. Самостійна робота №5. Прийоми роботи з об'єктами MS Word.....	77
2.2.6. Самостійна робота №6. Засоби підвищення швидкості роботи з документами MS Word.....	78
2.2.7. Самостійна робота №7. Колонтитули, стилі, шаблони в MS Word.....	78
2.3. POWER POINT.....	79
2.3.1. Самостійна робота. Створення презентації.....	79
2.4. ОСНОВИ ЕЛЕКТРОННИХ ТАБЛИЦЬ EXCEL.....	79
2.4.1. Самостійна робота № 1. Знайомство з Excel	79
2.4.2. Самостійна робота № 2. Створення електронних таблиць. Уведення й редагування даних.....	82
2.4.3. Самостійна робота № 3. Редагування таблиць. Формування робочих аркушів.....	84
2.4.4. Самостійна робота № 4. Створення й налаштування діаграм в Excel.....	87
2.5. МОВА ПРОГРАМУВАННЯ VBA – Visual Basic for Applications.....	89
2.5.1. Самостійна робота №1. Розгалужені обчислювальні процеси.....	89
2.5.2. Самостійна робота №2. Циклічні обчислювальні процеси.....	90
3. КОНТРОЛЬНІ РОБОТИ	96
3.1. ОС WINDOWS.....	96
3.2. ТЕКСТОВИЙ ПРОЦЕСОР MICROSOFT WORD.....	96
3.3. POWER POINT.....	97
3.4. МОВА ПРОГРАМУВАННЯ VBA – Visual Basic for Applications.....	97
3.4.1. Лінійні обчислювальні процеси.....	97
3.4.2. Розгалужені обчислювальні процеси.....	100
3.4.3. Циклічні обчислювальні процеси.....	105
3.4.4. Обробка списків та масивів.....	106
Літературні джерела.....	112

ВСТУП

На сучасному етапі розвитку суспільства ніщо не розвивається так інтенсивно, як інформатика та комп'ютерна техніка. Важко знайти сферу діяльності, де б не застосовувалися комп'ютери, при цьому область їхнього використання стає все ширшою. Розробка програмних комплексів, інтернет, електронна пошта, спеціалізовані програмні засоби в різноманітних галузях науки, техніки, економіки, енергетики, промисловості та сільського господарства, мультимедійні засоби, нарешті, вони просто витіснили друкарські машинки – ось далеко неповний перелік використання сучасних комп'ютерів.

Однак спочатку вони призначалися для виконання об'ємних і громіздких обчислювальних робіт, які були не під силу людині, рахівниці та калькулятору. Тому й зараз опанування комп'ютерної техніки у вузах неможливе не тільки без навичок оволодіння комп'ютером на рівні користувача, але й без вивчення основ алгоритмізації та програмування.

Найперший варіант мови програмування BASIC (Бейсик) був створений в 1964 році співробітниками Дартмутського коледжу Дж. Кемені і Т. Курцем. Історія походження назви мови теж вельми цікава. Аббревіатура BASIC так і розшифровується – «Beginner's All purpose Symbolic Instruction Code», що в перекладі означає «багатоцільову мову символічних команд для початківців».

Мова програмування VBA – Visual Basic for Applications, є результатом подальшого розвитку мови BASIC. Не дивлячись на те, що VBA для кожного з додатків, яке його підтримує (Microsoft Excel, Microsoft Word і так далі), має свій, особливий, набір можливостей, загальна основа у нього залишається незмінною. Тобто, освоївши одного разу VBA, наприклад, для Microsoft Excel, використання його, потім, наприклад, для автоматизації Microsoft Word не складе особливих труднощів.

1. ЛАБОРАТОРНІ РОБОТИ ТА ПРАКТИЧНІ ЗАНЯТТЯ

1.1 Операційна система Microsoft Windows

1.1.1 Лабораторна робота та практичне заняття № 1.

Знайомство з Windows. Елементи робочого стола, робота з вікнами

Порядок виконання роботи

1. За розділом «Навчальний матеріал» вивчіть поняття: графічний інтерфейс, робочий стіл, вікно, значок, ярлик, лінійка прокручування.
2. Увімкніть комп'ютер і проведіть аналіз роботи користувача з робочим столом.
3. Розгорніть меню кнопки «Пуск» і відобразіть у зошиті його основні елементи.
4. Отримайте доступ до вікна «Властивості: Миша», виконуючи послідовно переходи: кнопка «Пуск», «Панель керування», «Миша».
5. Клацніть закладкою «Показчики», вивчить колекцію покажчиків миші (ПМ), які використовуються у WINDOWS.
6. Змініть вигляд ПМ.
7. Не закриваючи панель "Миша", активізуйте вікно "Панель керування".
8. Переміщенням і зміною розмірів виберіть зручне розміщення вікон у полі екрана. При роботі з вікнами використовуйте лінійки горизонтального і вертикального прокручування.
9. Послідовно розташуйте вікна каскадом, зверху вниз, ліворуч, праворуч.
10. Звертаючи послідовно вікна, зверніть увагу на те, як змінюється вигляд панелі завдань.
11. Закрийте вікна.
12. Відкрийте вікно додатка «Калькулятор», виконуючи переходи: «Пуск», «Програми», «Стандартні», «Калькулятор».
13. Відкрийте вікно додатку «Блокнот», виконуючи переходи: «Пуск», «Програми», «Стандартні», «Блокнот».
14. Відобразіть у зошиті вигляд панелі інструментів.
15. Закрийте всі вікна.
16. За допомогою команди «Довідка й підтримка» отримайте довідку про новини у Windows.
17. Відкрийте вікно додатка WORD.

18. Виконайте операції переміщення і зміни розміру вікна додатка.
19. Виконайте команду: Вигляд, Панелі інструментів і видаліть всі інструментальні лінійки.
20. Створіть вікно додатка з панелями «Стандартна», «Форматування» з розмічальною лінійкою.
21. Закрийте вікно додатка WORD.
22. Закріпіть навички роботи з операційною системою WINDOWS, повторюючи завдання 11–19 необхідну кількість разів.
23. Кнопкою Пуск завершіть роботу з WINDOWS.

Контрольні запитання

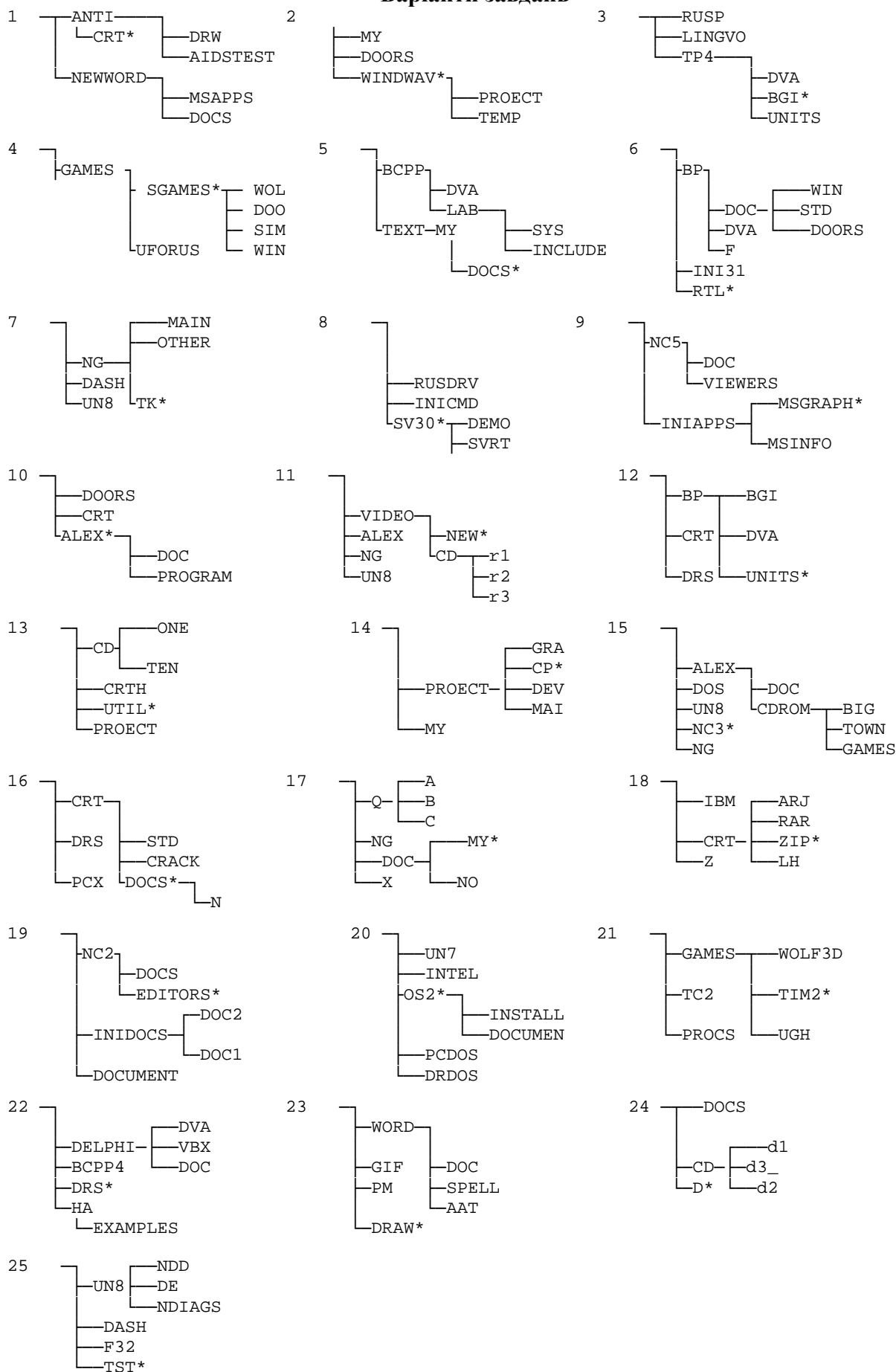
1. Графічний інтерфейс користувача WINDOWS.
2. Вікно, типи вікон, операції з вікнами.
3. Лінійки прокручування, їхнє призначення і використання.
4. Робота з програмним додатком "Довідка".
5. Налаштування графічного інтерфейсу користувача.
6. Що таке робочий стіл?
7. Назвіть елементи робочого стола.
8. Назвіть елементи керування вікном.
9. Як можна розташувати вікна на робочому столі?
10. Які стандартні програми Windows Ви знаєте?

1.1.2 Лабораторна робота та практичне заняття № 2. Програма «Провідник». Робота з файловою системою

Порядок виконання роботи

1. Вибрати свій варіант ("дерево папок") відповідно до Вашого номера в журналі в Варіантах завдань.
2. У папці, названій Вашим прізвищем, створити дерево каталогів за Вашим варіантом.
3. У папку, позначену (*), записати текстовий документ ВАШЕ_ІМ'Я.TXT, що містить наступну інформацію:
 - ПІБ (повністю);
 - стаття;
 - дата народження;
 - місце народження.
4. Створіть текстовий документ ВАШЕ_ІМ'Я1.TXT, який містить наступну інформацію:
 - домашня адреса;
 - ПІБ батьків і їхнє місце роботи;
 - який навчальний заклад закінчив;
 - захоплення.
5. Перемістіть файл ВАШЕ_ІМ'Я1.TXT у папку, позначену (*).
6. перейменувати файл ВАШЕ_ІМ'Я1.TXT у файл ВАШЕ_ПРІЗВИЩЕ.TXT.
7. Відкрити файл ВАШЕ_ПРІЗВИЩЕ.TXT і скопіювати в кінець файлу інформацію з файлу ВАШЕ_ІМ'Я.TXT.
8. Зберегти файл під тим же ім'ям.
9. Записати в зошиті хід роботи, "дерево каталогів", список використовуваних команд, шлях до текстових файлів і вміст файлів.
10. Перейти в кореневий каталог диска і за допомогою кнопки «Пошук» знайти всі файли, у яких ім'ям є Ваше прізвище.
11. За допомогою кнопки «Пошук» знайти на диску всі текстові файли, імена яких починаються з першої букви Вашого прізвища.
12. Результати пошуку записати у зошит.

Варианти завдань



Контрольні запитання

1. Як запустити програму «Провідник»?
2. Як вивести дерево папок?
3. Що означає знак «+» перед ім'ям папки?
4. Як змінити вигляд відображуваної інформації у правому вікні «Провідника»?
5. За якими параметрами і як можна впорядкувати інформацію в правому вікні «Провідника»?
6. Поясніть призначення основних кнопок програми «Провідник».
7. Як створити новий файл? Скільки способів Ви знаєте?
8. Як створити нову папку? Скільки способів Ви знаєте?
9. Як скопіювати об'єкт за допомогою клавіатури?
10. Як перемістити об'єкт за допомогою клавіатури?
11. Як скопіювати об'єкт за допомогою контекстного меню?
12. Як перемістити об'єкт за допомогою контекстного меню?
13. Скільки способів переміщення й копіювання Ви знаєте?
14. У чому різниця між переміщенням і копіюванням?
15. Що таке перейменування і як його здійснити?
16. У чому основна відмінність між папкою й файлом?
17. Що таке шаблон файлу?
18. Що замінюють знаки «*» і «?» у шаблоні файлу?
19. Які об'єкти можна шукати за допомогою кнопки «Пошук» програми «Провідник»?
20. Що таке повне ім'я файлу? Із чого воно складається?
21. Назвіть відомі Вам розширення файлів.

1.1.3 Лабораторна робота та практичне заняття № 3. Стандартні програми WINDOWS

Розглянемо деякі програми, які входять у стандартний пакет WINDOWS. Це програми «Блокнот», графічний редактор Paint і програма «Калькулятор».

Порядок виконання роботи

1. Створіть папку ПІБ_Родина, у якій відобразить Ваше сімейне дерево, використовуючи три рівні вкладення й давши папкам відповідні імена, наприклад, «МОЯ_МАМА», «МАМИН_БАТЬКО» і т. д.
2. У кожній папці створіть файл WordPad із короткою біографією свого родича, інформацією про його захоплення, місце роботи, домашніх вихованців і т. д., давши йому відповідне ім'я.
3. У файл додайте порожній перший рядок і додайте поточну дату та час.
4. У кожній папці створіть малюнок за допомогою додатка Paint, що характеризує захоплення вашого родича, використавши обов'язково коло й квадрат різних кольорів з ефектом прозорості.
5. У Ваш файл WordPad додайте свій малюнок, зберігши його під ім'ям «Автобіографія1».

Контрольні запитання

1. Як запустити програми «Блокнот», Paint, «Калькулятор»?
2. Як додати в текстовий файл поточну дату та час?
3. Чим відрізняються програми «Блокнот» та WordPad?
4. Як намалювати коло в програмі Paint?
5. Як намалювати квадрат у програмі Paint?
6. Як зробити заливання кольором у програмі Paint?
7. Як додати малюнок у текстовий файл?
8. Які режими має програма «Калькулятор»?
9. Які операції можна здійснювати в інженерному режимі програми «Калькулятор»?

1.2 Текстовий процесор Microsoft Word

1.2.1 Лабораторна робота та практичне заняття № 1.

Створення та збереження документів у текстовому процесорі Word.

Введення, редагування і форматування тексту

Порядок виконання роботи

1. Запустіть програму **Word** (**Пуск** > **Програми** > **Microsoft Word**).
2. Виконайте команду **Файл** > **Зберегти як** і збережіть документ у папці, ім'я якої відповідає назві вашої групи. Як ім'я файлу введіть своє прізвище.
3. Введіть первинний текст (див. нижче).
4. Перетворіть текст до вигляду, представленому в розділі «Результати роботи».
5. Збережіть створений документ.

Первинний текст

Що таке текстовий процесор

Текстовий процесор – це програма обробки тексту, що використовується для створення різних документів (наприклад, листів, звітів, бюлетенів). Microsoft Word дозволяє вводити, редагувати і формувати текст. За допомогою цієї програми ви можете вставляти в документ малюнки, таблиці й діаграми, а також автоматично виправляти орфографічні та граматичні помилки.

Microsoft Word має можливості, що дозволяють значно спростити процес створення й редагування документів, зробити його високопродуктивним. Далі приведений короткий огляд деяких часто використовуваних функцій.

Коли при введенні тексту курсор виявляється наприкінці рядка, Microsoft Word автоматично переводить його на наступний рядок. Тому наприкінці рядків не треба натискати клавішу **Enter** (окрім тих випадків, коли ви хочете почати новий абзац або створити додатковий порожній рядок).

Якщо при введенні тексту ви зробите помилку, функція автокорекції автоматично її виправить, а функція автоматичної перевірки орфографії підкреслить неправильно написані слова червоною хвилястою лінією, щоб ви звернули на них увагу і виправили.

Для представлення тексту у вигляді таблиці Microsoft Word пропонує дуже ефективні засоби. Якщо ж ваша таблиця містить цифрові дані, ви можете легко перетворити її у діаграму.

Режим попереднього перегляду дозволяє представити документ практично в тому вигляді, у якому він вийде друком. Крім того, режим дає можливість відобразити відразу декілька або всі сторінки, що зручно для внесення змін перед роздруковкою.

Програма пропонує також низку наступних корисних функцій.

Автотекст – для збереження та вставки часто уживаних слів, фраз або малюнків.

Стилі – для збереження і присвоєння відразу цілих наборів форматів.

Злиття – для створення серійних листів, роздруківки конвертів і етикеток.

Макроси – для виконання послідовності часто використовуваних команд.

«Майстри» – для створення професійно оформлених документів.

Результати роботи

Що таке текстовий процесор

<p>Текстовий процесор – це програма обробки тексту, що використовується для створення різних документів (наприклад, листів, звітів, бюлетенів). Microsoft Word дозволяє вводити, редагувати і формувати текст. За допомогою цієї програми ви можете вставляти в документ малюнки, таблиці та діаграми, а також автоматично виправляти орфографічні та граматичні помилки.</p>
--

Microsoft Word має можливості, що дозволяють значно спростити процес створення й редагування документів, зробити його високопродуктивним. Далі приведений короткий огляд деяких часто використовуваних функцій.

■ Якщо при введенні тексту ви зробите помилку, функція автокорекції автоматично її виправить, а функція автоматичної перевірки орфографії підкреслить неправильно написані слова червоною хвилястою лінією, щоб ви звернули на них увагу і виправили.

■ Коли при введенні тексту курсор виявляється наприкінці рядка, *Microsoft Word* автоматично переводить його на наступний рядок. Тому наприкінці рядків не треба натискати клавішу Enter (крім тих випадків, коли ви хочете почати новий абзац або створити додатковий порожній рядок).

■ Режим попереднього перегляду дозволяє представити документ практично в тому вигляді, у якому він вийде друком. Крім того, режим дає можливість відобразити відразу декілька або всі сторінки, що зручно для внесення змін перед роздруківкою.

■ Для представлення тексту у вигляді таблиці *Microsoft Word* пропонує дуже ефективні засоби. Якщо ж ваша таблиця містить цифрові дані, ви можете легко перетворити її у діаграму.

Програма пропонує також низку наступних корисних функцій:

1. «Майстри» – для створення професійно оформлених документів.
2. Макроси – для виконання послідовності часто використовуваних команд.
3. Автотекст – для збереження і вставки часто уживаних слів, фраз або малюнків.
4. Стилі – для збереження та присвоєння відразу цілих наборів форматів.
5. Злиття – для створення серійних листів, роздруківки конвертів і етикеток.

Контрольні запитання

1. Призначення і способи запуску текстового процесора Word.
2. Вікно програми Word і його елементи.
3. Режими відображення документів у вікні Word.
4. Створення, відкриття і збереження документів Word.
5. Загальні принципи виконання будь-яких дій над елементами документу.
6. Прийоми виділення елементів тексту.
7. Введення тексту і його редагування.
8. Форматування тексту.
9. Копіювання форматів.
10. Установлення полів за допомогою маркерів лінійки.

1.2.2 Лабораторна робота та практичне заняття № 2.

Пошук і заміна тексту. Автокорекція та автотекст

Порядок виконання роботи

1. Відкрийте папку, ім'я якої відповідає назві вашої групи. У цій папці є документ **Текст**. Створіть копію цього документа в тій же папці. Як ім'я цієї отриманої копії введіть своє прізвище.

2. На стор. 1 замінити словосполучення *Microsoft Word* на словосполучення «текстовий процесор» (із підкресленням подвійною хвилястою лінією)

3. На стор. 2 замінити формат «Курсив» на формат «Напівжирний підкреслений».

4. Створити елемент автокорекції mw, що відповідає словосполученню *Microsoft Word*. Додати в кінець документа наступний текст (користуючись елементом автокорекції):

Microsoft Word припускає різноманітне налаштування параметрів. У результаті активізація однієї або декількох опцій у діалогових вікнах може вплинути на вигляд екрана або на роботу деяких функцій.

Microsoft Word дозволяє налаштувати вікно програми відповідно до вашого стилю роботи та пріоритетів. Якщо по ходу уроку ви зміните вигляд екрана і вийдете з *Microsoft Word*, наступного разу, коли ви відкриєте вікно програми, воно стане в тому вигляді, у якому ви його залишили наприкінці останнього сеансу роботи.

5. Перевірте правопис доданого тексту за допомогою команди **Правопис** із пункту меню **Сервіс**.

6. Створіть елементи автотексту для введення близьких номерів телефонів і введіть у кінець документа наступний текст:

Список номерів телефонів, за якими можна одержати консультації:

(095)-333-66-71 (095)-333-66-72 (095)-333-66-73 (095)-333-66-74

7. Збережіть результати роботи.

Контрольні запитання

1. Пошук тексту.
2. Заміна тексту.
3. Автокореція.
4. Автотекст.
5. Перевірка правопису.
6. Підбір синонімів.

1.2.3 Лабораторна робота та практичне заняття № 3.

Підготовка та виведення документа на друк

Порядок виконання роботи

1. Відкрийте папку, ім'я якої відповідає назві вашої групи. У цій папці є документ **Текст**. Створіть копію цього документа в тій же папці. Як ім'я цієї отриманої копії введіть своє прізвище.
2. Встановіть параметри сторінки:
 - орієнтація – альбомна;
 - розміри полів: 2 см;
 - дві сторінки на аркуші;
 - застосувати до всього документа.
3. Розбити текст на два розділи. Кожний розділ починається словами: «Що таке текстовий процесор». Розділ починається з нової сторінки.
4. Установити нумерацію розділів. Для цього заголовку кожного розділу привласнити стиль «Заголовок 1». Виконати команду **Список** із меню **Формат**. На вкладці **Багаторівневий** вибрати другий тип у другому ряді.
5. Ввести виноски для кожної з назв розділу. У виносці записати слово «Розділ» і номер Розділу.
6. Ввести колонтитули:
 - верхній: дата (ліворуч), номер сторінки із вказівкою розділу (праворуч). Ці операції виконати за допомогою команд **Дата й час**, **Номер сторінки** з меню **Вставка**;
 - нижній: прізвище, ім'я, по батькові виконавця (посередині).
7. Виконати команду **Попередній перегляд**.
8. Зберегти документ.

Контрольні запитання

1. Вставка роздільників сторінок і розбивка документа на розділи.
2. Встановлення параметрів сторінки.
3. Колонтитули.
4. Вставка номерів сторінок, дати і часу, виноска із пункту меню **Вставка**.
5. Попередній перегляд документа.
6. Виведення документа на друк.

1.2.4 Лабораторна робота та практичне заняття № 4. Текстові колонки

Порядок виконання роботи

1. Відкрийте папку, ім'я якої відповідає назві вашої групи. У цій папці є документ **Текст**. Створіть копію цього документа в тій же папці. Як ім'я цієї отриманої копії введіть власне прізвище.
2. Видалити другу сторінку. Зробіть розрідженим шрифт у назві сторінки, що залишилася. Установіть колір напису назви – червоний.
3. Із тексту в рамці залишити слова «Текстовий редактор – це програма обробки тексту».
4. Скасувати списки.
5. Розбити документ на розділи. Другий розділ починається словами «Якщо при введенні...». Третій розділ складається з речення: «Програма пропонує також низку

наступних корисних функцій». Частина тексту, що залишилася, складає четвертий розділ. Усі розділи розташовуються на одній сторінці.

6. У другому розділі текст розташувати у двох колонках.

7. У четвертому розділі текст розташувати в п'ятьох колонках. Столпчики починаються словами: Майстри, Макроси, Автотекст, Стили, Злиття.

8. Зберегти результати роботи.

9. Результати роботи повинні мати такий вигляд, як на сторінці, що розташована після контрольних питань.

Контрольні запитання

1. Чим відрізняється представлення колонок у звичайному режимі від представлення в режимі розмітки?

2. Як розбити текст на колонки, використовуючи кнопку **Колонки** панелі інструментів

Стандартна?

3. Як забезпечити розбивку різних частин документа на різне число столпчиків?

4. Як вставити вручну розрив у колонку?

5. Як змінити ширину столпчиків і інтервал між ними?

6. Як встановити вертикальні роздільники між колонками?

Що таке текстовий процесор

Текстовий процесор – це програма обробки тексту.

Microsoft Word має можливості, що дозволяють значно спростити процес створення й редагування документів, зробити його високопродуктивним. Далі приведений короткий огляд деяких часто використовуваних функцій.

Якщо при введенні тексту ви зробите помилку, функція автокорекції автоматично виправить, а функція автоматичної перевірки орфографії підкреслить неправильно написані слова червоною хвилястою лінією, щоб ви звернули на них увагу і виправили.

Режим попереднього перегляду дозволяє представити документ практично в тому вигляді, у якому він вийде друком. Крім того, режим дає можливість відобразити відразу декілька або всі сторінки, що зручно для внесення змін перед роздрукуванням.

Коли при введенні тексту курсор виявляється наприкінці рядка, *Microsoft Word* автоматично переводить його на наступний рядок. Тому наприкінці рядків не треба натискати клавішу Enter (крім тих випадків, коли ви хочете почати новий абзац або створити додатковий порожній рядок).

Для представлення тексту у вигляді таблиці *Microsoft Word* пропонує дуже ефективні засоби. Якщо ж ваша таблиця містить цифрові дані, ви можете легко перетворити її у діаграму.

Програма пропонує також низку наступних корисних функцій

« <i>Майстри</i> » – для створення професійно оформлених документів	<i>Макроси</i> – для виконання послідовності часто використовуваних команд.	<i>Автотекст</i> – для збереження й вставки часто уживаних слів, фраз або малюнків.	<i>Стили</i> – для збереження й присвоєння відразу цілих наборів форматів.	<i>Злиття</i> – для створення серійних листів, роздруківки конвертів і етикеток.
---	---	---	--	--

1.2.5 Лабораторна робота та практичне заняття № 5.

Робота з таблицями

Порядок виконання роботи

1. Відкрийте папку, ім'я якої відповідає назві вашої групи. У цій папці створіть документ. Як ім'я цього документа введіть своє прізвище.
2. На першій сторінці створіть бланк квитанції оплати за комунальні послуги.

ОПЛАТА ЗА КОМУНАЛЬНІ ПОСЛУГИ

Особистий рахунок

Прізвище, ім'я, по батькові _____

Адреса: вул. _____ будинок _____ кв. _____

Пільга % – _____

Вид платежу	Міс.	Рік	Сума	Показання лічильника		Різн.
01 електроенергія						
02 квартплата						
03 опалення						
04 гаряча вода						
05 холодна вода						
06 каналізація						
07 газ варильний						
08 газ опалювальний						
інші						

Усього _____

Підпис платника _____

3. На другій сторінці створіть таблицю, як показано нижче:

			3	6	9
2		3		4	
9		8	5		
				3	4
	2			1	5
	3	10			

Контрольні запитання

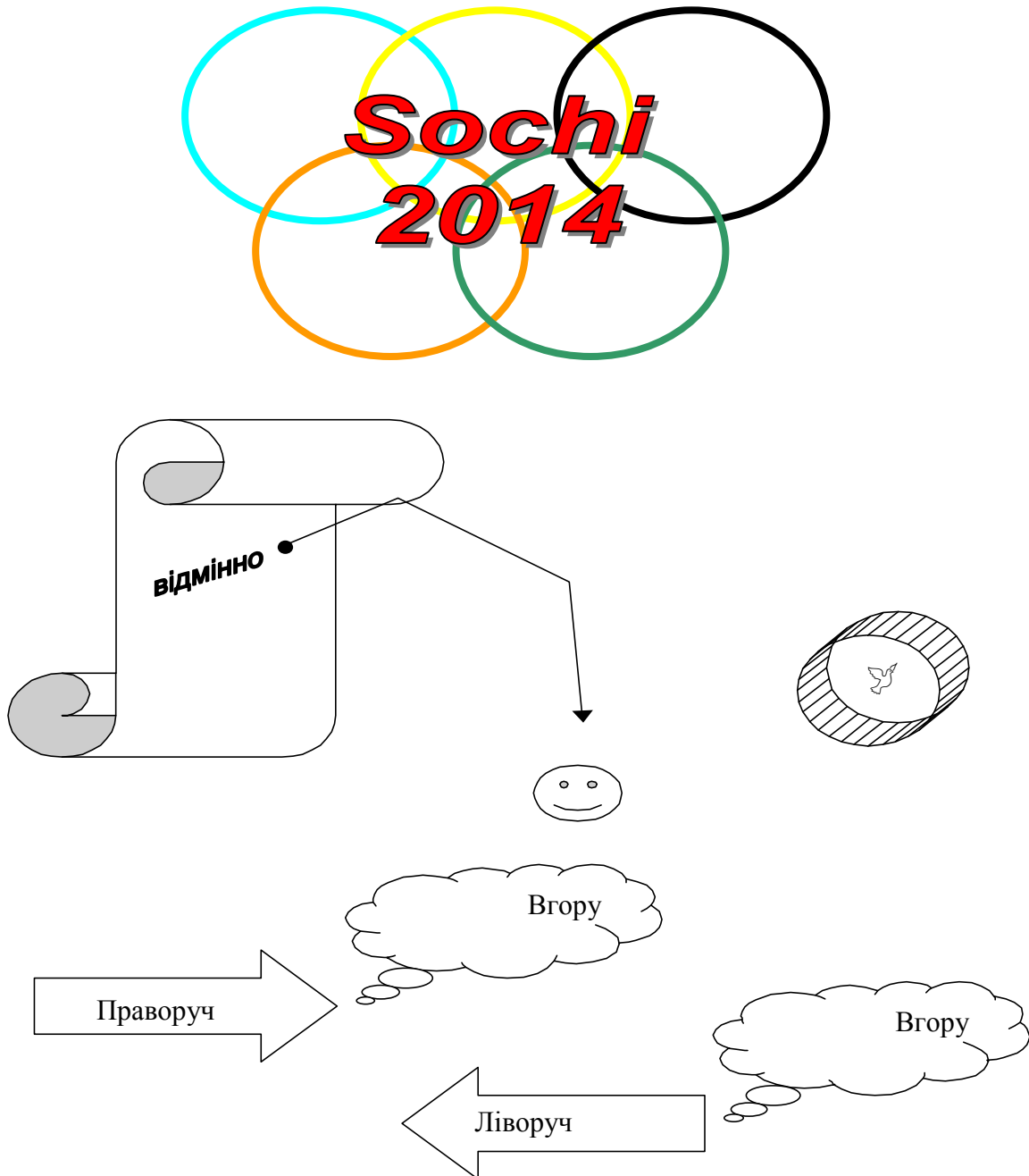
1. Способи створення і заповнення таблиць.
2. Виділення таблиці та її елементів.
3. Команди для роботи з таблицями.
4. Зміна ширини і висоти комірки.
5. Форматування таблиць.

1.2.6 Лабораторна робота та практичне заняття № 6.

Робота з графічними об'єктами

Порядок виконання роботи

1. Відкрийте папку, ім'я якої відповідає назві вашої групи. У цій папці створіть документ. Як ім'я документа введіть своє прізвище.
2. На сторінці розташуйте малюнки, як показано нижче.



Контрольні запитання

1. Малювання фігур і ліній.
2. Вставка автофігури.
3. Вставка тексту й об'єктів ClipArt та WordArt.
4. Виділення графічних об'єктів і зміна їхніх розмірів.
5. Редагування графічних об'єктів за допомогою команд меню Дії.
6. Форматування графічних об'єктів.
7. Створення фігур з ефектом об'єму.

1.3 Створення презентацій за допомогою Power Point

1.3.1 Лабораторна робота та практичне заняття №1.

Знайомство з презентаціями

Порядок виконання роботи

1. У папці **Приклад** запустіть навчальну презентацію «Введення в бази даних1».
2. Зверніть увагу на об'єкти, які можуть бути вмістом слайду.
3. Завершіть показ слайдів.
4. Перейдіть в папку **Приклад** і відкрийте навчальну презентацію «Введення в бази даних1».
5. Відкрийте **Довідку Microsoft PowerPoint**.
6. Перегляньте розділ Довідки *Робота в різних режимах ->Режими PowerPoint*.
7. Запишіть у зошит головні опції.
8. Увімкніть різні режими для відкритої презентації.
9. Перегляньте розділ Довідки **Робота зі слайдами**.
10. Створіть презентацію «Ваше прізвище» й скопіюйте в неї три перших слайди навчальної презентації.
11. Перегляньте розділи Довідки **”Додавання й форматування тексту”, ”Маркіровані й нумеровані списки”, ”Малювання й робота з об'єктами”**.
12. Запишіть у зошит головні опції.
13. Створіть свій варіант перших трьох слайдів, описуючи відомий Вам додаток Windows ”Блокнот”. Це треба виконати в презентації «Ваше прізвище».
14. Перегляньте розділи Довідки **”Підготовка показів слайдів”, ”Запуск і керування показом слайдів”**.
15. Налаштуйте різні стилі показу слайдів презентації «Ваше прізвище».
16. Збережіть файл презентації та покажіть свою роботу викладачеві.

1.3.2 Лабораторна робота та практичне заняття № 2.

Керування об'єктами на слайді

Порядок виконання роботи

1. Увійдіть в папку *Приклад* і відкрийте навчальну презентацію «Введення в бази даних1».
2. Знайдіть об'єкти: напис, малюнок, організаційна діаграма, керуюча кнопка.
3. Подивіться властивості цих об'єктів, виділяючи їх і клацаючи опцію *властивості* на правій кнопці миші.
4. Відкрийте презентацію «Ваше прізвище» і додайте новий слайд (меню Вставка → Новий слайд), використовуючи будь-який макет.
5. Потренуйтеся у додаванні та видаленні об'єктів на слайді (керуючі кнопки *Автофігури* → *Керуючі кнопки*).
6. Перейдіть у папку *Приклад* і відкрийте навчальну презентацію «Введення в бази даних1».
7. Перегляньте налаштування вікна *Налаштування дії всіх керуючих кнопок*.
8. Відкрийте презентацію «Ваше прізвище» і додайте п'ять нових слайдів (меню Вставка → Новий слайд), використовуючи будь-який макет.
9. На кожен слайд поставте керуючу кнопку, яка викликає перехід до наступного слайда.
10. На кожен слайд поставте керуючу кнопку, що викликає перехід до файлу завдання, який Ви читаєте.
11. Заповніть нові слайди написами й малюнками з описом розпорядку свого навчального дня.
12. Збережіть файл презентації та покажіть свою роботу викладачеві.

1.4 Основи електронних таблиць EXCEL

1.4.1 Лабораторна робота та практичне заняття №1.

Введення, редагування і форматування даних і формул

Порядок виконання роботи

Вправа 1. Обробка даних.

1. Запустіть програму Excel (**Пуск > Програми > Microsoft Excel**).
2. Виконайте команду **Файл > Зберегти як** і збережіть робочу книгу в папці, ім'я якої відповідає назві вашої групи. Як ім'я файлу уведіть своє прізвище.
3. Зробіть поточною клітинку **A1** на **аркуші 1** і введіть в неї заголовок **Перше число**.
4. Введіть **1, 3, 5, 9, 11** у послідовні комірки стовпчика **A**, починаючи з комірки **A2**.
5. Введіть у комірку **B1** рядок **Друге число**.
6. Введіть **10, 8, 6, 4, 2** у послідовні комірки стовпчика **B**, починаючи з комірки **B2**.
7. Введіть у комірку **C1** рядок **Сума**.
8. Введіть у комірку **D1** рядок **Різниця**.
9. Введіть у комірку **E1** рядок **Добуток**.
10. Введіть у комірку **F1** рядок **Частка**.
11. Введіть у комірку **G1** рядок **Різниця квадратів**.
12. Введіть у комірку **C2** формулу $=A2+U2$.
13. Введіть у комірку **D2** формулу $=A2-U2$.
14. Введіть у комірку **E2** формулу $=A2*U2$.
15. Введіть у комірку **F2** формулу $=A2/U2$.
16. Введіть у комірку **G2** формулу $=A2*A2-U2*B2$.
17. Виділіть протяганням комірки **C2, D2, E2, F2, G2**.
18. Наведіть покажчик миші на маркер заповнення в правому нижньому куті рамки, що охоплює виділений діапазон. Натисніть ліву кнопку миші та перетягніть цей маркер, щоб рамка охопила стільки рядків у стовпчиках **C, D, E, F, G**, скільки маємо чисел у стовпчику **A**.
19. Переконайтеся, що формули автоматично модифікуються так, щоб працювати зі значеннями комірок у стовпчиках **A** і **B** поточного рядка.
20. Виділіть комірки **F2:F6**. Виконайте команду **Формат > Комірка**. У діалоговому вікні, що з'явилося, виберіть вкладку **Число**. Встановіть формат **Числовий**, число десяткових знаків – **2**.
21. Двічі клацніть на границі між заголовками стовпчиків **A** і **B**, **B** і **C**, **C** і **D**, **D** і **E**, **E** і **F**. Зверніть увагу, як при цьому змінюється ширина стовпчиків.
22. Змініть у клітинці **A6** число **11** на число **7** і переконайтеся, що відповідні значення у стовпчиках **C, D, E, F, G** у цьому ж рядку були автоматично перераховані.
23. Збережіть робочу книгу, клацнувши на кнопці **Зберегти** панелі інструментів **Стандартна**.

Вправа 2. Використання абсолютного посилання.

1. Зробіть поточний **Лист 2**.
2. У комірку **A2** введіть рядок **Гірські вершини**.
3. У комірку **B2** введіть рядок **Висота (м)**.
4. У комірку **C2** введіть рядок **Висота (км)**.
5. У комірку **A3** введіть рядок **Джомолунгма**.
6. У комірку **A4** введіть рядок **Мак-Кінлі**.
7. У комірку **A5** введіть рядок **Ельбрус**.
8. У комірку **A6** введіть рядок **Арарат**.
9. У комірку **A7** введіть рядок **Монблан**.
10. Встановіть ширину стовпчиків **A, B, C**, щоб у комірках містилися всі назви.
11. У послідовні комірки стовпця **B**, починаючи з комірки **B3** введіть числа **8848, 6193, 5642, 5165, 4810**.
12. У комірку **E2** введіть рядок **Коефіцієнт масштабування =**.

13. У комірку **G2** введіть значення **0,001**.
14. Введіть у комірку **C3** формулу **=U3*N2**. Натисніть на клавішу **F4**. Переконайтеся, що формула тепер виглядає як **=U3*\$H\$2** і тільки потім натисніть на клавішу **Enter**.
15. Використайте метод автозаповнення, щоб скопіювати цю формулу в комірки стовпчика **C**, що відповідають заповненим коміркам стовпчика **A**.
16. Встановіть у комірках **A2:C2** шрифт **Курсив**.
17. Встановіть у комірках **A3:A7** шрифт **Напівжирний, Курсив**.
18. Встановіть у комірках **B3:B7** колір шрифту **Червоний**.
19. Встановіть у комірках **C3:C7** число десяткових знаків 2, колір шрифту Синій.
20. Виділіть методом протягання діапазон **A2:37**. Клацніть на кнопці, що розкриває, яка розташована поруч із кнопкою Границі на панелі інструментів **Форматування**, задайте для цих комірок широку зовнішню рамку (кнопка в правому нижньому куті палітри, що відкрилася.).
21. Виділіть діапазон **E2:H2**. Встановіть колір шрифту Червоний і задайте для цих комірок широку зовнішню рамку.
22. Збережіть робочу книгу.

Вправа 3. Підготовка і форматування прайс-листа

1. Зробіть поточним **Аркуш 3**.
2. У комірку **A1** введіть текст **Прейскурант**.
3. У комірку **A2** введіть текст **Курс перерахування**. У комірку **B2** введіть текст **1\$=**. У комірку **C2** введіть поточний курс перерахування (число), установіть формат осередку **Грошовий**, число десяткових знаків – 2, позначення **грн. Український**.
4. У комірку **A3** введіть текст **Найменування товару**. У комірку **B3** введіть текст **Ціна (грн)**. У комірку **C3** введіть текст **Ціна (\$)**.
5. У наступних п'ять комірок стовпчика **A** введіть назви товарів, що ви хочете включити в прейскурант.
6. Двічі клацніть на границі між заголовками стовпчиків **A** і **B**, **B** і **C**, **C** і **D**.
7. У відповідні комірки стовпчика **B** введіть ціни товарів у гривнях. Встановіть формат осередків **Грошовий**, число десяткових знаків – 2, позначення **грн. Український**.
8. У комірку **C4** введіть формулу: **=B4/\$C\$2**, що використовується для перерахування ціни з гривні в долари. Встановіть формат осередку **Грошовий**, число десяткових знаків – 2, позначення **\$ Англійський (США)**.
9. Методом автозаповнення скопіюйте формули в усі комірки стовпчика **C**, яким відповідають заповнені комірки стовпчиків **A** і **B**.
10. Змініть курс перерахування у комірку **C2**. Зверніть увагу, що всі ціни в доларах при цьому оновлюються автоматично.
11. Виділіть методом протягання діапазон **A1:C1** і виконайте команду **Формат > Комірка**. На вкладці **Вирівнювання** задайте вирівнювання по горизонталі **По центру** й установіть прапорець **Об'єднання комірок**.
12. На вкладці **Шрифт** задайте розмір шрифту в **14** пунктів і в списку **Накреслення** виберіть варіант **Напівжирний**. Клацніть на кнопці **ОК**.
13. Клацніть правою кнопкою миші на **B2** і виберіть у контекстному меню команду **Формат комірок**. Задайте вирівнювання по горизонталі **По правому краю** і клацніть на кнопці **ОК**.
14. Клацніть правою кнопкою миші на комірку **C2** і виберіть у контекстному меню команду **Формат комірок**. Задайте вирівнювання по горизонталі **По лівому краю** і клацніть на кнопці **ОК**.
15. Виділіть методом протягання діапазон **B2:32** і задайте для цих комірок широку зовнішню рамку.

Вправа 4. Створення таблиці обліку переміщення матеріалів.

Використовуючи засоби форматування, побудувати на новому аркуші під назвою *Таблиця* наступну таблицю:

Таблиця обліку переміщення товарів та матеріалів

Ном	Назва	Од. вим.	Залишок		Прихід		Видаток		Перехідний залишок		Ціна
			кільк.	сума	кільк.	сума	кільк.	сума	кільк.	сума	
1.											
...											
7.											
Усього											

Посада

Ваше прізвище (повністю)

1. Дані таблиці задати наступним чином:

- кількість матеріалів у списку – 7;
- назву матеріалів, кількість, ціну, прихід, видаток і перехідний залишок задати за власним розсудом;
- сума = кількість * ціна (формула);
- залишок = перехідний залишок + прихід – видаток (формула).

2. Використовуючи команду «Формат», встановити в комітках таблиці, в яких є грошові суми, числовий (грошовий, фінансовий) формат числа з двома знаками після коми без позначення грошових одиниць.

3. У рядку *Усього* одержати підсумкові дані за відповідними діапазонами.

4. Збережіть робочу книгу.

Контрольні запитання

1. Призначення і можливості Excel.
2. Вікно Excel і його елементи.
3. Типи даних.
4. Введення тексту і чисел.
5. Використання формул. Посилання на комірки.
6. Абсолютні й відносні посилання.
7. Методи копіювання і переміщення комірок.
8. Команда «Спеціальна вставка» і її можливості.
9. Як вставляти і видаляти комірки, рядки і стовпчики.
10. Автозавершення.
11. Автозаповнення числами і формулами.
12. Інструменти форматування даних.
13. Розташування довгих підписів у комітках.
14. Об'єднання комірок.

1.4.2 Лабораторна робота та практичне заняття № 2.

Використання функцій. Робота з масивами даних

Порядок виконання роботи

Вправа 1. Використання функцій.

1. Створіть робочу книгу Excel у папці, ім'я якої відповідає назві вашої групи. Як ім'я робочої книги введіть своє прізвище.

2. У комірки **A2:D5** Аркуша 1 введіть таблицю чисел.

12	1	11	5
7	13	2	3
6	8	15	14
9	10	4	16

Установіть колір шрифту чисел – зелений.

3. Уведіть текст: у комірку F1 – **логарифм**; у комірку G1 – **експонента**; у комірку H1 – **синус**. Установіть колір шрифту – синій.

4. У комірку F2 уведіть формулу **=LN(A2)**. Методом автозаповнення скопіюйте формулу у комірки F3:F5.

5. У комітках **G2:G5** обчисліть значення функції **EXP** для значень, записаних у комітках **B2:B5**.

6. У комірку **H2** уведіть формулу **=SIN(C2/D2)**. Методом автозаповнення скопіюйте цю формулу в комірки **H3:H5**.

7. Комірки **F2:H5** виділити жовтим кольором, а також задати для них границі.

8. Уведіть текст: у комірку A7 – **сума**; у комірку B7 – **мінімум**; у комірку C7 – **максимум**, у комірку D7 – **середнє**. Встановіть шрифт – напівжирний.

9. У комірку A8 уведіть формулу **=СУМ(A2:A5)**

10. За допомогою підсумкових функцій **МІН**, **МАКС**, **СРЗНАЧ** визначте мінімальне, максимальне і середнє значення для чисел, записаних у комітках стовпчиків B, C, відповідно.

11. У комітках **A8:D8** встановіть колір шрифту – червоний.

Вправа 2. Робота з масивами.

- Зробіть активним **Аркуш 2**.
- У комірки **A1:D4** Аркуша 2 скопіюйте зміст комірок **A2:D5** Аркуша 1.
- Використовуючи формулу масиву, помножте елементи діапазону **A1:D4** на 2, результати запишіть у діапазон **A6:D9**.
- Розділіть елементи діапазону **A1:D4** на 5, результати запишіть у діапазон **A11:D15**.
- Додайте до кожного елемента діапазону **A1:D4** число 5, результати запишіть у діапазон **F1:I4**.
- Відніміть від кожного елемента діапазону **A1:D4** число 5, результати запишіть у діапазон **K1:N4**.
- Поелементно додайте масиви **F1:I4** і **K1:N4**, результати запишіть у діапазон **F6:I9**.
- Від елементів масиву **F1:I4** відніміть відповідні елементи масиву **K1:N4**, результати запишіть у діапазон **K6:N9**.
- За допомогою функції **МОБР** обчислити матрицю, яка є зворотною до записаної в діапазоні **A1:D4**, результати запишіть у діапазон.
- Використовуючи функцію **МУМНОЖ** у діапазон **K11:I14**, запишіть добуток матриць, розташованих у комітках **A1:D4** і **F11:I14**.

Вправа 3. Розв'язання системи лінійних рівнянь.

- Зробіть активним **Аркуш 3**. Вирішується система, що складається з чотирьох рівнянь з чотирма невідомими. Коефіцієнти при невідомих записуються в масив **A1:D4**, стовпчик вільних членів – у масив **F1:F4**, результат – у масив **H1:H4**.
- У масив **A1:D4** уведіть формулу масиву, в якій до кожного елемента масиву **A1:D4** Аркуша 2 необхідно додати число, рівне половині порядкового номера студента у списку групи.
- У комірки масиву **F1:F4** уведіть відповідно числа **1, 2, 3, 4**.
- У комірки масиву **H1:H4** уведіть формулу масиву **=МУМНОЖ(МОБР(A1:D4),F1:F4)**, з огляду на особливості роботи з вкладеними функціями.

5. У комірки **J1:J4** запишемо результат перевірки отриманого вирішення. Для цього в ці комірки введемо формулу масиву, у якій треба перемножити матрицю **A1:D4** і вектор-стовпчик **H1:H4**.

Контрольні запитання

1. Введення функцій.
2. Які списки містить діалог **Майстер функцій**? Яке призначення цих списків?
3. Робота з палітрою формул.
4. Функції для підсумкових обчислень. Особливості роботи з цими функціями.
5. Вкладені функції. Особливості введення вкладених функцій.
6. Визначення формули масиву. Правила введення формули масиву.
7. Які операції можна виконати над масивами?
8. Функції для роботи з матрицями.

1.4.3 Лабораторна робота та практичне заняття № 3.

Побудова і редагування діаграм

Порядок виконання роботи

Вправа 1. Побудова діаграм.

1. Створити робочу книгу Excel у папці, ім'я якої відповідає назві вашої групи. Як ім'я робочої книги ввести своє прізвище. Двічі клацніть на ярличку **Аркуш1** і дайте цьому робочому аркушу ім'я **Впр1**.

2. Побудувати таблицю

	2010	2011	2012	2013
S1	3	16	12	19
S2	7	11	18	10
S3	9	22	15	17
S4	5	14	7	15
S5	14	8	5	11
S6	12	9	10	16

3. Побудувати впроваджену діаграму (гістограму) для рядів: S1, S2, S3. Ввести назву діаграми «Діаграма», осі X (категорій) – «Роки», осі Y (значень) – «Значення». Під віссю категорій мають бути проставлені значення категорій.

4. Побудувати кругову діаграму для ряду S6 на окремому аркуші. Указати підписи категорій і часток.

Вправа 2. Редагування діаграми (гістограми).

1. Скопіювати аркуш **Впр1**. Новий аркуш автоматично одержить ім'я **Впр1(2)**. Двічі клацніть на ярличку цього аркуша і дайте йому ім'я **Впр2**. Далі операції виконуються на цьому аркуші.

2. Видалити ряд S3 з діаграми.
3. Додати ряд S4 у діаграму (засобом перетягнення).
4. Видалити ряд S2 з діаграми.
5. Додати ряд S5 у діаграму, використовуючи команди меню.
6. Легенду розмістити внизу діаграми.
7. Збільшити область побудови діаграми.
8. Ввести сітку по осі Y з основними і проміжними розподілами.
9. Для назви діаграми установити розмір шрифту – 10, колір – червоний.
10. Колір осей зробити синім.
11. Для ряду S1 показати тренд. Вивести рівняння тренда на екран.
12. Скопіювати **Впр2**. Новий аркуш автоматично отримає ім'я **Впр2(2)**.
13. Видалити лінію тренда з діаграми на цьому листі.
14. Змінити тип діаграми на об'ємний. Змінити перспективу об'ємної діаграми.

Вправа 3. Побудова експериментального графіка.

1. Виберіть щигликом на ярличку робочий аркуш, що не використовується, або створіть новий (Вставка > Лист). Двічі клацніть на ярличку нового аркуша і перейменуйте його як **Впр3**.
2. У стовпчик А, починаючи з комірки А1, введіть довільний набір значень незалежної перемінної (десять чисел у зростаючому порядку).
3. У стовпчик В, починаючи з комірки В1, введіть довільний набір значень функції (десять довільних чисел).
4. Методом протягання виділіть всі заповнені комірки стовпчиків А і В.
5. Клацніть на значку Майстер діаграм на стандартній панелі інструментів.
6. У списку **Тип** виберіть пункт **Точечна** (для відображення графіка, заданого парами значень). У палітрі **Вигляд** виберіть середній пункт у першому стовпчику (маркери, з'єднані гладкими кривими). Клацніть на кнопці **Далі**.
7. Тому що діапазон комірок був виділений заздалегідь, майстер діаграм автоматично визначає розташування рядів даних. Переконайтеся, що дані на діаграмі обрані правильно. На вкладці **Ряд** у поле **Ім'я** вкажіть: **Результати вимірів**. Клацніть на кнопці **Далі**.
8. Виберіть вкладку **Заголовки**. Переконайтеся, що задана назва ряду даних автоматично використана як заголовок діаграми. Замініть його, увівши в поле **Назва діаграми** заголовок **Експериментальні точки**. Клацніть на кнопці **Далі**.
9. Встановіть перемикач на **Окремому**. За бажанням, задайте довільне ім'я робочого аркуша, що додається. Клацніть на кнопці **Готово**.
10. Переконайтеся, що діаграма побудована і введена в новий робочий аркуш. Розгляньте її і клацніть на побудованій кривій, щоб виділити ряд даних.
11. Дайте команду **Формат > Виділений ряд**. Відкрийте вкладку **Вигляд**.
12. На панелі **Лінія** відкрийте палітру **Колір** і виберіть червоний колір. У списку **Тип лінії** виберіть пунктир.
13. На панелі **Маркер** виберіть у списку **Тип маркера** трикутний маркер. У палітрах **Колір** і **Тло** виберіть зелений колір.
14. Клацніть на кнопці **ОК**, зніміть виділення з ряду даних і подивіться, як змінився вид графіка.
15. Збережіть робочу книгу.

Контрольні запитання

1. Визначення понять діаграма, ряди даних, категорії.
2. Типи і види діаграм.
3. Основні компоненти діаграм.
4. Порядок побудови діаграм.
5. Виділення діаграми та її елементів і зміна їхніх розмірів.
6. Додавання і видалення рядів даних.
7. Форматування елементів діаграм.
8. Побудова лінії тренда і завдання її параметрів.
9. Способи настроювання об'ємної діаграми.
10. Додавання графічних елементів на діаграму і на аркуш.

1.4.4 Лабораторна робота та практичне заняття № 4.

Організація та документування робочих книг.

Виведення на друк робочих аркушів і діаграм

Порядок виконання роботи

Вправа 1.

1. Створити робочу книгу Excel у папці, ім'я якої відповідає назві вашої групи. Як ім'я робочої книги ввести своє прізвище. Двічі клацніть на ярличку **Аркуш1** і дайте цьому робочому аркушу ім'я **Впр1**. Створіть таблицю, яка має вигляд, приведений на малюнку. При заповненні таблиці числами врахуйте, що в кожному рядку послідовність чисел є арифметичною послідовністю.

2. Вставити перед таблицею п'ять рядків.

3. У комірку, в яку записано **1998**, вставити **Примітка**. Текст примітки: «Ряд даних для кругової діаграми». Спосіб відображення примітки: **Примітка й індикатор**.

	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010
s1	10	13	16	19	22	25	28	31	34	37	40	43	46
s2	7	11	15	19	23	27	31	35	39	43	47	51	55
s3	9	12	15	18	21	24	27	30	33	36	39	42	45
s4	5	8	11	14	17	20	23	26	29	32	35	38	41
s5	14	16	18	20	22	24	26	28	30	32	34	36	38
s6	4	8	12	16	20	24	28	32	36	40	44	48	52

4. У комірку, в яке записано **2001**, вставити **Примітка**. Текст примітки: «Ряд даних для розрізної кругової діаграми». Спосіб відображення примітки: **Примітка й індикатор**.

5. Побудувати дві колові діаграми, відповідно до типу зазначеним у примітках, для рядів даних, розташованих у стовпчиках з назвами 1998 і 2001.

6. Побудувати діаграму-графік із з'єднаними точками для рядів, розташованих у рядках з назвами S2 і S3.

7. В області побудови діаграми задати текстове поле, у якому записати ряд S3. Установити розмір шрифту – 10, колір – зелений.

8. Намалювати стрілку від текстового поля до ряду S3.

Вправа 2.

1. Скопіювати аркуш **Впр1**. Переименувати новий аркуш у **Впр2**. Змінити орієнтацію сторінки на альбомну.

2. У комітках, у яких записані значення **1998** і **2001**, видалити примітки.

3. Видалити текстове поле і стрілку на діаграмі.

4. Розмістити зміст робочого аркуша на одній сторінці.

5. Скопіювати аркуш **Впр2**. Аркуш автоматично буде мати ім'я **Впр2(2)**.

6. Повернути книжкову орієнтацію сторінки.

7. Встановити масштаб 100 %.

8. Розмістити зміст робочого аркуша на чотирьох сторінках відповідно за приведеною нижче схемою (для цього необхідно скористатися командою **Розрив сторінки** з меню **Вставка**).

Таблиця (стовпчики 1998–2004)	Продовження таблиці (стовпчики 2005–2010)
Колова діаграма для стовпчика 1998	Діаграма-графік
Колова діаграма для стовпчика 2001	

9. Встановити друк заголовків рядків на кожній сторінці. Виконати команду **Попередній перегляд**.

Вправа 3.

1. Закрити вікно **Попереднього перегляду**.

2. Скопіювати аркуш **Впр2(2)**. Переименувати новий аркуш у **Впр3**.

3. Встановити значення полів: верхнє і нижнє – 5 см, ліве – 2 см і праве – 2,5 см (пункт меню **Файл**, команда **Параметри сторінки**, вкладка **Поля**).

4. Встановити режим виведення сітки і заголовків рядків і стовпчиків.

5. Змінити порядок виводу сторінок з опції вниз – вправо на опцію вправо – униз.

6. Вибрати один із стандартних нижніх колонтитулів.

7. Створити користувальницький верхній колонтитул за наступним зразком.

Дата і час	Прізвище, ім'я, по батькові	Ім'я файлу
------------	-----------------------------	------------

8. Виконати команду **Попередній перегляд**. Закрити вікно **Попереднього перегляду**.

9. Встановити режим друку робочих аркушів без впровадженої діаграм-графік (виділити діаграму, пункт меню **Формат**, команда **Формат об'єкта**, вкладка **Властивості**). Виконати команду **Попередній перегляд**. Закрити вікно **Попереднього перегляду**.

10. Виділити колову діаграму-1998. Виконати команду **Попередній перегляд**. Закрити вікно **Попереднього перегляду**.

Контрольні запитання

1. Вставка, видалення, копіювання і переміщення робочих аркушів.
2. Форматування аркуша.
3. Створення, редагування і видалення приміток.
4. Способи відображення приміток.
5. Копіювання приміток.
6. Як провести розмітку документа по сторінках?
7. Як вказати, на якій кількості аркушів папера варто розмістити таблицю, що друкується?
8. Створення колонтитулів. Користувальницькі колонтитули.
9. Наскрізні рядки і стовпчики: їхнє призначення і завдання.
10. Як задати послідовність виводу сторінок при друку?
11. Розбивка робочого аркуша на сторінки за бажанням користувача.
12. Друк робочих аркушів без впроваджених у них діаграм.

1.5 Мова програмування VBA – Visual Basic for Applications

1.5.1 Лабораторна робота № 1 «Створення і виконання макросів»

Вивчимо: редактор VBA, запис і збереження макросів, написання нового модуля процедури, оператори виведення.

Як записати новий макрос

Перш ніж редагувати макрос за допомогою VBA, вам потрібно спочатку його записати. Розглянемо основні поняття, необхідні для того, щоб навчитися записувати макроси.

Як правило, запис макросу складається з чотирьох кроків.

1. *Створення початкових умов для макросу* має на увазі приведення вашого програмного середовища в точно такий стан, в якому ви припускаєте виконувати записуваний макрос.

2. *Запустити запис макросу і задати йому ім'я*. Як тільки ви запустите запис макросу, ви маєте дати йому відповідне ім'я і вказати, де зберегти цей макрос. Записуючи макрос у програмі Excel 2000, ви можете також призначити йому комбінацію клавіш.

3. *Виконати дії, які ви хочете записати*. Можуть бути записані будь-які дії, які ви виконуєте за допомогою миші або клавіатури, включаючи виконання раніше записаних макросів. Що саме ви запишете, залежить від того, для якого завдання ви припускаєте використовувати певний макрос.

4. *Зупинити запис макросу*. Після зупинки запису ваші дії вже не фіксуються. Новий макрос готовий до застосування відразу після зупинки запису.

Виконаємо послідовно всі чотири кроки.

Перш ніж записувати макрос, ви маєте створити такі самі умови, в яких потім ви його виконуватимете. Припустимо, наприклад, що ви хочете створити макрос, який задаватиме шрифт, кегель і колір тексту у виділених осередках аркуша Excel. Початковими умовами для такого макросу буде відкритий лист Excel з виділеними одним або декількома осередками.

Ви обов'язково маєте створити початкові умови перед початком запису, тому що, якщо ви почнете запис, а потім відкриєте аркуш і почнете виділяти осередки, всі ці дії будуть записані в макросі. У результаті ваш макрос буде вже дуже специфічний — він завжди відкриватиме одну і ту ж книгу, один і той же аркуш і формувати в ньому одні й ті ж осередки. Для того щоб створити універсальний макрос для форматування клітинок, вам потрібно відкрити книгу, вибрати в ній аркуш і виділити осередки до початку запису.

1. Створіть початкові умови для запису макросу. Для цього запустіть Excel, відкрийте нову книгу і заповніть осередки B4:D8 довільним текстом і формулами, а в осередок B2 введіть своє прізвище.

Для того щоб почати запис макросу в програмі Excel 2000, виберіть у меню "Сервіс-макрос-почати запис". Excel розкриє діалогове вікно "Запис макросу". У цьому вікні ви маєте задати ім'я макросу і вказати, де його зберегти. Вказати ім'я і місце збереження потрібно до того, як ви почнете запис макросу. У цьому ж вікні ви можете задати комбінацію клавіш, якою макрос запускатиметься на виконання.

У діалоговому вікні Запис макросу є чотири керівники елемента.

- *Текстове поле "Ім'я макросу"*. Це перший параметр, який вам потрібно вказати в цьому діалоговому вікні. За умовчанням VBA пропонує вам ім'я, що складається із слова "Масо" з подальшою цифрою, яка відповідає порядковому номеру створюваного в цьому сеансі роботи макросу. Краще замінити це ім'я на яке-небудь осмислене. Наприклад, якщо ви створюєте макрос, що будує діаграму продажів на підставі даних поточного аркуша, можете назвати цей макрос "Діагр_продаж".

- *Текстове поле "Поєднання клавіш"*. У цьому полі ви можете задати комбінацію клавіш, за допомогою якої ваш макрос запускатиметься на виконання. Просто введіть у поле відповідний символ на клавіатурі. Цим є сенс користуватися тільки в тому випадку, якщо ви збираєтеся застосовувати свій макрос часто. Усі комбінації клавіш для виклику макросів у програмі Excel складаються з клавіші <Ctrl> і якого-небудь символу. Якщо ви введете в цьому полі букву <a>, комбінацією для виклику макросу буде <Ctrl+a>, якщо ви введете <A>, комбінацією буде <Ctrl+Shift+A>.

- *Список "Зберегти в"*. У цьому списку ви вказуєте, де має бути збережений новий макрос. Можливі варіанти такі: "Особиста книга макросів", "Нова книга", "Ця книга". Якщо ви виберете "Особиста книга" макросів, ваш новий макрос буде записаний у спеціальній книзі з назвою Personal.xls, яка відкривається при кожному запуску програми Excel. Такий вибір доречний тоді, коли ви хочете, щоб новий макрос був доступний при будь-якому сеансі роботи з Excel. Якщо ви виберете "Ця книга", ваш макрос запишеться в поточній відкритій книзі. Цей варіант годиться, якщо ви хочете, щоб макрос був доступний тільки при роботі з цією книгою. Вибравши "Нова книга", ви змусите Excel створити нову книгу і зберегти макрос у ній. При цьому активною залишиться та книга, яка була у вас відкрита перед записом макросу, і всі ваші дії будуть застосовані саме до неї, а не до нової книги. Незалежно від того, в якій книзі ви вирішите зберегти макрос, він буде записаний в ній у вигляді *модуля*.

- *Текстове поле "Опис"*. Інформація в цьому полі макросом не використовується, а служить для вас і тих, хто користуватиметься вашим макросом. Тут дуже корисно вказати, для чого макрос призначений і що він робить. На початку запису Excel поміщає сюди поточну дату й ім'я користувача, яке вказане на вкладці "Загальні" діалогового вікна "Параметри" в меню "Сервіс".

2. Виберіть у меню "Сервіс-макрос-начать запис". Excel розкриє діалогове вікно У текстовому полі "Ім'я макросу" введіть як ім'я макросу "Form12". Залишіть без змін текст, який Excel вставила в поле Опис, але додайте наступне: Форматує текст в осередку шрифтом Arial, напівжирний, 12 пунктів. У списку "Зберегти у" вкажіть, де має бути збережений макрос. Виберіть "Ця книга". Якщо ви впевнені, що часто користуватиметесь цим макросом, можете привласнити йому комбінацію клавіш для швидкого виклику. У такому разі введіть клавішу в поле "Поєднання клавіш" діалогового вікна "Запис макросу". І запишіть цю комбінацію в зошит. Клацніть на кнопці ОК.

Запуститься програма запису макросу і відкриється панель «Зупинити запис».



(зараз запис не зупиняти)

3. Виділіть діапазон осередків B4:D8. Відкрийте діалогове вікно "Формат осередків" і встановіть шрифт Arial, напівжирний, 12 пунктів. Закрийте вікно, клацнувши ОК. Закінчить запис макросу, клацнувши на кнопці "Зупинити запис".

4. Перейдіть на Аркуш2 і заповніть осередки B4:D8 довільною інформацією.

5. Запустіть макрос Form12, вибравши в меню Сервіс – Макрос – Макроси, вибравши ім'я макросу і клацнувши на кнопці "Виконати". Що ви спостерігаєте?

Знайомство з редактором Visual Basic

Для того щоб дізнатися, які модулі записані в певній книзі і подивитися текст цих модулів, вам знадобиться редактор мови Visual Basic. Редактор Visual Basic — це інструмент для створення модулів і проглядання їх вмісту, створення і редагування тексту макросів, створення діалогових вікон і вирішення інших завдань, з якими вам доводиться зустрічатися при роботі з програмами на мові Visual Basic. Надалі ми називатимемо цей редактор просто VB-редактором.

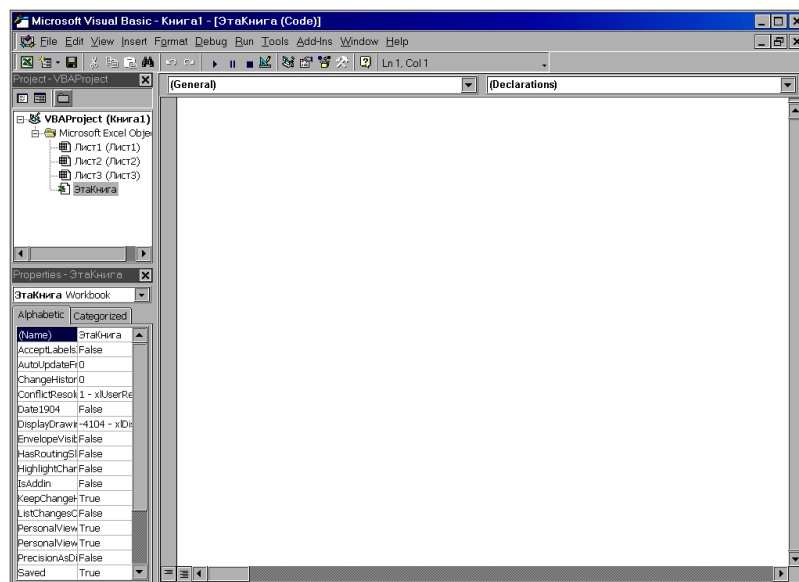
VB-редактор додатка Excel абсолютно аналогічний VB-редакторам інших застосувань, що підтримують мову VBA. Фактично, працюючи з Excel, ви користуєтеся тим самим VB-редактором, що і при роботі з Microsoft Word, Access та іншими додатками Office. Розглянемо, як запустити редактор Visual Basic, пояснимо призначення його вікон і команд меню.

Запускаємо редактор VB

Зробити це можна двома засобами.

- Виберіть у меню "Сервіс-Макрос-редактор Visual Basic".
- Натисніть <Alt+F11>.

Який би засіб ви не вибрали, запуститься редактор Visual Basic. Розгляньте вікно цього редактора.



Вікно редактора Visual Basic містить три дочірні вікна. Кожне з дочірніх вікон надає деяку інформацію про ваш проект (Проектом називаються модулі й інші об'єкти, збережені в робочій книзі Excel або в шаблоні книги). Зазвичай дочірні вікна розміщені біля кордону батьківського вікна (*docked position*) — правої, лівої, верхньої або нижньої. За умовчанням дочірні вікна редактора Visual Basic розташовані у лівій межі.

Приведемо опис вікон редактора Visual Basic.

- Вікно Project (Проект). У вікні Project ви бачите деревоподібну структуру, що описує ваш проект: відкриті в певний момент книги, об'єкти, що містяться в цих книгах (модулі, посилання, форми і т. д.). Це вікно призначене для пошуку модулів й інших об'єктів у вашому проекті.
- Вікно Properties (Властивості). У вікні Properties приведені властивості вибраного в певний момент об'єкта. Іноді властивості об'єкта складаються тільки з його імені. На вкладці Alphabetic (За абеткою) об'єкти приведені в алфавітному порядку. На вкладці Categorized (За категоріями) об'єкти відсортовані за категоріями.
- Вікно Code (Програма). У вікні Code ви бачите текст макросів. Саме у цьому вікні ви читатимете, створюватимете і редагуватимете свої макроси.

За умовчанням вікно Project виглядає подібно вікну програми Explorer і відображає теки вашого проекту. Клацання на знаку "+" ліворуч від елемента розгортає певну гілку "дерева", а клацання на знаку "—" згортає цю гілку.

Для того щоб вибрати об'єкт у вікні Project, потрібно клацнути на цьому об'єкті. У режимі відображення об'єкти згруповані в теках відповідно до їхнього типу - об'єкти Excel, модулі та посилання. Для перемикання режимів відображення клацніть на кнопці Toggle Folders (Теки). У вікні Project, крім кнопки Toggle Folders, є ще дві кнопки: View Code (Програма) і View Object (Об'єкт). Кнопка View Object відображає об'єкт, вибраний у вікні Project, а кнопка View Code відображає текст модуля у вікні Code.

Розгляньте вікно Code, розмір якого збільшений у порівнянні із стандартним так, щоб можна було розмістити в ньому якомога більше тексту і краще були видні різні його властивості. Ви бачите текст макросів, записаних вами.

За умовчанням модуль відображається в режимі Full Module View (Представлення повного модуля). У цьому режимі перегляду у вікні відображається весь модуль зі всіма вхідними в нього макросами. Кожен макрос відділяється від сусідніх тонкою межею. Редактор Visual Basic також дозволяє вам проглядати модуль у режимі Procedure View (Представлення процедури). Макрос іноді називають процедурою. Режим перегляду можна міняти, клацаючи на кнопках у лівому нижньому кутку вікна Code.

6. Запустіть редактор VBA і розгляньте вікна, що відкрилися. У вікні Code виведіть текст вашого макросу Form12. Перемкніть режим на Procedure View. Що змінилося?

7. Розгляньте стандартну панель інструментів редактора VBA. Законспекуйте в зошиті призначення кожної кнопки.

Панелі інструментів редактора Visual Basic

Більшості користувачів легко клацнути мишкою на кнопці, чим вибрати команду з меню. Тому в редакторі Visual Basic для найчастіше використовуваних команд передбачені кнопки на панелі інструментів. Якщо ви інтенсивно працюєте з Visual Basic, то напевно відмітили, що кнопки набагато прискорюють роботу.

Зазвичай Visual Basic відображає тільки одну панель інструментів — стандартну. Але, крім стандартної панелі інструментів, є ще три панелі — Edit, Debug і UserForm.

На панелі інструментів Edit є декілька кнопок, які можуть дуже стати в нагоді при роботі з початковим текстом макросу у вікні Code. Більше того, на цій панелі є навіть декілька кнопок, яких ви не знайдете в меню Edit.

Для того щоб вибрати, яку панель інструментів відображати, а яку ні, ви маєте скористатися командою меню View-Toolbars. Оскільки панель Edit за умовчанням не відображається, вам доведеться вивести її на екран самостійно. Може трапитися так, що хтось прибере з екрану стандартну панель інструментів, тоді вам уручну доведеться відновити на місці і її. Для того щоб прибрати або вивести на екран якусь панель інструментів, зробіть наступне.

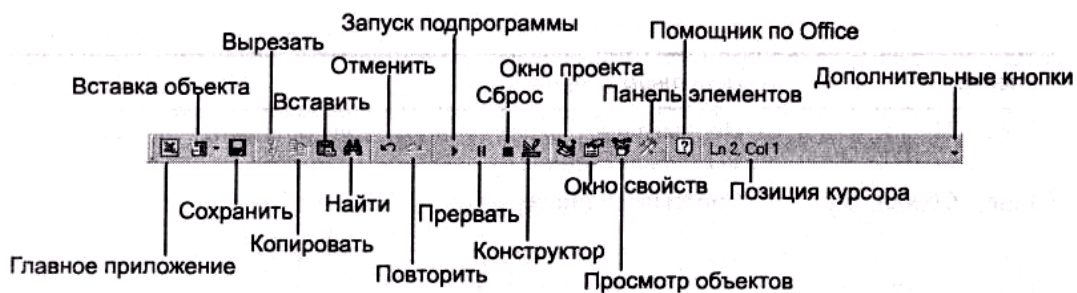
1. Виберіть у меню View-Toolbars. При цьому ви побачите додаткове меню із списком всіх доступних панелей інструментів.

2. Клацніть на імені потрібної панелі інструментів. Наприклад, клацніть на рядку Edit. Visual Basic відобразить вибрану панель.

Зазвичай редактор Visual Basic відображає стандартну панель інструментів притиснутою до верхньої межі вікна. Але ви можете розташувати будь-яку панель інструментів де завгодно, у будь-якому місці вікна. Таку панель інструментів називають *плаваючою*. Вона виглядає як повноцінне вікно, має межі та рядок заголовка. У редакторі Visual Basic з плаваючими панелями поведуться так, як і в програмі Excel або в будь-якій іншій — її можна перетягувати, міняти розміри і форму.

Зараз ми розглянемо команди стандартної панелі інструментів, а пізніше — команди панелі Edit.

Стандартна панель інструментів. На стандартній панелі інструментів у редакторі Visual Basic є 18 кнопок. Кожній кнопці відповідає деяка команда меню. Вивчаючи таблицю, дивіться на вікно редактора, щоб зрозуміти, який малюнок на кнопці відповідає тій або іншій команді.



Кнопки стандартної панелі інструментів

Кнопка	Дія
View (Вигляд) <додаток>	Перемикає вас на початкове застосування, з якого був викликаний редактор Visual Basic. Малюнок на цій кнопці залежить від того, з якого застосування визапускали Visual Basic.
Insert (Вставити)	Клацніть на стрілці праворуч від кнопки, і ви побачите <об'єкт> список об'єктів, які можна вставити у ваш проект. Це об'єкти UserForm, Module, Class Module (Модуль класу) або Procedure. Діє аналогічно меню Insert.
Save (Зберегти)	Зберігає поточний проект. Діє аналогічно до меню File-Save.
Cut (Вирізувати)	Вирізує виділений текст і переносить його в буфер обміну. Діє аналогічно до меню Edit-Cut.
Copy (Копіювати)	Копіює виділений текст і поміщає його в буфер обміну. Діє аналогічно до меню Edit-Copy.
Paste (Вставити)	Вставляє текст із буфера обміну в позицію курсора. Діє аналогічно до меню Edit-Paste.
Find (Знайти)	Розкриває діалогове вікно для пошуку тексту. Діє аналогічно до меню Edit-Find.
Undo (Відмінити)	Відмінює останню команду. Відмінити можна не всяку команду. Діє аналогічно до меню Edit-Undo.
Redo (Повторити)	Повторює останню виконану команду. Діє аналогічно до меню Edit-Redo.
Run (Запуск)	Запускає макрос, який ви зараз редагуєте, тобто той макрос, у тексті якого знаходиться текстовий курсор. Діє аналогічно до меню Run-Sub/UserForm.
Break (Перервати)	Уривається виконання макросу. Діє аналогічно до меню Run-Break.
Reset (Скидання)	Скидає значення всіх змінних модуля і очищає стек викликів. Діє аналогічно до меню Run-Reset.
Design Mode	Включає або відключає режим конструктора. Діє (Конструктор) аналогічно до меню Run-Design Mode.
Project Explorer	Розкриває вікно Project. Діє аналогічно до меню View-(Вікно проекту) Project Explorer.
Properties Window	Розкриває вікно Properties. Діє аналогічно до меню (Вікно властивостей) View-Properties Window.
Object Browser	Розкриває вікно Object Browser. Діє аналогічно до меню (Проглядання об'єктів) View-Object Browser.
Toolbox (Панель)	Розкриває панель інструментів Toolbox. Діє аналогічно до меню – (Проглядання елементів) View-Toolbox.
Office Assistant	Викликає вікно довідки. Діє аналогічно до меню Help-(Помічник по Office) Microsoft Visual Basic Help.
Cursor Position	Ця область на панелі не є командною кнопкою. (Позиція курсора) Тут просто указується позиція курсора в текстовому вікні Code, а саме номер рядка і стовпчика.
More Buttons	Клацнувши на стрілці праворуч від кнопки, ви побачите (Додаткові додаткове меню з командами, для яких кнопки) можна створити кнопки на панелі інструментів.

8. Відредагуйте створений макрос, додавши відрядкові коментарі і своє прізвище, групу і курс.

Редагуємо макрос

Для того щоб почати редагувати макрос, ви маєте спочатку відкрити модуль, що містить цей макрос, і знайти в модулі текст макросу. Наприклад, для того щоб редагувати макрос Forml2, який ви записали, ви повинні спочатку відкрити модуль, в якому зберігається початковий текст макросу Forml2. Для цього виконайте наступне.

1. Відкрийте редактор Visual Basic, натиснувши <Alt+Fll>, якщо він ще не відкритий.
2. Розкрийте вікно Project, якщо воно ще не відкрите. Для цього виберіть у меню View-Project.
3. Знайдіть у деревоподібному списку вікна модуль Project, який ви хочете відобразити. Якщо у вікні Project ввімкнений режим відображення Folders, ви знайдете всі модулі свого проекту в теці Modules.

4. Двічі клацніть на потрібному модулі. При цьому його текст розкриється у вікні Code.

Після цього ви можете скористатися списком Procedure, в якому ви знайдете потрібний макрос.

Для того щоб додати до тексту макросу коментарі, зробіть наступне.

1. Установіть курсор у кінець останнього рядка стандартного коментаря.
2. Установіть курсор у кінець останнього рядка стандартного коментаря.
3. Введіть символ апострофа (') — всі коментарі починаються з цього знаку.
4. Введіть текст коментаря.
5. Збережіть змінений макрос.

Ви маєте отримати код, аналогічний приведеному нижче.

```
Sub Макрос2()  
" Макрос2 Макрос  
' Макрос записаний 15.11.2012 (ps)  
' Виконав студент 2 курсу Іванов Андрій  
' факультет МЕТ  
' Вибір діапазону  
Range("B3:D8").Select  
' Установка параметрів шрифту  
With Selection.Font  
    .Name = "Arial"  
    .FontStyle = "напівжирний"  
    .Size = 12  
    .Strikethrough = False  
    .Superscript = False  
    .Subscript = False  
    .OutlineFont = False  
    .Shadow = False  
    .Underline = xlUnderlineStyleNone  
    .ColorIndex = xlAutomatic  
End With  
End Sub
```

9. За допомогою редактора VBA напишіть процедуру, яка виводитиме на екран вікно вітання "Hello, World!". Збережіть її під ім'ям Hello.

Пишемо текст процедури

Перш ніж починати запис тексту процедури в модуль, будь то новий або існуючий раніше модуль, ви маєте помістити курсор у вікно Code в те місце, куди хочете вставити текст процедури.

Ви можете вводити текст процедури в будь-яке місце в модулі, аби це було після слова End Sub, яке закінчує попередню процедуру, і перед словом Sub, яке починає наступну процедуру. Більшість програмістів поміщають нову процедуру в кінець модуля.

Коли ви пишете процедуру, ви маєте перш за все вказати її ім'я і помістити в текст слова Sub на початку процедури і End Sub в кінці. Якщо хоч одну з цих вимог не буде дотримано,

синтаксичні правила будуть порушені й редактор Visual Basic видасть повідомлення про помилку при спробі запустити процедуру.

У будь-якій мові програмування класичним прикладом простої програми є програма, що виводить на екран слова **Hello, World!** У лістингу ви бачите таку програму, що складається з єдиної процедури.

```
Sub hello()  
MsgBox "Hello, World!","", "Вікно виведення повідомлень на екран"  
End Sub
```

Для того щоб ввести таку *програму* самостійно, зробіть наступне.

- Переконайтеся, що курсор знаходиться на початку першого рядка у вікні Code і введіть текст із лістингу. У кінці кожного рядка натискайте <Enter>, починаючи новий рядок.
- Вводьте текст точно так, як і він приведений у лістингу.
- Перейменуйте новий модуль, давши йому змістовне ім'я.

Visual Basic багато в чому допомагає вам писати програму. Наприклад, після того як ви введете рядок, що містить слово Sub та ім'я процедури, він автоматично введе рядок зі словами End Sub. Таким чином, зменшується вірогідність того, що ви забудете ввести щось важливе.

Крім того, редактор Visual Basic має властивість, яка називається Auto Quick Info. Як тільки ви введете MsgBox і натиснете пропуск, з'явиться допоміжне вікно із списком аргументів функції, ім'я якої ви тільки що ввели, у цьому випадку функції MsgBox. Аргумент, значення якого ви маєте зараз вводити, виділяється напівжирним шрифтом. Допоміжне вікно Quick Info закривається, як тільки ви перемістите курсор на інший рядок. Його можна закрити також, натиснувши клавішу <Esc>.

Перший рядок у лістингу — це оголошення процедури. Оголошення процедури служить для того, щоб повідомити Visual Basic про існування процедури і про місце початку її первинного тексту. Кожне оголошення процедури складається зі слова Sub, пропуску й імені процедури. У лістингу ім'я процедури — Hello. Закінчується оголошення процедури парою порожніх дужок. Ці дужки є обов'язковим елементом. Якщо ви не поставите дужок, Visual Basic зробить це за вас, коли ви натиснете <Enter> в кінці рядка.

Другий рядок у лістингу складає "тіло" процедури і є єдиною інструкцією, яка щось робить. Тіло процедури може складатися з декількох інструкцій. Інструкція MsgBox виводить на екран вікно з повідомленням користувача "Hello, World!". Вікно має заголовок "Вікно виведення повідомлень на екран".

Третій і останній рядок у нашій процедурі — інструкція End Sub. Вона служить для того, щоб вказати VBA кінець процедури. VBA, зустрівши таку інструкцію, закінчує виконання. Як і оголошення процедури, інструкція End Sub має стояти на початку рядка і бути єдиною інструкцією в рядку, але ви можете ставити коментарі після неї. Як вже згадувалося, Visual Basic автоматично додає цю інструкцію в кінці процедури.

10. Запустіть процедуру на виконання.

1. Відкрийте вікно Macro, вибравши в меню Tools-Macros.
2. У списку Macro Name виберіть процедуру Hello.
3. Клацніть на кнопці Run.

У результаті виконання процедури Visual Basic виведе вікно вітання. Клацніть на кнопці OK і закрийте вікно.

11. Закрийте редактор, збережіть книгу Excel на диску D/курс 2/Ваша група/FIO.xls, де FIO – початкові букви вашого прізвища, імені, по батькові. Вийдіть з Excel.

12. Запишіть макрос NewFIO (де FIO – початкові букви вашого прізвища, імені, по батькові), який відкриватиме нову книгу і потім зберігатиме її під ім'ям NEWFIO у вашому каталозі та закривати її. Запустіть його на виконання 2 рази. Результати поясніть та запишіть в зошиті.

13. Напишіть процедуру, яка виводитиме повідомлення «Привіт від Excel!». Вікно повідомлення повинне мати заголовок «Повідомлення Excel».

Варіанти завдань

У поточній робочій книзі створити макрос, який:

1. Створює нову робочу книгу з ім'ям, вказаним користувачем.
2. Додає в поточну робочу книгу лист з ім'ям, вказаним користувачем.
3. Встановлює в активному осередку шрифт розміром у 14 пунктів.
4. Встановлює в активному осередку напівжирне зображення шрифту.
5. Встановлює в активному осередку курсивне зображення шрифту.
6. Встановлює в активному осередку шрифт Arial.
7. Записує в активний осередок Ваше ім'я.
8. Встановлює навколо активного осередку широку рамку.
9. Записує в активний осередок формулу $=A1 + A2$.
10. Відкриває вказану користувачем робочу книгу.
11. Робить активним вказаний користувачем робочий лист.
12. Робить активним вказаний користувачем осередок.
13. Вирівнює вміст активного осередку по центру.
14. Встановлює в активному осередку червоний колір шрифту.
15. Встановлює в активному осередку зелений колір заливки.
16. Встановлює вказане користувачем ім'я для виділеного робочого аркуша.
17. Об'єднує два суміжні осередки.
18. Копіює вміст вказаного користувачем осередку в активний осередок.
19. Привласнює активному осередку вказане користувачем ім'я.
20. Встановлює в активному осередку довільний грошовий формат.
21. Записує в активний осередок символ авторського права ©.
22. Записує в активний осередок формулу обчислення площі круга, якщо радіус зберігається в осередку A1.
23. Записує в активний осередок формулу обчислення площі прямокутника, якщо його висота зберігається в осередку A1, а ширина – в B1.
24. Записує в активний осередок символ Товарний знак TM.
25. Вставляє порожній рядок перед активним осередком.
26. Вставляє порожній стовпчик перед активним осередком.
27. Видаляє рядок у позиції активного осередку.
28. Видаляє стовпчик у позиції активного осередку.
29. Встановлює в активному осередку Фінансовий формат.
30. Записує в активний осередок формулу обчислення площі прямокутного трикутника, висота і ширина якого зберігаються в осередках A1 і B1, відповідно.

Контрольні питання

1. Що таке макрос? Для чого він використовується?
2. Де Excel зберігає записані макроси?
3. Як у програмі Excel виконується записаний раніше макрос?
4. Як ви запускаєте програму запису макросів?
5. Де ви збережете макрос, якщо ви хочете, щоб він був доступний для всіх відкритих книг Excel в будь-якому сеансі роботи?
6. Які макроси ви побачите в списку діалогового вікна Макрос після виконання роботи?
7. Назвіть три основні вікна VB-редактора. Опишіть призначення кожного з них.
8. Що таке *модуль VBA*?
9. Що таке коментарі? Як їх вставити в текст макросу?
10. Що таке *ключове слово VBA*?
11. Які необхідні елементи містить процедура?
12. Що таке *тіло процедури* і де воно розташоване?
13. Чому текст записаних макросів має відступи? Навіщо робити відступи в текстах програм?
14. Для чого призначена процедура MsgBox?

1.5.2 Лабораторна робота № 2 «Лінійний обчислювальний процес»

Вивчимо: типи даних, змінні, константи, оператори і вирази, стандартні функції VBA, обробка помилок, відладка програми, вікно відладки.

Познайомимося з типами даних

Тип даних – це термін, що описує природу даних, які VBA розрізняє і уміє з ними поводитися. У таблиці перераховані типи даних VBA і даний короткий опис типів і діапазон значень для кожного типу.

Таблиця. Типи даних мови Visual Basic

Ім'я типу	Розмір в байтах	Опис і діапазон значень даних
Byte	1 (8 битий)	Позитивні числа від 0 до 255
Boolean	2 (16 бит)	Логічні значення. Тільки значення True і False
Currency	8 (64 бит)	Від -9223372036854775808 до 9223372036854775807
Date	8 (64 бит)	Зберігає комбінацію дати і часу. Дата від 1 січня 100р. до 31 грудня 9999 р. Час від 00:00:00 до 23:59:59
Double	8 (64 бит)	Негативні числа від -1.79×10^{308} по -4.9×10^{-324} . Позитивні числа від 4.9×10^{-324} до 1.79×10^{308}
Integer	2 (16 бит)	Числа від -32768 до 32767
Long	4 (32 бит)	Числа від -2147483648 до 2147483647
Object	4 (32 бит)	Використовується для доступу до будь-якого об'єкта VBA. Зберігає адресу об'єкту
Single	4 (32 бит)	Негативні числа від -3.4×10^{38} до -1.4×10^{45} Позитивних чисел від 1.4×10^{45} до 3.4×10^{38}
String	1 байт на символ	Служить для зберігання тексту. Може містити до 2 мільйонів символів. Рядки в програмі розміщуються у подвійні лапки.
Variant	16 байт + 1 байт	Служить для зберігання даних будь-якого типу

Ви можете перетворювати більшість типів даних з одного вигляду в інший, крім того, Visual Basic автоматично перетворить дані.

Дані типу Variant

Variant — це спеціальний тип даних, в якому можна зберігати всі типи даних, перераховані в таблиці, включаючи типи Object і Array. VBA використовує тип даних Variant завжди, коли ви не вказуєте тип явно.

Дані типу Variant мають властивості тих даних, які фактично в них записані. Наприклад, якщо в даних типу Variant записаний текстовий рядок, вони мають властивості даних типу String, а якщо там записано число, то вони мають властивості числового типу, частіше за все тип Double, хоча це може бути і Integer, і Long, і Single, і Currency.

Дані типу Variant використовують найкомпактніше з можливих уявлень, наприклад, якщо в даних типу Variant записано ціле число, тоді ці дані трактуватимуться як Integer або як Long, залежно від розміру цього числа. Число 15 у змінній типу Variant трактуватиметься як Integer, а число 1000000 — як Long.

VBA зберігає числа з плаваючою комою в змінних типу Variant як Double і поводить з ними так, що ви не зобов'язані завжди піклуватися про точний тип змінних у своїх програмах. Будь-яка змінна, для якої ви не вказали тип явно, стає змінною типу Variant.

Хоча використання змінних типу Variant зручне і звільняє вас від багатьох турбот при написанні процедур, такі змінні вимагають більше пам'яті й більшість операцій виконується на них повільніше, ніж з іншими типами. Взагалі кажучи, вам слід уникати невинновданого використання змінних типу Variant, оскільки це може зробити ваші програми повільними й ускладнити їх читання, крім того, це може привести до появи помилок, що важко виявляються.

Поняття про змінні

Змінна — це ім'я, яке ви, програміст, дали ділянці комп'ютерної пам'яті для того, щоб зберігати в ньому дані деякого типу. Ви можете уявити собі змінну як гніздо, в якому записано якесь значення для подальшого використання. Ім'я змінної — це ярлик, за яким здійснюється доступ до ділянки пам'яті. Вміст ділянки (значення змінної) може мінятися, але ім'я залишається незмінним. У змінній VBA можуть зберігатися дані будь-якого з типів, перерахованих у таблиці.

У деякому розумінні змінна подібна до імені осередку в аркуші Excel. Ви можете дати осередку листа Excel ім'я і потім звертатися до цього осередку за певним ім'ям, не піклуючись про те, які номери стовпчика і рядка має осередок.

Імена змінних мають підкорятися наступним правилам.

- Ім'я змінної починається з букви.
- За буквою може йти будь-яка комбінація букв, цифр і символів підкреслення.
- Ім'я змінної не має містити пропусків, крапок і символів математичних операцій.
- Ім'я змінної не має перевищувати 255 символів у довжину.
- Ім'я змінної не має співпадати ні з одним з ключових слів (званих також зарезервованими словами). Якщо такий збіг матиме місце, VBA видасть повідомлення про синтаксичну помилку.
- Ім'я змінної має бути унікальним у процедурі. Якщо ви дасте двом змінним однакові імена або ім'я змінної співпаде з ім'ям процедури в тому самому модулі, при виконанні процедури VBA видасть повідомлення про помилку.

Приклади правильних імен змінних.				
MyVar,	PayDate,	Percent,	WholePart,	Line12
А це приклади неправильних імен змінних.				
New Iten	Не дозволений символ пропуску			
5thDimention	Починається з цифри			
Dim	Співпадає із зарезервованим словом			
Week/Day	Містить знак арифметичної операції			

Вибираючи імена змінних, прагніть робити їх описовими. Набагато краще дати змінній ім'я Percent, ніж X або Y. Наприклад, якщо у змінній записується температура у градусах за Цельсієм, назвіть змінну CelsiusTemp або DEGREESC.

Створення змінних

Найпростіший спосіб створити змінну — це просто згадати її у виразі. VBA створює змінну і виділяє для неї пам'ять, як тільки виявить її, зазвичай це відбувається при записі в змінну деякого значення.

Запис значення в змінну називається привласненням значення змінної. Ви привласнюєте значення змінної, використовуючи оператора привласнення, який виглядає як знак рівності (=). Ось приклад оператора привласнення.

MyVar=15

Цього оператора записує число 15 в ділянку пам'яті, з якою пов'язано ім'я змінної MyVar. Якщо це перша згадка імені змінної, то VBA створює змінну, виділить для неї пам'ять і записує в неї число 15.

Якщо така змінна була створена раніше, то оператор привласнення просто записує в ділянку пам'яті, на яку указує змінна, число 15, при цьому все, що було записане в цю пам'ять раніше, безповоротно стирається.

Такий спосіб створення змінної простою її згадкою називається неявним оголошенням змінної. Згадуючи змінну у виразі, ви побічно говорите програмі VBA про те, що хочете створити таку змінну. Усі змінні, створені у такий спосіб, отримують тип Variant. Неявне створення змінних називають також створенням змінних "на льоту" (on-the-fly declaration).

З деяких причин VBA надає вам спосіб оголошувати змінні явно, що позбавить вас від можливих помилок. Явне оголошення змінних має багато переваг:

- при явному оголошенні змінних ваша програма працює помітно швидше. VBA створює всі змінні перед тим, як програма почне працювати, і на це не йтиме час при роботі програми;
- явне оголошення змінних допомагає уникнути багатьох помилок, які підстерігають вас при неявному оголошенні;
- явне оголошення змінних полегшує читання програми людині. Бачивши на початку процедури або модуля список всіх змінних, людині легко зрозуміти логіку роботи програми;
- при явному оголошенні змінних вам не потрібно стежити за правильним використанням верхнього і нижнього регістрів, оскільки за вас це робитиме VBA. Замість того щоб підганяти всі входження імені змінної під останню її згадку, VBA виправлятиме регістр імені відповідно до того написання, яке було використано при явному оголошенні.

Для того щоб оголосити змінну, використовується інструкція **Dim** з наступним синтаксисом:

Dim ім'я

Ім'я — це будь-який ідентифікатор. Наприклад:

Dim PcntProfit

Така конструкція створює змінну з ім'ям PcntProfit (ключове слово Dim є скороченням від dimension — розмірність). Усі змінні, які створюються цим способом, мають тип Variant. Імена декількох змінних можуть бути перераховані через кому.

Оголошувати змінну можна тільки один раз. Це може бути зроблено в будь-якому місці програми, але обов'язково до першої згадки змінної в процедурі. Проте хорошою практикою слід вважати винесення всіх оголошень в один блок на початку програми.

У лістингу приведена процедура, перетворена так, що змінна HelloMsg оголошується явно. Процедура Hello виводить повідомлення:

Процедура Hello з явним оголошенням змінних

```
1: Sub Hello ()
2: Dim HelloMsg 'текст повідомлення для функції MsgBox
3: HelloMsg = "Hello, World!"
4: MsgBox HelloMsg, "Вітання"
5: End Sub
```

Вказівка типу змінної в інструкції Dim

Для того щоб оголосити тип змінної в інструкції Dim, додайте за ім'ям змінної ключове слово As і вкажіть тип змінної. Синтаксис цієї конструкції наступний.

Dim ім'я As тип,

де ім'я в цьому випадку — ідентифікатор змінної, а тип — один з типів змінних, приведених у таблиці.

У наступних рядках приведені приклади правильного оголошення типу змінних.

Dim PcntProfit As Single

Dim Gross_Sales As Currency

Dim PayDay As Date

Dim Massage As String

Dim Counter As Integer

Оголошуючи змінну неявно, ви можете вказати її тип, додаючи в кінці імені змінної спеціальний символ. Такі символи називаються символами визначення типу. У таблиці перераховані символи визначення типу і відповідні їм типи.

Символи визначення типу

!	Single
@	Currency
#	Double
\$	String
%	Integer
&	Long

Зверніть увагу на те, що таких символів всього шість. Не існує символів визначення типу для таких типів, як Byte, Boolean, Date, Object або Array. Символи визначення типу можуть знаходитися тільки в кінці імені змінної.

Поняття про константи

Константа – це така величина у програмі, яка ніколи не міняється. Константи можуть бути пойменованими і непойменованими. У загальному випадку слід використовувати пойменовану константу завжди, коли в програмі зустрічається значення, що повторюється, або значення, яке важко запам'ятати, або якщо саме значення на момент написання програми невідоме.

Ім'я константи підкоряється тим же правилам, що й ім'я змінної. Правила створення імен змінних ми розглянули вище в цьому розділі.

Як і у випадку із змінною, константа має бути оголошена до того, як ви перший раз до неї звернетеся. Але на відміну від змінної, константа завжди оголошується явно, і служить для цього ключове слово Const. Ось синтаксис оголошення пойменованої константи.

Const ім'я = значення

Значення може бути будь-якого типу — числом, рядком або датою. Приведемо декілька прикладів правильного оголошення констант.

Const	BillingPoint = 212
Const	Ціна = 8.25
Const	Вітання = "Привіт, колеги!"

За бажання можна оголосити декілька констант в одному рядку, розділяючи їх комами. Наступний рядок повністю еквівалентний попереднім трьом.

Const BillingPoint = 212, Ціна = 8.25, Вітання = "Привіт, колеги!"

Ось приклад правильного оголошення констант.

Const BillingPoint = 212

Const DangerZone = BillingPoint+50

Оголошувати змінну можна тільки один раз. Це може бути зроблено в будь-якому місці програми, але обов'язково до першої згадки змінної в процедурі. Проте хорошою практикою слід вважати винесення всіх оголошень в один блок на початку програми.

Інструкція Dim з'являється в другому рядку лістингу програми Hello. Коли VBA виконує цей рядок, він створює нову змінну і виділяє для неї пам'ять (змінна HelloMsg має тип Variant, оскільки спеціально її тип не вказаний). У цьому ж рядку ви бачите коментарі, що пояснюють призначення цієї змінної.

У рядку 3 лістинги привласнюється значення змінної HelloMsg. У цій змінній міститься текст "Hello, World!". У наступному рядку лістингу змінна HelloMsg передається процедурі MsgBox, яка виводить на екран вікно повідомлення.

Коли ви створюєте пойменовану константу, VBA приписує їй тип відповідно до типу виразу, який їй привласнюється. Наприклад, вважається, що константа, що містить рядок, має тип String.

Але іноді ви можете вказати тип константи явно. Це може знадобитися для підвищення точності обчислень: константа, тип якої вказаний як Double, використовує ширший діапазон значень. Ви можете вказати тип константи Integer, Long, Currency або інший для того, щоб результат обчислень мав, у свою чергу, потрібний тип.

Константа може мати будь-який з типів, перерахованих для змінних, окрім Object і Array. Тип константи оголошується точно так, як і тип змінної, просто оголошення починається із слова Const. Синтаксис оголошення типу константи виглядає таким чином.

Const ім'я As Тип = значення

Тут ім'я — це ідентифікатор константи, тип — один з можливих VBA-типів, значення — це те значення, яке ви привласнюєте константі. Ось приклад правильного оголошення константи.

Const Pi As Double = 3.14

Тут оголошена константа Pi, що має тип Double, і їй привласнено значення 3.14.

У VBA є декілька констант, значення яких заздалегідь визначене для вас. Такі константи називають ще внутрішніми. Окрім внутрішніх констант VBA, існує декілька

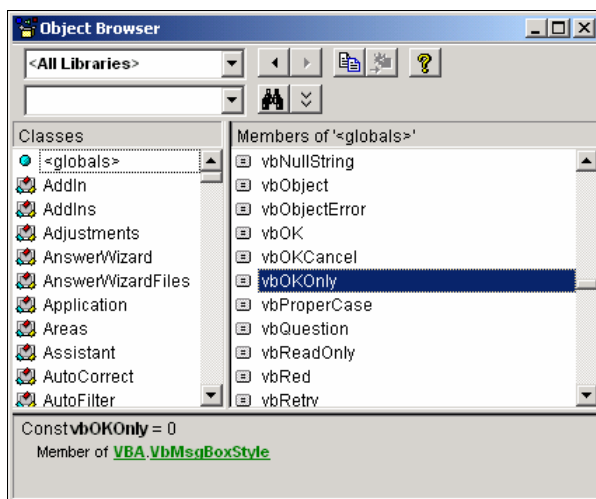
констант, визначуваних додатком, з якого викликається VBA. Зокрема, Excel 2000 визначає декілька констант для роботи з книгами Excel, MS Word визначає константи для роботи з документами і шаблонами, MS Access має декілька констант для роботи з базами даних.

Усі внутрішні константи, визначені у VBA, мають ім'я, що починається з букв vb. Наприклад, такі константи, як vbOKOnly, vbOKCancel, vbAbortRetryIgnore визначені в мові Visual Basic. У додатку Excel визначені, наприклад, такі константи: xlChart, xlCountrySetting, xlWorksheet. Як бачите, їх імена починаються з букв xl.

Внутрішні константи служать для спрощення роботи з деякими вбудованими процедурами VBA, такими як вже знайома вам MsgBox або InputBox, з якою ви познайомитеся пізніше. Процедура MsgBox має обов'язковий аргумент, який вказує на кількість кнопок у вікні. Аргументи, відповідні кнопкам, зазвичай задаються за допомогою внутрішніх констант VBA.

Для того щоб проглянути повний список зумовлених констант як VBA, так і головного застосування, необхідно скористатися вікном **Object Browser (Проглядання об'єктів)**, яке активізується:

- ☐ по команді **View ⇨ Object Browser**
- ☐ по натисненню кнопки  **Object Browser** на панелі інструментів **Standard** або
- ☐ по натисненню клавіші **F2**.



Введення даних користувача у вашу процедуру

Ви вже навчилися користуватися процедурою MsgBox для виведення на екран повідомлень користувачеві. Крім того, ви познайомилися зі змінними і константами і тепер готові до того, щоб навчитися отримувати інформацію від користувача.

Отримання даних, збереження їх у змінній, обробка цих даних або іншої інформації на основі цих даних — ось головні складові уміння писати інтерактивні програми (інтерактивною називається програма, яка інтенсивно спілкується з користувачем, виводячи на екран інформацію й отримуючи інформацію від користувача). Такі програми часто виявляються ефективнішими, ніж просто записані макроси, оскільки користувач може управляти їх роботою залежно від умов, що змінилися.

Дані, отримані від користувача, називаються введеними даними, або просто введенням. Для отримання даних від користувача процедури застосовується функція InputBox (функція — це особливий вид процедури, яка повертає деяке значення). Функція InputBox виводить на екран вікно з текстом, що пропонує ввести деяке значення, і полем введення для цього значення. Крім того, у вікні функції InputBox присутні кнопки OK і Cancel.

Перш ніж викликати функцію InputBox, ви має сформулювати текст запрошення. Крім того, ви можете задати заголовок вікна, яке виведе на екран InputBox.

Синтаксис виклику InputBox виглядає таким чином.

Рядок = InputBox (Запрошення [, Заголовок]),

де рядок — це будь-яка змінна, в якій може зберігатися текстовий рядок; може мати тип String або Variant. Запрошення — пойменована або непойменована константа або змінна, що містить текст. Це обов'язковий аргумент, ви маєте указувати його у виклику функції InputBox.

Заголовок — це необов'язковий другий аргумент функції InputBox (усі необов'язкові елементи в описі синтаксису беруться у квадратні дужки). Заголовком також може бути пойменована або непойменована константа або змінна, що містить текст. Якщо ви опустите цей аргумент у виклику функції, у заголовку буде виведений рядок Microsoft Excel. У лістингу, приведеному нижче, ви бачите модуль, що містить єдину процедуру, і константу, і змінну, оголошені на рівні модуля. Ця процедура обчислює площу круга. Ця процедура питає у користувача, для якого радіуса слід обчислювати площу.

Отримання введення користувача за допомогою функції InputBox

```
Dim CircleArea As Single 'сюди запишемо обчислену площу
```

```
Const Pi As Single = 3.14 'приблизне значення числа Pi..
```

```
Const BoxTitle = "Площа круга"
```

```
Sub Calc_CircleArea()
```

```
    Dim Radius As Single, Temp As String
```

```
    6: Temp = InputBox("Введіть радіус круга", "BoxTitle")
```

```
    7: Radius = CSng(Temp)
```

```
    8: CircleArea = Pi * (Radius * Radius)
```

```
    9: MsgBox CircleArea, "BoxTitle"
```

```
End Sub
```

Процедура називається Culc_CircleArea.

У процедурі оголошуються *константи*: Pi і BoxTitle, яка використовується для виведення заголовків у вікна, що створюються цією процедурою, і *змінні* CircleArea, Radius, Temp.

Змінна CircleArea і константи Pi і BoxTitle оголошуються за межами процедури, тобто будуть доступними всім процедурам цього модуля.

Зверніть увагу на рядок 6. У цьому рядку викликається функція InputBox, яка виводить свій перший аргумент як запрошення, пояснюючи користувачеві, чого від нього чекають. У цьому випадку виводиться рядок “Введіть радіус круга”. Другий аргумент функції — це заголовок вікна; в даному випадку для заголовка вікна використовується змінна BoxTitle.

Після виклику функції, повертане нею значення привласнюється змінною Temp; результат функції InputBox — це завжди рядок, тому змінна Temp була оголошена з типом String. Якщо користувач введе цифру і клацне ОК, у змінної Temp виявиться рядок, введений у вікні повідомлення. Перед тим як використовувати результат у математичних обчисленнях, потрібно перетворити його в число. Це робиться в наступному рядку.

У рядку 8 обчислюється площа круга, а в рядку 9 – виводиться на екран обчислене значення.

1. Запустіть Excel і відкрийте свій файл.

2. Увійдіть у редактор VBA і відредагуйте записану Вами процедуру Hello, ввівши явний опис змінних.

3. Запустіть на виконання процедуру Hello. Чи змінився результат виконання процедури?

4. Напишіть програму, яка запрошуватиме значення змінної з екрану, щоб прочитувати це значення і виводити його знову на екран разом з ім'ям змінної.

Поняття про операторів і вирази

Вираз – це величина або група величин, об'єднаних в єдину суть. Кожен вираз має певне і єдине значення. До складу виразу можуть входити: константи, змінні будь-якого типу, оператори, масиви, елементи масивів, функції. Вираз може мати одне з особливих значень — Empty (порожньо) і Null. Для позначення цих особливих значень у мові VBA існують відповідні зарезервовані слова. Значення Empty відповідає неініціалізованій змінній типу Variant або виразу, в який входить така змінна. Значення Null відповідає виразу, що містить невірні дані, тобто невизначеному виразу. Нижче наведено декілька прикладів виразів.

Вираз	Пояснення
5	Значенням є число 5
"5"	Значення — рядок, що складається з єдиного символу 5
"сего"&"дня"	Значення — рядок сьогодні
C*(p/100)	Значення — число, отримане після множення значення змінної Z на значення, отримане після ділення змінної P на константу 100
CStr(UOO)	Значення рівне результату, повертаному функцією CStr. У цьому випадку це рядок 1200 (функція CStr перетворить числа в рядки)
MyValue <= 7	Логічний вираз. Його значення залежить від того, чи дійсне значення змінної MyValue менше або рівно числу 7

Для об'єднання, порівняння або іншого маніпулювання значеннями, що входять у вираз, використовуються оператори. Величини, що входять у вираз (це можуть бути змінні або константи), називаються операндами; більшість операторів вимагають наявності двох операндів. Наприклад, у виразі 2+1 число 2 і 1 — операнди, об'єднані оператором складання. У виразі може бути будь-яка кількість операторів.

Для об'єднання, порівняння або іншого маніпулювання значеннями, що входять у вираз, використовуються оператори. Величини, що входять у вираз (це можуть бути змінні або константи), називаються операндами; більшість операторів вимагають наявності двох операндів. Наприклад, у виразі 2+1 число 2 і 1 — операнди, об'єднані оператором складання. У виразі може бути будь-яка кількість операторів.

- Дата. Це будь-який вираз, який має тип Date. Такі вирази можуть містити константи дати, змінні числового типу або типу Date, числові константи і дати, які повертаються функціями.

- Числовий вираз. Це вираз один з наступних типів: Byte, Integer, Long, Single, Double або Currency. У числовий вираз можуть входити числові змінні та константи, функції, що повертають числа, і знаки арифметичних операцій. Крім того, ці вирази можуть включати рядки, якщо їх можна перетворити в число (рядок можна перетворити в число, якщо в нього входять тільки цифри; наприклад, рядок "36" можна перетворити в число 36.)

- Строкові вирази. Це вирази мають тип String. Такий вираз може включати строкові константи і змінні, функції, що повертають рядки, і операторів конкатенації рядків. У строкові вирази можуть також входити числові вирази, якщо їх можна перетворити в рядки.

- Логічні вирази. Це вираз, який має одне з булевих значень, тобто True або False. Такі вирази можуть складатися з логічних змінних і констант, функцій, які повертають логічне значення, операторів порівняння і логічних операторів.

- Об'єктний вираз. Це вираз, значенням якого є посилання на об'єкт.

Не всі типи даних сумісні один з одним. Не можна комбінувати несумісні типи в одному виразі. Наприклад, не можна скласти текстовий рядок "Привіт!" з числом — така операція не має сенсу, і її значення не визначене.

У той же час деякі типи є сумісними. Ви можете вільно комбінувати різні числові типи між собою, VBA автоматично виконає необхідні перетворення. Іноді VBA може автоматично перетворювати й інші типи даних, досягаючи їх сумісності, але не завжди.

Дуже важливо знати типи вхідних у вираз значень, оскільки якщо у виразі зустрінуться несумісні дані, то при виконанні програми VBA видасть вам повідомлення про невідповідність типів. Коли VBA зустрічає у виразі різні типи даних, він спочатку намагається залагодити конфлікт, автоматично перетворюючи дані до сумісних типів. Але якщо це не вдається, ви отримуєте повідомлення про невідповідність типів. Те саме відбудеться, якщо ви спробуєте привласнити змінній значення виразу, тип якого несумісний з типом змінної. Повідомлення про невідповідність типів не видається, якщо один з операндів має тип Variant. Як ми вже говорили, найпростіше перетворення типів виконується тоді, коли змінна має тип Variant, але не слід цим зловживати.

Оператор привласнення (=) служить для того, щоб записати значення виразу змінну або значення однієї змінної в іншу. Оператора привласнення має дві синтаксичні форми, які

обидві законні й абсолютно еквівалентні. У першому випадку використовується ключове слово **Let** і синтаксис має наступний вигляд.

Let ім'я = вираз,

де ім'я — ідентифікатор змінної, а вираз — будь-який допустимий у VBA вираз. Така форма оператора привласнення прийшла зі старої мови Basic. Ось приклад використання оператора привласнення.

Let X = Y 'Значення Y записується в змінну X

Друга форма більш загальноприйнята і використовується повсюдно. Синтаксис її такий:

ім'я = вираз,

де ім'я — ідентифікатор змінної, а вираз — будь-який допустимий у VBA вираз. Ось приклад використання оператора привласнення в цій простій і загальноприйнятій формі.

X = Y

MyVar = 12 - YourVar

В обох формах оператора привласнення змінна ліворуч від нього набуває значення виразу праворуч. При виконанні оператора привласнення спочатку обчислюється вираз, що стоїть праворуч від оператора, а потім набутого значення привласнюється змінною, такою, що стоїть ліворуч.

У багатьох процедурах ви можете бачити таку конструкцію.

Count = Count + 1

GrossTotal = GrossTotal + SubTotal

Це часто застосовується для накопичення значення якого-небудь лічильника. Сенса написаного не відразу очевидний; на перший погляд навіть може здатися, що це противоречить логіці. Але ви зрозумієте написане, якщо пригадаєте, що при виконанні оператора привласнення спочатку обчислюється вираз, що стоїть праворуч від оператора, а потім набутого значення привласнюється змінною, такою, що стоїть зліва.

Арифметичні оператори

VBA може виконувати всі стандартні арифметичні дії: додавання, віднімання, множення і ділення. Крім того, є оператор для піднесення числа до ступеня і додаткових операторів для цілочисельного ділення і для ділення по модулю. У таблиці зібрані арифметичні оператори VBA.

Оператор	Синтаксис	Пояснення
+	N1 + N2	Складає N1 і N2
-	N1 - N2	Віднімає N2 з N1
- N1	Унарний мінус (від'ємне число)	
*	N1 * N2	Помножує N1 на N2
/	N1 / N2	Ділить N1 на N2
\N1 \N2	Цілочисельне ділення. Ділить N1 на N2, відкидаючи дробову частину	
ModN1 Mod N2	Ділення по модулю. Ділить N1 на N2, повертаючи залишок від ділення	
^	N1 ^ N2	Зводить N1 в ступінь N2

5. Запишіть програму, що виконує обчислення площі круга з введенням радіуса з екрана і виведенням результату на екран.

6. Запустіть програму і виконайте її в покроковому режимі.

Оператори порівняння

Оператори порівняння іноді називають операторами відношення. Зазвичай їх застосовують для задання критерію на ухвалення рішення або для формування умови виконання деякої послідовності операцій. Результат оператора порівняння має тип Boolean і приймає тільки два значення — True і False. Порівнювати можна поіменовані та неіменовані константи і змінні схожих типів.

Таблиця. Оператори порівняння

Оператор	Синтаксис	Опис
=	E1 = E2	Рівно. True, якщо E1 рівне E2, False інакше
<	E1 < E2	Менше. True, якщо E1 менше E2, False інакше
<=	E1 <= E2	True, якщо E1 менше або рівне E2, False інакше
>	E1 > E2	True, якщо E1 більше E2, False інакше
>=	E1 >= E2	True, якщо E1 більше або рівне E2, False інакше
<>	E1 <> E2	Не рівно. True, якщо E1 не рівне E2, False інакше
Is	E1 Is E2	Є. Обидва операнди повинні мати тип Object. True, якщо E1 і E2 вказують на один і той же об'єкт, False інакше
Like	E1 Like E2	Подібно. Обидва операнди повинні мати тип String. True, якщо E1 відповідає зразку, що міститься в E2, False інакше

Логічні оператори

Зазвичай логічні оператори застосовуються для того, щоб, комбінуючи умовні вирази, створювати складні умови для ухвалення рішень у процедурі, або для формування умов виконання деякої групи інструкцій.

Як операнд у логічному операторі може використовуватися будь-яке значення типу Boolean або число, яке можна перетворити до цього типу. VBA вважає, що 0 відповідає значенню False, а будь-яке інше число — значенню True.

Таблиця. Логічні оператори

Оператор	Синтаксис	Пояснення
And	E1 And E2	Кон'юнкція. Вираз істинний, якщо істинні обидва його операнда
Or	E1 Or E2	Диз'юнкція. Вираз істинний, якщо істинний хоч би одного його операнд
Not	Not E1	Заперечення. Вираз істинний, якщо E1 помилково, і навпаки

Таблиця істинності оператора And.

Операнд1	Операнд2	Значення виразу
True	True	True
True	False	False
False	True	False
False	False	False

Оператор And використовується для того, щоб перевірити, чи є обидві умови істиною одночасно. Наприклад, розглянемо оператора And у наступному виразі.

(Gross_Sales < 50000) And (Net_Profit < 10000)

Цей вираз має значення True, якщо значення змінної Gross_Sales менше 50000, а значення змінної Net_Profit менше 10000.

Таблиця істинності оператора Or.

Операнд1	Операнд2	Значення виразу
True	True	True
True	False	True
False	True	True
False	False	False

Оператор Or використовується для того, щоб перевірити, чи є хоча б одна з умов істиною. Наприклад, розглянемо оператора Or у наступному виразі.

(Gross_Sales < 50000) Or (Net_Profit < 10000)

Цей вираз має значення True, якщо значення змінної Gross_Sales менше 50000 або значення змінної Net_Profit менше 10000.

Оператор Not називається логічним *запереченням*. Він інвертує будь-яке значення, до якого застосовується. Його значенням буде True, якщо значення операнда — False, і навпаки.

Синтаксис оператора Not має наступний вигляд.

Not Операнд1

Операнд1 — це будь-який логічний вираз мови VBA.

Основний оператор конкатенації: "&".

Знак амперсанда має єдине застосування в мові VBA — для злиття рядків. Ніяких інших функцій цей знак не виконує. Синтаксис оператора конкатенації має наступний вигляд.

Операнд1 & Операнд2

Операнд1 і Операнд2 — це будь-які вирази строкового або числового типу, VBA String. Якщо один з операндів має значення Null або Empty, VBA трактує його як рядок нульової довжини, тобто рядок, що не містить символів.

Пріоритет операцій

Пріоритет операторів указує порядок їх обробки в складному виразі. Якщо у виразі використовується більш за одного оператора, то всі вони виконуються в строго певному порядку, що забезпечує однозначне трактування значення цього виразу.

У таблиці, приведеній далі, представлені всі оператори VBA в порядку зменшення їх пріоритетів.

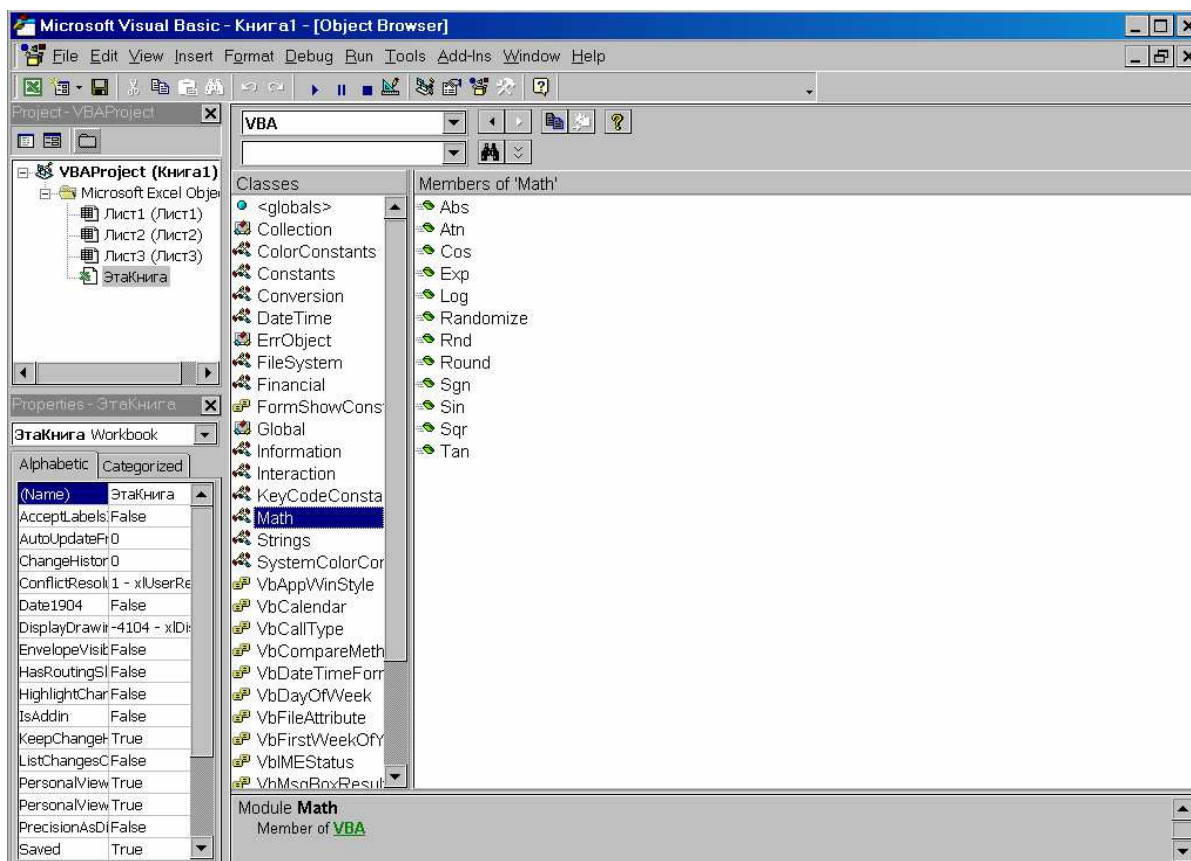
Пріоритет	Операція	Опис
1	()	Круглі дужки
2	^	Піднесення до ступеня
3	-	Унарний мінус
4	*, /	Множення і ділення
5	\	Цілочисельне ділення
6	Mod	Ділення по модулю
7	+, -	Додавання і віднімання
8	&	Конкатенація
9	>, <, >=, <=, <>, Is, Like	Оператори порівняння
10	Not	НЕ
11	And	І
12	Or	АБО
13	Xor	Що виключає АБО
14	Eqv	Еквівалентність
15	Imp	Імплікація
16	=	Привласнення

7. Запишіть програму згідно з вашим варіантом. Запустіть програму на виконання.

Відладьте програму. Відкрийте вікно спостереження змінних і виконайте програму в покроковому режимі. Складіть у зошиті таблицю змін значень змінних. Покажіть результат викладачеві.

Для того щоб проглянути повний список функцій як VBA, так і головного застосування, необхідно скористатися вікном **Object Browser (Проглядання об'єктів)**, яке активізується описаним вище способом. У вікні **Classes** виберіть Math, потім у вікні **Members** виберіть потрібну вам функцію, прогляньте внизу синтаксис цієї функції. Функція **Abs** обчислює модуль аргументу, **Exp** – зводить *e* у ступінь, тригонометричні функції легко впізнаються.

Відкрити вікно спостереження значень змінних у редакторі VBA можна так: виділити змінну, значення якої ви хочете спостерігати, потім вибрати **Debug – Quick Watch**. Щоб додати для спостереження ще одну змінну, потрібно виділити її та увійти **Debug – Add**.



8.

Індивідуальні завдання

№	Завдання	№	Завдання
1	$Y = a \cdot x + b;$	2	$Y = \cos(a) \cdot x + b;$
3	$Y = \sin(a) \cdot x + b / c;$	4	$Y = \operatorname{tg}(a) \cdot x + b;$
5	$Y = a / x + b;$	6	$Y = (a + b) / x;$
7	$Y = \operatorname{tg}(a) + b / c;$	8	$Y = a - b / x;$
9	$Y = \cos(a) + x / b;$	10	$Y = c - b / x;$
11	$Y = \cos(a) + b;$	12	$Y = \sin(a) + b;$
13	$Y = \sin(a) + b / c;$	14	$Y = c - b / a ;$
15	$Y = \sin(a + b);$	16	$Y = a \cdot x + b;$
17	$Y = \cos(a + x) + b;$	18	$Y = \sin(a) / x + b / c;$
19	$Y = \operatorname{tg}(a) \cdot x + b;$	20	$Y = a / (x + c);$
21	$Y = (a + b) / x;$	22	$Y = e^a + b / c;$
23	$Y = a - b / x;$	24	$Y = \sin(a + x / b);$
25	$Y = c - b / x ;$	26	$Y = e^a + b;$
27	$Y = \sin(a + b / x);$	28	$Y = e^a + b / c;$
29	$Y = \sin c - b / a ;$	30	$Y = \cos(a + b);$

Контрольні питання

1. Скільки числових типів даних налічується у VBA?
2. Яка різниця між числами типу Integer і Single? Між Integer і Long?
3. Що таке символи визначення типу?
4. Що означає вислів "визначити змінну неявно"? А що таке "явно"?
5. Які правила в мові VBA для імен змінних? Чим ще слід керуватися при виборі імен змінних?
6. Які переваги дає явний опис змінних?
7. Чому в процедурах краще використовувати поймаєні константи?
8. Для чого служить функція InputBox?
9. Які аргументи функції InputBox є обов'язковими?
10. Якого типу значення повертає функція InputBox?
11. Як можна скористатися результатом виразу? Чи повинні ви використовувати результат виразу?
12. Для яких цілей використовується знак рівності як оператор?

Вправи

1. Вирішіть, яка з перерахованих нижче величин має бути змінною, а яка — константою. Виберіть для них імена і напишіть визначення типу (якщо можна, зробіть це і за допомогою ключових слів, і за допомогою символів визначення типу).

- А Обчислена кількість стовпчиків в аркуші Excel
- Б Обсяг продажів для підрозділу компанії
- В Передбачувана кількість респондентів в опитуванні
- Г Обчислена площа поверхні циліндра
- Д Коефіцієнт для переведення дюймів у сантиметри
- Е Рентабельність операції, виражена у відсотках

2. Напишіть уручну процедуру з назвою EchoThis, в якій використовується функція InputBox для введення користувачем деякої фрази, а потім ця фраза виводиться на екран функцією MsgBox.

3. За допомогою програми запису макросів створіть у книзі Personal.xls макрос, який відкриватиме існуючу книгу Excel і вибиратиме в ній аркуш з ім'ям ЛІСТ3. Зупиніть запис макросу. Відредагуйте записаний макрос так, щоб він питав у користувача ім'я файлу книги і потім відкривав книгу з цим ім'ям. (Підказка. Спочатку уважно ознайомтеся з текстом записаного макросу і зрозумійте, в якому рядку відкривається файл книги. Помістіть виклик функції InputBox безпосередньо перед цим рядком. Привласніть введене користувачем значення змінної, а потім підмініть цій змінній непойменовану константу, створену при записі макросу.)

4. Розставте дужки в приведених нижче виразах так, щоб їх результат відповідав вказаному.

$$3*5-7 -6$$

$$4-7 + 26 / 10 \quad 3.07$$

$$312 / 47 + 16 - 2 \quad 5.114754$$

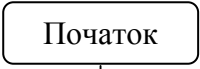

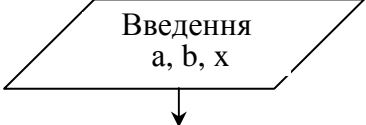
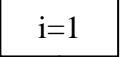
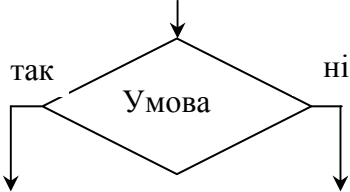
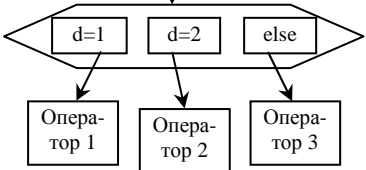
1.5.3 Лабораторна робота № 3

«Обчислювальний процес, що розгалужується»

Алгоритмізація

Гарну програму простіше написати, якщо, заздалегідь проаналізувавши завдання, скласти алгоритм її вирішення, тобто алгоритм програми. Алгоритм можна скласти у вигляді словесного опису послідовності дій або представити у вигляді блок-схеми.

Блок-схема складається з графічних об'єктів різних типів, що містять усередині себе текст, що пояснює, що саме виконується в цьому фрагменті. Графічні об'єкти з'єднуються стрілками, що показують послідовність обробки блоків. Перелік типів блоків і їх опис приведений в наступній таблиці.

№	Тип блоку	Опис блоку
1		Ознака початку алгоритму.
2		Ознака кінця алгоритму.
3		Блок введення/виведення, означає реалізацію введення даних з клавіатури (або виведення даних на екран). У даному випадку це блок введення й еквівалентом його у VBA є оператор InputBox.
4		Блок присвоєння змінним деяких значень або виразів. У цьому прикладі змінній i присвоюється одиниця.
5		Блок умови застосовується коли, залежно від істинності (або помилковості) деякої умови, виконуватися будуть ті (або інші) оператори. Якщо Умова буде істинною, то виконуватися буде ліва гілка, а права виконуватися не буде. Якщо Умова буде помилковою, то виконуватися буде права гілка, а ліва виконуватися не буде.
6	Блок множинного галузження використовується, коли залежно від значення деякого ключа-ідентифікатора, обчислювальний процес розгалужується на більше ніж дві гілки. У VBA для цього використовується оператор «Select Case». У цьому прикладі утворюється три гілки, але їх може бути і більше. Обчислювальний процес йде тільки по одній гілці.	
Приклад		«Else» є необов'язковим параметром і означає випадок, коли ключ-ідентифікатор прийме таке значення, якого немає серед перерахованих значень (для цього прикладу $d \neq 1$ і $d \neq 2$ наприклад 3 або 0 або щось інше).

Типи обчислювальних процесів

1. Лінійний обчислювальний процес

Якщо при запуску програми виконуються всі її оператори, починаючи з першого і кінчаючи останнім, то це називається *лінійним обчислювальним процесом*. Наприклад, обчислення площі круга, виконане в попередній роботі.

Освоєнню лінійного обчислювального процесу присвячена друга лабораторна робота.

2. Обчислювальний процес, що розгалужується

Якщо в результаті виконання програми, процес обчислення може пройти тільки по одному з декількох (у загальному випадку багато) альтернативних шляхів, то такий обчислювальний процес називається таким, що розгалужується. У простому випадку він реалізується оператором IF, який має на увазі дві альтернативи: одну, якщо умова виконується, й іншу, якщо умова не виконується. Наприклад, якщо студент іногородній, то йому надається гуртожиток, якщо немає, то гуртожиток не надається.

Приклад 1.

У деякому магазині надається знижка у розмірі 10 %, якщо сума придбаного товару перевищує 1000 грошових одиниць. Написати процедуру, яка підраховуватиме суму цієї знижки.

Лістинг процедури **Discount1**, що підраховує суму знижки, приведений далі.

```
1: Sub Discount1()  
2:   Dim dCost As Double      ' Вартість покупки  
3:   Dim dDiscount As Double ' Знижка  
4:  
5:   dDiscount = 0#           ' Обнулення знижки  
6:   dCost = InputBox("Введіть вартість покупки")  
7:   If dCost > 1000 Then dDiscount = dCost * 0.1  
8:   MsgBox "Знижка півна: " & dDiscount  
9: End Sub
```

У ній:

Рядки 2 і 3 – оголошення два змінних типу **Double** для зберігання вартості покупки і суми знижки, відповідно.

Рядок 5 – початково ніяка знижка не надається.

Рядок 6 – введення вартості придбаного товару.

Рядок 7 – підрахунок суми знижки, якщо вартість покупки перевищує 1000 грошових одиниць.

Рядок 8 – виведення результату.

Розглянемо інструкцію **If** мови програмування VBA формальніше і детальніше.

Синтаксис команди:

```
If <умова> Then <інструкції1> [Else <інструкції2>]
```

Працює інструкція **If** таким чином:

1. На початку обчислюється значення умови, яка може бути **True** або **False**.
2. Якщо умова має значення **True**, то виконується група інструкції1, причому декілька інструкцій мають поміщатися в одному рядку, а група інструкції2 – пропускається.
3. Якщо ж умова має значення **False**, то, навпаки, виконується група інструкції2, а група інструкції1 – пропускається.
4. Виконання програми у будь-якому випадку продовжується з наступною за If інструкцією.
5. Конструкції, що знаходяться у квадратних дужках, є необов'язковими. Тобто може бути відсутньою конструкція Else і інструкції2, які виконуються, якщо умова помилкова.

У приведеному вище Прикладі 1 використовується тільки одна інструкція

dDiscount = dCost * 0.1, яка виконується, якщо умова істинна, тобто вартість придбаного товару перевищує 1000 грошових одиниць. Якщо ж умова помилкова, то ніякі інструкції не виконуються.

Групи інструкції1 і інструкції2 можуть складатися не з однієї інструкції, як в попередньому прикладі, а з декількох, розділених двокрапкою (:). Наприклад:

```
If A > 10 Then X = X + 1: Y = Y + X: Z = Z + Y
```

Приведену вище процедуру **Discount1** можна написати і по-іншому, наприклад, задіявши гілку **Else** інструкції **If**. Лістинг відкоректованої таким чином процедури, тепер уже названою **Discount2**, приведений далі.

```
10: Sub Discount2()  
11:   Dim dCost As Double      ' Вартість покупки  
12:   Dim dDiscount As Double ' Знижка  
13:  
14:   dCost = InputBox("Введіть вартість покупки")  
15:   If dCost > 1000 Then dDiscount = dCost * 0.1 _  
16:   Else dDiscount = 0#  
17:   MsgBox "Знижка півна: " & dDiscount  
18: End Sub
```

У ній значення знижки рівне 0 привласнюється змінною dDiscount не на початку процедури априорі, а тільки якщо сума покупки не перевищує 1000 грошових одиниць. При

такому записі процедури довжина інструкції **If** стала занадто великою, і її довелося перенести на наступний рядок (для перенесення рядка використовується знак «_»). Якщо ж у кожній гілці інструкції **If** буде задіяно по декілька інструкцій, то вся інструкція **If** стане зовсім нечитабельною. Для цього у VBA є блокова форма інструкції If, яка має такий вигляд:

```
If <умова> Then
<инструкции1>
[Else
[<инструкции2>]]
End If
```

Працює вона точно так, як і приведений раніше її однорядковий варіант.

Тоді, з урахуванням блокового синтаксису інструкції **If**, процедуру визначення знижки можна записати таким чином, назвавши її при цьому вже **Discount3**, відповідно:

```
19: Sub Discount3()
20:     Dim dCost As Double      ' Вартість покупки
21:     Dim dDiscount As Double ' Знижка
22:
23:     dCost = InputBox("Введіть вартість покупки")
24:     If dCost > 1000 Then
25:         dDiscount = dCost * 0.1 _
26:     Else
27:         dDiscount = 0#
28:     End If
29:     MsgBox "Знижка півна: " & dDiscount
30: End Sub
```

Хоча процедура **Discount3** декілька довше за свої попередні варіанти, зате вона більш структурована і візуально сприймається набагато краще. *Однорядковий варіант інструкції If слід використовувати, якщо при виконанні (або невиконанні) деякої умови необхідно виконати одну просту дію (як в Прикладі 1), у решті всіх випадків набагато ефективніше застосовувати її блоковий варіант.*

Мова програмування VBA допускає будь-яку глибину вкладеності інструкцій If одна в одну. Тобто як інструкція в будь-якій його гілці, у свою чергу, може виступати інструкція **If**. Єдине і природне обмеження, що виникає при цьому, полягає в тому, що *вкладена інструкція If має цілком і повністю поміщатися у відповідній гілці зовнішньої інструкції.*

1. Запустіть Excel і відкрийте свій файл.
2. Увійдіть у редактор VBA і запишіть процедуру, що обчислює знижку на покупку залежно від вартості товару за наступною схемою:
 - а. Якщо вартість товару < 1000, знижка = 0 %;
 - б. Якщо вартість товару ≥ 1000 і менше 5000, знижка = 10%;
 - с. Якщо вартість товару ≥ 5000, знижка = 20%.
3. Запустіть на виконання процедуру і виконайте її в покроковому режимі, вводячи значення вартості товару з різних діапазонів. Зверніть увагу, які оператори виконуються залежно від значення, що вводиться.
4. Написати програму нарахування податку на прибуток, використовуючи вкладених операторів If, відповідно до наступної таблиці:

Прибуток [грошова одиниця]	Ставка податку [%]
до 500	10
від 500 до 1000	15
1000 і більш	20

Оскільки на збитки, або негативний прибуток, податок не нараховується, то в програмі необхідно передбачити обробку такої ситуації. У ній так само необхідно передбачити дві змінні типу **Double** для зберігання: суми прибутку і суми податку, що нараховувався.

Вкладені інструкції **If** для перевірки трьох і більше умов візуально стають мало зрозумілими. Тому у VBA була передбачена ще одна її форма:

```
If <умова> Then
[<інструкції1>]
[ElseIf <умован> Then
[<інструкціїn>] ...
[Else
[<інструкції2>]]
End If
```

У ній може бути скільки завгодно гілок **ElseIf** для реалізації альтернативних варіантів. Єдине обмеження при цьому – всі вони мають іти до останньої гілки **Else**.

5. Скласти блок-схему алгоритму відповідно до свого варіанту. Введення координат точки здійснити з екрана; як заголовок вікна введення взяти своє прізвище; визначити, чи потрапляє крапка в задану область, і вивести на екран координати точки і відповідь про місцеположення крапки.

6. Написати і відладати програму процесу, що розгалужується, із використанням оператора *If*.

Розглянемо інструкцію **Select Case** мови програмування VBA, яка використовується в тому випадку, якщо потрібно вибрати більш ніж з двох гілок виконання процедури. Синтаксис команди:

```
Select Case Вираз
Case Логіч.вираз1
Інструкції1
Case Логіч.вираз2
Інструкції2
.
.
.
Case Логіч.виразN
Інструкціїn
[Case Else
ІнструкціїElse]
End Select
```

Працює інструкція **Select Case** таким чином:

1. На початку обчислюється значення Виразу, і потім результат цього обчислення з кожним з виразів, що зберігаються в кожному Логіч.виразі.

2. Якщо значення Виразу задовольняє Логіч.виразу для одного з розділів Case, Інструкції цього розділу виконуються.

3. Якщо Вираз відповідає більш, ніж одному розділу Case, виконуються інструкції тільки першого з них.

4. Після виконання інструкцій вибраного розділу Case виконується наступна за ключовими словами End Select інструкція.

5. Якщо значення Виразу не задовольняє Логіч.виразу ні для одного з розділів Case, виконуються інструкції ІнструкціїElse.

В окремих розділах Case може стояти більш, ніж одна Інструкція, розділені комами. ІнструкціїN мають наступний синтаксис:

```
Інструкція1, Інструкція2, ІнструкціяN.
```

Логічні вирази можуть бути деякими числовими, строковими або логічними виразами, вони можуть бути представлені поряд зі значеннями з використанням оператора To :

```
Вираз1 To Вираз2.
```

Приклад використання оператора Select Case:

```
Sub Temp()  
Dim temp  
Temp=Application.InputBox(_  
    Prompt:="Скільки градусів?",_  
    Title:="Процедура Temp",_  
    Type:=1)  
  
Select Case temp  
    Case Is>40  
        MsgBox "Дуже гаряче!"  
    Case 37 To 40  
        MsgBox "Нормально!"  
    Case 34 To 36  
        MsgBox "Добре!"  
    Case 30 To 33  
        MsgBox "Прохолодно!"  
    Case Else  
        MsgBox "Підігрійте!"  
End Select  
End Sub
```

У мові програмування VBA існують інструкції безумовного переходу, які використовуються в тому випадку, якщо потрібно перейти в яке-небудь місце процедури без перевірки умов; в основному ця інструкція застосовується при обробці помилок. Синтаксис команди:

GoTo Line

де Line – номер рядка в тій же процедурі, що і інструкція GoTo. Номер рядка Line має наступний синтаксис:

name:

Тут name: – це ідентифікатор VBA.

7. Написати і відладити програму процесу, що розгалужується, з використанням оператора *Select Case* відповідно до свого варіанта.

8. Показати результати роботи викладачеві.

Варіанти завдань

1. Чи потрапляє точка A в круг радіусом $R=3$ з центром у точці C(3, 4)?
2. Чи потрапляє точка A в круг радіусом $R=5$ з центром у точці C(6, 8)?
3. Чи потрапляє точка A у прямокутник $2 \leq x \leq 4$ і $2 \leq y \leq 5$?
4. Чи потрапляє точка A в смугу $0 \leq x \leq 5$?
5. Чи потрапляє точка A в смугу $4 \leq x \leq 6$?
6. Чи перевищує відстань від точки A до точки B 5 одиниць. Координати точки Y ввести з клавіатури?
7. Чи лежить точка A на прямій $y = a + 4$?
8. Чи належить точка A графіку функції $y = 4x^2 - 5$?
9. Чи потрапляє точка A у прямокутник $6 \leq x \leq 10$ і $7 \leq y \leq 9$?
10. Чи потрапляє точка A в один з кругів $R1=1$ C1(1, 1); $R2=2$ C2(5, 4)?
11. Чи потрапляє точка A в перший квадрант?
12. Чи потрапляє точка A у другий квадрант?
13. Чи потрапляє точка A у третій квадрант?
14. Чи потрапляє точка A в четвертий квадрант?
15. Чи потрапляє точка A в область додатних значень X?
16. Чи потрапляє точка A в область від'ємних значень X?
17. Чи потрапляє точка A в область непозитивних значень осі y?
18. Чи потрапляє точка A в область не негативних значень осі y?
19. Чи потрапляє точка A в точку перетину графіків функцій $y = x + 1$ і $y = x^2 - 4 \cdot x + 2$?

20. Чи належить точка А графіку функції $y = 4x^2 + 8$?
21. Чи потрапляє точка А в область перетину кругів C1: R1=3, центр у точці C1(0, 0) і C2: R2=5, C2(5, 0)?
22. Чи потрапляє точка А у прямокутник $-1 \leq x \leq 5$ і $-2 \leq y \leq 3$?
23. Чи потрапляє точка А на коло $R = 4$, C(1,1) ?
24. Чи потрапляє точка А на коло $R = 3$, C(4,5) ?
25. Чи перевищує відстань від т. А до осі Х 6 одиниць?
26. Чи перевищує відстань від т. А до осі У 7 одиниць?
27. Чи перевищує відстань від т. А до початку координат 5 одиниць?
28. Чи потрапляє точка А на одне з кіл: R1=1, C1(1,2) і R2=2, C2(2,3) ?
29. Чи перевищує відстань від т. А до прямої $x = 4$ шість одиниць?
30. Чи перевищує відстань від т. А до прямої $y = -5$ десять одиниць?

Контрольні питання

1. Що таке умовний і безумовний перехід?
2. Які ви знаєте інструкції умовного переходу?
3. Які ви знаєте інструкції безумовного переходу?
4. Коли і чому можна використовувати інструкцію Select Case?
5. Скільки пропозицій Case може включати інструкція Select Case?
6. Як визначити гілку інструкції для виконання, якщо в інструкції Select Case немає жодної пропозиції Case? Куди в цьому випадку переходить виконання усередині інструкції Select Case?
7. Що таке складна умова? Як воно будується?
8. Скільки гілок має оператор If Then?
9. Скільки гілок має оператор Select Case?
10. Як перейти на потрібний рядок програми?

1.5.4. Лабораторна робота № 4 «Циклічний обчислювальний процес»

Одним з недоліків записаних макросів є їх нездатність виконатися кілька разів, поки це не буде виконано вручну. У VBA є декілька сильних засобів, що дозволяють легко повторювати операції та управляти їх повторенням.

Програмні структури, що дають можливість повторюватися одній або декільком інструкціям, називаються циклами. Кожний раз, коли VBA завершує один повний цикл виконання всіх інструкцій, що знаходяться усередині циклічної структури, називається ітерацією циклу.

Деякі циклічні структури створені так, щоб вони виконувалися фіксоване число разів. Вони називаються певними циклами. Інші виконуються невизначене число разів, залежно від деякої умови. З огляду на те, що це число повторень невизначене, вони називаються невизначеними циклами.

І в тих, і в інших структурах є вирази, що показують, до яких пір слід повторювати цикл. Вони називаються визначниками циклу. У фіксованій структурі такий визначник майже завжди є числовим виразом. У невизначеній структурі визначник зазвичай є логічним виразом, який описує умову, при якій виконання циклу продовжуються або зупиняється. Логічні вирази для циклічних структур створюються і використовуються точно так, як і в разі умовних інструкцій галудження у VBA.

Прийнято два основні засоби побудови невизначеного циклу. У першому випадку VBA перевірятиме умови визначника до виконання циклу. Якщо умова виконання має значення False, то інструкції усередині циклу просто будуть пропущені. У другому випадку можна створити такий цикл, щоб умова у визначнику перевірялася після виконання цих інструкцій.

Можна також створювати невизначений цикл, у якого у визначнику немає ніякої умови зовсім. Такого роду цикли виконуються вічно і називаються нескінченними. Вони ніколи не закінчуються; більшість таких циклів є результатом помилки програміста.

Повторення фіксоване число разів: цикли For

Це є найпростішим з циклів. У VBA є дві різні структури таких циклів: **For .. Next** і **For Each ... Next**. Обидві ці структури називаються циклами **For**, тому що завжди виконуються певне число разів.

Застосування циклу For.. Next

Цикл For.. Next має такий синтаксис:

For counter = start to end [Step StepSize]

Statements

Next [counter]

Тут **counter** представляє будь-яку числову змінну, відповідну правилам VBA, зазвичай з типом даних Integer або Long. Вираз **start** є будь-яким числовим виразом і визначає початкове значення змінної лічильника counter. Вираз **end** – також числовий вираз і визначає кінцеве значення тієї ж змінної.

За умовчанням VBA кожного разу після завершення інструкцій з циклу збільшує значення змінної **counter** на 1. Інше значення збільшення можна задати за допомогою необов'язкового ключового слова **Step**. Після нього необхідно вказати величину (крок, на який треба збільшити значення **counter**). У нашому синтаксисі **StepSize** представляє будь-який числовий вираз і якраз указує цю саму величину.

Statements представляє одну або декілька інструкцій VBA (або взагалі жодної з них). Вони складають "тіло циклу", і кожна з них виконується всякий раз, коли виконується сам цикл.

Ключове слово **Next** сигналізує VBA, що досягнутий кінець циклу, необов'язкова змінна **counter** після **Next** має бути тою же змінною, яка знаходиться після ключового слова **For** на початку циклічної структури. Її треба писати після **Next**, щоб легко читати початковий текст програми (особливо з вкладеними циклами **For.. Next**).

При виконанні циклу **For.. Next** спочатку змінною counter привласнюється значення, представлене змінною **start**. Потім VBA виконує всі інструкції, представлені в **Statements**, поки не буде досягнуто ключове слово Next. Ключове слово Next сигналізує VBA, що досягнутий кінець тіла циклу. Потім значення змінної counter збільшується на величину змінної **StepSize** (крок), якщо є необов'язковим ключове слово **Step**. Якщо цього слова немає, то значення counter збільшується на 1. Відбувається перехід на початок циклу, а потім поточне значення counter порівнюється із значенням, представленим end. Якщо counter буде менший або рівний end, то цикл виконається знову. Якщо counter більше, ніж end, то виконання продовжується з першої інструкції, яка знаходиться після ключового слова Next.

Лічильник у цьому типі циклів може бути як із зростаючим лічильником, так і з тим, що убыває. Для цього необхідно використовувати ключове слово **Step**, з негативним значенням StepSize, унаслідок чого значення counter зменшуватиметься від великого значення до меншого.

Приклад циклу For.. Next із зростаючим лічильником:

Ця процедура отримує від користувача два числа, у заданому ними діапазоні підсумовує всі цілі числа, а потім виводить на екран набутого значення. Якщо, наприклад, ви запустили цю програму і ввели значення 4 і 8, то будуть підсумовані числа 4, 5, 6, 7 і 8, а на екрані буде відображено число 30.


```
Sub Demo_ForNext()
Dim до As Integer
Dim uStart As String
Dim uEnd As String
Dim uSum As Long
```

```
uStart = InputBox("Введіть натуральне число:")
uEnd = InputBox("Введіть друге натуральне число")
```

```
uSum = 0
```

Введенні значення ще не оброблялись, тому сума має бути нульовою

```
For до = uStart To uEnd
    uSum = uSum + до
Next до
```

Цикл **For..Next**

```
MsgBox "Сума натурального ряду від" & uStart & _
" до" & uEnd & " рівна: " & uSum
```

Повторіть призначення оператора

```
End Sub
```

Приклад циклу For.. Next з убуваючим лічильником:

Цей приклад робить те ж саме, але лічильник циклу працює з убуванням.

```
Sum Demo_ForNextDown()
```

```
Dim до As Integer
Dim uStart As Integer
Dim uEnd As Integer
Dim uSum As Long
```

```
uStart = InputBox("Введіть натуральне число:")
uEnd = InputBox("Введіть друге натуральне число")
```

```
uSum = 0
```

```
For до = uStart To uEnd Step -1
    uSum = uSum + до
Next до
```

Цикл **For..Next**,
убуваючий

```
MsgBox "Сума натурального ряду від" & uStart & _
" до" & uEnd & " рівна: " & uSum
```

```
End Sub
```

1. Введіть обидва приклади, знайдіть відмінності між ними.
2. Запустіть їх на виконання, як перше число введіть 3, а друге - 15. Обчисліть і запишіть результат кожного прикладу.
3. Зробіть те саме, але числа введіть навпаки, тобто спочатку введіть 15, а потім 3.
4. Поясніть отримані результати.

Повторення невизначене число разів: цикли Do

У VBA є дуже могутня інструкція для створення в процедурах і функціях циклічних структур, що повторюються невизначене число разів. Усі такі цикли створюються за допомогою єдиної інструкції — інструкції **Do**. Вона володіє такими можливостями і такою гнучкістю, що у неї фактично чотири конструкції, що належать до двох основних категорій.

Ці категорії складаються з циклів **Do**, які перевіряють умову визначника перед виконанням тіла циклу, і, по-друге, з тих циклів **Do**, які перевіряють цю умову після виконання тіла циклу. Незалежно від того, чи перевіряється визначник до або після

виконання тіла циклу, існує два способи контролювати, скільки разів виконується невизначений цикл.

- *Цикли, контрольовані лічильниками.* У цьому випадку тіло циклу виконується, поки значення деякого лічильника знаходиться нижче або вище вказаної межі; у цьому є певна схожість з циклом For .. Next, за винятком того, що на програмістові лежить відповідальність за ініціалізацію цього лічильника, а також за збільшення або зменшення його значення. Такі цикли Do має сенс писати, коли лічильник не є регулярним, а також коли не можна визначити кінцеву межу, поки не почалося виконання самого циклу. Наприклад, потрібно пройти перші 16 рядків листа, іноді збільшуючи за один раз значення рядка на 1, а іноді за один раз – на 2. У зв'язку з цим, збільшення номера рядка (тобто крок лічильника) міняється, цикл For .. Next використовувати не можна; замість нього треба досліджувати цикл Do.

- *Цикли, контрольовані подіями.* У цьому випадку значенням визначника має значення True або False, залежно від деякої події або дії, що мала місце під час виконання циклу. Наприклад, ви написали цикл, що виконується невизначене число разів, поки користувач не введе відоме число в діалоговому вікні. Введення такого числа якраз і є тією подією, яка завершить цикл. А ось інший приклад: можна виконувати операції над осередками аркуша, поки не буде досягнутий порожній осередок. Перехід в порожній осередок і буде подією, що завершує цикл.

Цикли з перевіркою умови перед виконанням

Щоб умови визначника перевірялися перед виконанням тіла циклу, треба логічний вираз визначника просто помістити на початку інструкцій, які складають "тіло циклу". Це можна реалізувати за допомогою ключових слів While і Until.

Створення циклів за допомогою Do While

Першою циклічною структурою, де умова визначника перевіряється перед виконанням циклу, є **Do While**, що має наступний синтаксис:

Do While condition

Statements

Loop

Тут **condition** представляє логічний вираз для визначника циклу, **statements** — одну або велику кількість інструкцій VBA (або взагалі жодний з них), які складають "тіло циклу"; усі вони виконуються кожного разу, коли виконується цикл. Ключове слово **Loop**, що знаходиться після **statements**, сигналізує що досягнутий кінець циклу, і указує точку, з якої виконується перехід у початок циклу, щоб перевірити умову визначника.

В інструкції **Do While** вираз **condition** знаходиться на початку циклу, тому умова визначника перевіряється перед виконанням самого циклу. Оскільки в цьому випадку використовується ключове слово **While**, цикл виконуватиметься до тих пір, поки логічний вираз, представлений **condition**, матиме значення **True**.

Якщо логічний вираз, представлений condition, буде рівний False при першому виконанні інструкції Do While, то цикл буде просто пропущений без жодного виконання.

Приклад циклу Do While

Ця процедура рахує введені користувачем непарні числа, припиняючи роботу тоді, коли таких чисел стане 10. Потім введені числа виводяться на екран.

Зверніть увагу на використання функції Mod.

```
Sub Count_OddNums()  
Const ocTitle = "Підрахунок непарних чисел"  
Dim OddCount As Integer ' лічильник непарних чисел  
Dim OddStr As String ' рядок для відображення непарних чисел  
Dim Num ' змінна типу Variant для введення  
  
OddStr = " " ' Рядок для чисел поки що порожній  
OddCount = 0 ' Лічильник чисел поки що рівний нулю  
Do While OddCount < 10  
    Num = InputBox("", ocTitle)  
    If (Num Mod 2) <> 0 Then  
        OddCount = OddCount + 1  
        OddStr = OddStr & Num & " "  
    End If  
Loop  
MsgBox "Ви ввели такі непарні числа:" & Chr(13)_  
    & OddStr,, ocTitle  
End Sub
```

Цикл
Do While...Loop

5. Введіть приклад. Виконаєте його.

6. Як зміниться процедура, якщо підраховувати всі числа, кратні 3? А якщо непарні?

Відповідь запишіть у зошит.

7. Процедуру для введення п'яти чисел кратних 3 введіть і перевірте.

Створення циклів за допомогою Do Until

У інструкції **Do** варіант **Do While** — це тільки один з тих способів її створення, в яких умова визначника перевіряється перед виконанням циклу. Другим з цих способів є варіант **Do Until**, що має наступний синтаксис:

```
Do Until condition  
statements  
Loop
```

У цьому випадку **condition** представляє логічний вираз для визначника циклу, а **statements** — інструкції VBA, які складають тіло циклу. Ключове слово **Loop**, що знаходиться після **statements**, сигналізує, що досягнутий кінець циклу, і указує точку, з якої виконується перехід в початок циклу, щоб перевірити умову визначника.

В інструкції **Do until** вираз **condition** знаходиться на початку циклу, тому умова визначника перевіряється перед виконанням самого циклу. У зв'язку з тим, що в цьому випадку використовується ключове слово **Until** (що означає поки немає чого-небудь), цикл виконуватиметься до тих пір, поки логічний вираз, представлений condition, матиме значення **False**.

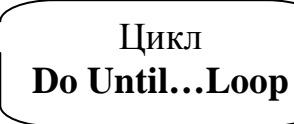
Якщо логічний вираз, представлений **condition**, буде рівний **True** при першому виконанні інструкції **Do Until**, то цикл буде просто пропущений без жодного виконання.

Приклад циклу Do Until

Ця жартівлива процедура пропонує вводити вам дату у форматі Хх.Хх.Хх (наприклад 18.03.03) до тих пір, поки ви не введете дату якої-небудь неділі.

Зверніть увагу на використання функції WeekDay.

```
Sub Stop_AtEvenNums()  
Const evTitle = "В очікуванні неділі..."  
Dim EventFlag As Boolean ' Ознака виходу — Змінна для виходу з циклу  
Dim Num ' Змінна для дати, описана як тип Variant  
EventFlag = False ' Ознака циклу поки рівна False  
Do Until EventFlag = True  
    Num = InputBox("Введіть дату:", evTitle)  
    If Weekday(Num, vbMonday) = 7 Then  
        MsgBox "Ви ввели дату неділі", , "Запам'ятаєте цю дату"  
        EventFlag = True  
    Else  
        MsgBox "Шкода, але відпочити не вдасться", , "Не треба засмучуватися"  
    End If  
Loop  
MsgBox "Добре повеселитися!", , "Ми раді за Вас!"  
End Sub
```



8. Введіть приклад. Виконаєте його.
9. Додайте в нього ще одну перевірку для суботи, з коментарем типу «Ура, це вже завтра!» і також з виходом із циклу.
10. Знайдіть в довідці опис функції WeekDay і законспекуйте в зошит її аргументи.

Цикли з перевіркою умови після виконання

Щоб умови визначника перевірялися після виконання "тіла циклу" логічний вираз визначника треба просто помістити в кінці інструкції, яка складає "тіло циклу", а саме після ключового слова Loop, яке сигналізує про кінець циклу.

Створення циклів за допомогою Do...Loop While

Першою циклічною структурою, де умова визначника перевіряється після виконання циклу, є **Do...Loop While**, що має наступний синтаксис:

Do

statements

Loop While *condition*

У цьому випадку **condition** представляє логічний вираз для визначника циклу, **statements** — одну або велику кількість інструкцій VBA (або взагалі жодну з них), які складають тіло циклу; всі вони виконуються кожного разу коли виконується цикл. Ключове слово **Loop**, що знаходиться після *statements*, сигналізує що досягнутий кінець циклу, і указує точку, з якої виконується перехід в початок циклу, щоб перевірити умову визначника.

У інструкції **Do...Loop While** вираз **condition** знаходиться в кінці циклу, тому умова визначника перевіряється після виконання самого циклу.

Оскільки в цьому випадку використовується ключове слово **While**, цикл виконуватиметься до тих пір, поки логічний вираз, представлене **condition**, матиме значення **True**.

Яким би не було значення логічного виразу, представленого **condition**, цей цикл має виконатися хоча б один раз.

Створення циклів за допомогою Do...Loop Until

Іншою циклічною структурою **Do**, де умова визначника перевіряється після виконання циклу, є **Do...Loop Until**, що має такий синтаксис.

Do

Statements

Loop Until *condition*

У цій інструкції вираз *condition* знаходиться в кінці циклу, тому умова визначника повторюється після виконання самого циклу. Оскільки в цьому випадку використовується ключове слово **Until**, то цикл виконуватиметься до тих пір, поки логічний вираз, представлене *condition*, матиме значення **False**.

При виконанні циклу **Do...Loop Until** спочатку виконуються інструкції, представлені в *statements*. Коли досягнуте ключове слово **Loop**, перевіряється логічний вираз, представлене **condition**; якщо його значення **False**, то відбувається перехід у початок структури і знову виконується "тіло циклу". Коли в кінці циклу знов досягнуте ключове слово **Loop** і якщо значення **condition** змінилося на **True**, то виконання продовжується з інструкцій, що знаходяться після строки із словом **Loop**.

Яким би не було значення логічного виразу, представленого **condition**, цей цикл має виконатися хоча б один раз.

Приклад циклу Do...Loop Until

У процедурі, приведеній нижче, потрібно вгадати число, що випадково згенеровано, від 1 до 10. Цикл виводитиме вікно до тих пір, поки ви не вгадаєте «загадане». У кінці циклу ви дізнаєтеся, скільки ж спроб вам знадобилося.

Зверніть увагу на генератор випадкових чисел.

```
Sub Угадайка()  
Dim EventFlag As Boolean  
Dim g As Integer  
EventFlag = False  
Counter = 0 ' лічильник ваших спроб  
Randomize ' запуск генератора випадкових чисел  
z = Round(10 * Rnd + 1, 0) ' генерація випадкового числа від 1 до 10  
Do  
g = InputBox("Вгадайте число від 1 до 10")  
If g = z Then  
EventFlag = True  
MsgBox "Ти вгадав з " & Counter & " разу"  
End If  
Counter = Counter + 1  
Loop Until EventFlag  
End Sub
```

11. Введіть приклад. Знайдіть в довідці генератор випадкових чисел і законспекуйте опис і приклади.

12. Подумайте, за допомогою яких циклічних конструкцій цей приклад записати можна, а за допомогою яких не можна. Відповідь напишіть в зошиті і обґрунтуйте.

13. Запишіть цю програму в зошиті за допомогою іншої циклічної конструкції по вашому вибору. Введіть і виконайте її.

14. Виведіть повідомлення про удачливість гравця, залежно від числа спроб, наприклад, при спробах більше 5 — «Щось день сьогодні не дуже!», а менше «Ну ти і везунчик!».

Варіанти завдань

1. Напишіть програму, яка виводить усі числа першої сотні, що закінчуються на 5.
2. Напишіть програму, яка вводить число, а виводить таблицю множення цього числа на всі числа від 1 до 9.
3. Напишіть програму, яка виводить таблицю ступенів з 0-ї по 9-у числа 2.
4. Напишіть програму, яка виводить таблицю значень квадратного кореня на інтервалі $[2;4]$ з кроком 0.1
5. Напишіть програму виведення таблиці квадратів перших 10 цілих додатних чисел.
6. Напишіть програму, виведення всіх чисел, що діляться на 13 без залишку в інтервалі $[1;100]$.
7. Напишіть програму обчислення середнього арифметичного 10-и введених чисел.
8. Напишіть програму, яка 10 разів виводить Ваше ім'я і прізвище.
9. Напишіть програму, обчислення суми перших n цілих додатних чисел. Кількість підсумовуваних чисел n вводиться на початку роботи програми.

10. Напишіть програму, яка обчислює суму перших n членів ряду $1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$.

Кількість підсумовуваних членів ряду задається під час роботи програми.

11. Напишіть програму, яка виводить таблицю значень функції $y = -2.4x^2 + 5x - 3$ у діапазоні $[-2;2]$ з кроком 0.5.

12. Напишіть програму, яка виводить таблицю значень функції $y = |x|$ у діапазоні $[-4;4]$ з кроком 0.5.

13. Напишіть програму, яка виводить таблицю значень функції $y = |x + 2|$ у діапазоні $[-4;4]$ з кроком 0.5.

14. Напишіть програму, яка виводить таблицю значень функції $y = |x - 2| + |x + 1|$ у діапазоні $[-4;4]$ з кроком 0.5.

15. Напишіть програму, яка виводить таблицю вартості яблук у діапазоні від 1 кг до 100 г з кроком 100 р. Вартість 1 кг яблук вводиться під час роботи програми.

16. Напишіть програму підрахунку суми і середнього арифметичного всіх цілих додатних чисел із заданого діапазону. Діапазон задається під час роботи програми.

17. Напишіть програму обчислення факторіалу $y = n!$ за формулою $n! = 1 * 2 * 3 * \dots * (n - 1) * n$; $0! = 1$. Значення n вводиться під час роботи програми.

18. Напишіть програму обчислення $y = 1! + 2! + 3! + \dots + n!$. Введення n здійснюється під час роботи програми.

19. Напишіть програму визначення числа Ейлера e – підстави натурального логарифма, за допомогою ряду: $e = 1 + 1/1! + 1/2! + 1/3! + \dots + 1/n!$ для всіх значень n від 1 до m . Введення m здійснюється під час роботи програми.

20. Напишіть програму обчислення $e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \dots + \frac{x^n}{n!} + \dots = 1 + \sum_{i=0}^n \frac{x^i}{i!}$ із заданою точністю $\varepsilon > 0.0$, а також загальне число доданків n . Кожен черговий доданок обчислюється через попередній за наступною рекурентною формулою: $u_i = \frac{x}{i} u_{i-1}$, де $i = 1, 2, 3$ і так далі.

Введення x і ε здійснюється під час роботи програми.

21. Напишіть програму обчислення числа π із заданою точністю $\varepsilon > 0.0$ за допомогою наступного ряду: $\pi/4 = 1/1 - 1/3 + 1/5 - 1/7 + 1/9 \dots$, загальний член якого обчислюється за формулою: $(-1)^k / (2k + 1)$, де k – номер члена ряду. Підрахувати і вивести при цьому також загальне число підсумованих членів ряду. Введення ε здійснюється під час роботи програми.

22. Напишіть програму визначення варіанта лабораторної роботи за двома останніми цифрами номера залікової книжки. Варіант лабораторної роботи обчислюється шляхом віднімання 30 з числа, утвореного двома останніми цифрами номера залікової книжки, до

тих пір, поки воно більше 30. Дві останні цифри номера залікової книжки вводяться під час роботи програми.

23. Напишіть програму добування квадратного кореня $y = \sqrt{x}$ методом Ньютона із заданою точністю $\varepsilon > 0.0$ за наступною рекурентною формулою: $y_{n+1} = y_n + (x / y_n - y_n) / 2$, де $n = 1, 2, 3$ і так далі. Введення x і ε здійснюється під час роботи програми.

24. Напишіть програму визначення значення величини: $y = \sum_{n=1}^m \frac{n}{n+1}$. Введення m здійснюється під час роботи програми.

25. Напишіть програму визначення значення величини: $y = \prod_{n=1}^m ((n+1)/n! + n)$. Введення m здійснюється під час роботи програми.

26. Напишіть програму, яка послідовно вводить у змінну цілого типу довільні числа до тих пір, поки чергове введене число не буде рівне нулю, і визначає їх суму.

27. Напишіть програму, яка послідовно вводить у змінну цілого типу довільні числа до тих пір, поки чергове введене число не буде рівне нулю, і визначає їх середнє арифметичне.

28. Напишіть програму, яка послідовно вводить n разів у змінну цілого типу довільні числа і визначає їх середнє арифметичне.

29. Напишіть програму друку таблиці переведення температури за шкалою Фаренгейта в температуру за шкалою Цельсія в діапазоні від x до y з кроком z за формулою $C(F) = (5/9)(F - 32)$, де C – температура за шкалою Цельсія, F – температура за шкалою Фаренгейта. Введення x , y і z виконується під час роботи програми.

30. Напишіть програму друку таблиці переведення температури за шкалою Фаренгейта в температуру за шкалою Цельсія в діапазоні від x до y з кроком z за формулою $F(C) = (9/5)(C + 32)$ $C(F) = (5/9)(F - 32)$, де C – температура за шкалою Цельсія, F – температура за шкалою Фаренгейта. Введення x , y и z виконується під час роботи програми.

Контрольні питання

1. Що таке цикли і для чого вони призначені?
2. Які типи циклів ви знаєте?
3. Що таке визначник циклу? У яких структурах він є числом?
4. Чим відрізняються цикли передумови й умови поста?
5. Запишіть синтаксис усіх циклів і охарактеризуйте його за таким планом: синтаксис; тип циклу; тип умови; характер визначника. Наприклад:

Цикл	Тип циклу	Тип умови	Визначник
Do <i>Statements</i> Loop Until condition	невизначений	післяумова	умова

1.5.5. Лабораторна робота № 5 «Масиви»

Блок-схеми циклічних структур

Суть блоку <i>цикл</i> полягає в багатократному повторенні деяких операторів (операторів циклу, тобто операторів, що знаходяться в циклі). Блок цикл може бути зображений двома способами: детальним і компактним.		
Детальний спосіб		<p>Якщо <i>Умова</i> істинна, то виконуються Оператори циклу, потім виконується <i>Зміна змінної циклу</i>, після чого знову перевіряється <i>Умова</i>. Ці дії виконуються багато разів до тих пір, поки <i>Умова</i> не стане помилковою. Якщо <i>Умова</i> помилкова, то відбувається вихід з циклу.</p> <p>Цикл використовується для того, щоб багато разів повторювати деякі дії, які й оформляються як Оператори циклу.</p>
Слід зазначити, що існує декілька модифікацій цієї схеми: «так і ні» можуть мінятися місцями і «Зміна змінної циклу» може стояти перед перевіркою «Умови». Ці модифікації змінюють кількість виконань «Операторів циклу» і програміст повинен уміти скласти цикл так, щоб «Оператори циклу» виконалися необхідне йому число разів.		
Компактний		<p>«Нач. зн.» означає початкове значення змінної циклу. «Кінц. зн.» означає останнє значення змінної циклу, при якому Оператори циклу виконуватимуться. «Крок» визначає інтервал між двома сусідніми значеннями змінної циклу.</p>

1. Побудуйте блок-схеми для всіх прикладів процедур з циклічними процесами, створеними в попередній лабораторній роботі

Поняття про масиви

Масив — це набір змінних одного типу даних із загальним ім'ям. Масивами зазвичай користуються, щоб представити списки або таблиці з інформацією, де всі елементи дані одного типу, тобто в списку знаходяться тільки числа Integer, дані одного з типів String, Currency і т. д. Масив дає можливість зберігати й обробляти всі свої елементи, використовуючи одну змінну. Крім того, що скорочується загальна кількість імен змінних, іншою істотною перевагою є те, що для обробки елементів у масиві можна використовувати циклічні структури (в основному такі як For.Next або For.Each). З їх допомогою можна малим числом інструкцій переробити "гори" даних. Якщо для тих же завдань на кожен елемент даних задавати по змінній, то число інструкцій зросте до декількох сотень.

Що таке одновимірні масиви?

Найменш складний масив — це список елементів даних: масив такого роду називається простим або одновимірним. Одновимірні масиви називаються так тому, що у списку тільки одна міра — довжина.

На рис. 1 показана схема одновимірного масиву.

Одновимірний масив	
10.2	←0-й елемент
1.2	←1-й елемент
9.3	←2-й елемент
88.1	←3-й елемент
23.4	←4-й елемент
12.1	←5-й елемент
41.2	←6-й елемент
55.4	←7-й елемент
86.1	←8-й елемент
99.02	←9-й елемент

Рис. 1. Одновимірний масив з 10 елементів типу Double з нумерацією від 0 до 9

Кожна «цеглинка» даних з масиву називається його елементом. У масиві на рис. 1 міститься 10 елементів, в кожному з них зберігається число типу Double. Зверніть увагу, що елементи масиву, яких всього 10, пронумеровані від 0 до 9 (цей спосіб нумерації, поширений в програмуванні, відомий як нумерація з нижньою межею — 0).

Щоб дістати доступ до окремого елемента масиву, вказується ім'я масиву, за яким слідує номер його елемента, чий вміст треба узяти або змінити (цей номер називається індексом). Індекс завжди треба брати в дужки. Наприклад, якщо масив на рис. 1 називається NumArray, то в наступній інструкції змінної AnyNum привласнюється число 55,4.

AnyNum = NumArray(7)

У цій інструкції число 7 — це індекс масиву; звернете увагу, що він поміщений у дужки і ніякими пропусками не відокремлений від імені масиву. З огляду на те, що нумерація починається з 0, ця інструкція насправді посилається на восьмий елемент вказаного масиву. Індеси використовуються і при внесенні даних до окремого елемента масиву. У наступній інструкції число 12 запам'ятовується у восьмому елементі масиву (рис.1):

NumArray(7)= 12

Коли VBA виконує цю інструкцію, то поміщає значення 12 у вказаний елемент масиву, затираючи значення, що при цьому знаходилося в ній, — знову ж таки, як і при будь-якому іншому привласненні. Елемент масиву можна використовувати в будь-якому виразі VBA точно так, як і значення будь-якої константи або змінної.

- Індекс масиву слід брати в круглі дужки.
- Як індекси масивів потрібно використовувати цілі числа.

Не відокремлюйте пропусками імена масивів та індекси: у програмі набирайте ім'я, дужки й індекси як одне слово

Статичні і динамічні масиви

Звичайне число елементів вказується при його описі. Опис повідомляє VBA, яка величина у кожній з розмірностей масиву. Після того як масив, описаний VBA, виділяє під всі його елементи потрібну кількість пам'яті. Для масиву на рис. 1 її треба виділити стільки, щоб поміщалася 10 чисел типу Double, для масиву на рис. 2 потрібно пам'ять під 20 таких чисел.

VBA резервують пам'ять під елементи масиву на весь час існування його змінної. Такого роду масиви називаються *статичними*, оскільки число елементів не міняється.

Вибір розміру масиву може викликати значні труднощі, коли невідома кількість даних, які в ньому зберігатимуться або коли це число може сильно мінятися. Якщо для цього іноді потрібно 100 значень, а іноді — тільки 10, то місце для 90 значень буде витрачатися даремно (різниця між найбільшою і найменшою кількістю елементів). Для такого роду масивів у VBA передбачений особливий тип масивів, які називаються *динамічними*. Своїй назві вони зобов'язані тому, що в них можна міняти число елементів по ходу виконання програми.

Опис масивів

Вам вже відома інструкція **Dim**, вживана для опису змінних. Ця ж інструкція використовується для опису масивів. Фактично слово **Dim** є скороченням від англійського Dimension — розмірність.

При описі масиву за допомогою інструкції **Dim** використовується наступний синтаксис:

Dim VarName ([Subscripts]) [As Type]

VarName представляє будь-яке ім'я масиву, яке відповідає стандартним правилам VBA. **Subscripts** є розмірність (розмірності) масиву. Для одновимірного масиву в **Subscripts** описується одна розмірність; для двовимірного — дві розмірності (вирази для кожної відокремлено комою) і так далі, до тієї кількості розмірностей, яке вам потрібно.

Вираз **Subscripts** має такий синтаксис:

[lower To] upper[[lower To] upper].

Змінна **Lower** вказує на нижню межу індексу масиву, а **upper** — на верхню. Зверніть увагу, що у виразі верхня межа обов'язкова, а нижньої немає. Якщо буде вказана тільки верхня межа, то VBA нумеруватиме елементи масиву від 0 до значення **upper**.

Якщо у виразі **Subscripts** знаходитимуться **lower to**, то це значно полегшуватиме розуміння програм і знаходитиме в них помилки, що зробить програми зрозумілішими і надійнішими. Використання **lower To** також дає можливість вказати початкове значення індексу, яке відрізняється і від 0. Наприклад, якщо є необхідність створити масив з елементами, пронумерованими від 5 до 10 або від -5 до 0, то така можливість існує.

Як і в стандартних змінних, тип даних можна описати і для масивів. Для цього в опис масиву треба включити такий вираз, як **As Type**. Тоді у кожного елемента масиву буде вказаний тип даних. **Type** представляє будь-який з можливих у VBA тип даних: **Currency**, **Double**, **String** і так далі. Якщо **type** буде пропущений, то у всіх елементів масиву буде тип **Variant**. При ініціалізації елементи числових масивів елементи набувають значення 0, а в строкових — значення порожнього рядка.

Скласти блок-схему алгоритму, написати, відладити і протестувати програму, що реалізовує наступне:

- a) Сформувати випадковим чином одновимірні числові масиви заданого розміру і діапазону;
- b) Виконати задані перетворення;
- c) Обчислити деякі величини;
- d) Вивести масиви і значення обчислених величин на екран.

Варіанти завдань

- 1.a) Одновимірний масив з 10 елементів цілого типу в діапазоні [-10, 20].
 - b) Замінити в початковому масиві одиницями елементи з четвертого по сьомий.
 - c) Обчислити суму елементів масиву, що залишилися.
 - d) Вивести початковий, отриманий масиви і значення суми.
- 2.a) Одновимірний масив з 12 елементів цілого типу в діапазоні [-1, 36].
 - b) Утворити новий масив, перенісши в нього чотири останні елементи початкового масиву.
 - c) Обчислити суму елементів отриманого масиву.
 - d) Вивести початковий, отриманий масиви і значення суми.
- 3.a) Одновимірний масив з 10 елементів цілого типу в діапазоні [45, 100].
 - b) Вставити два елементи по 1000, починаючи з четвертого елемента початкового масиву, зрушуючи його елементи.
 - c) Обчислити суму елементів отриманого масиву.
 - d) Вивести масив до і після зміни, і значення суми.
- 4.a) Одновимірний масив з 8 елементів цілого типу в діапазоні [16, 40].
 - b) Переставити четвертий елемент масиву на останнє місце, зрушуючи елементи масиву.
 - c) Обчислити суму елементів масиву, починаючи з п'ятого.
 - d) Вивести масив до і після зміни, і значення суми.

- 5.a)** Два одновимірні масиви по 6 елементів цілого типу в діапазоні $[-10, 10]$.
- b)** Переставити три перші елементи першого масиву на три останні місця другого масиву.
- c)** Обчислити суми початкового і перетвореного масивів.
- d)** Вивести два початкові масиви, змінений масив і значення суми.
- 6.a)** Два одновимірні масиви по 6 елементів цілого типу в діапазоні $[1, 40]$.
- b)** Переставити на друге місце в другому масиві, зрушуючи його елементи, другий елемент першого масиву.
- c)** Знайти мінімальний елемент перетвореного другого масиву.
- d)** Вивести початковий, отриманий масиви і мінімальний елемент.
- 7.a)** Два одновимірні масиви по 6 елементів цілого типу в діапазоні $[10, 20]$.
- b)** Переставити на п'яте місце в першому масиві, зрушуючи його елементи, останній елемент другого масиву.
- c)** Знайти максимальний елемент перетвореного другого масиву.
- d)** Вивести початковий, отриманий масиви і максимальний елемент.
- 8.a)** Одновимірний масив з 10 елементів цілого типу в діапазоні $[-14, 80]$.
- b)** Зрушити елементи початкового масиву на сім значень праворуч, звільняючи сім місць, і заповнити їх нулями.
- c)** Знайти максимальний елемент перетвореного другого масиву.
- d)** Вивести початковий, отриманий масиви і максимальний елемент.
- 9.a)** Два одновимірні масиви по 5 елементів цілого типу в діапазоні $[-100, 100]$.
- b)** Переставити на третє місце в першому масиві, зрушуючи його елементи, четвертий елемент другого масиву.
- c)** Обчислити суму елементів отриманого масиву.
- d)** Вивести початковий, отриманий масиви і значення суми.
- 10. a)** Два одновимірні масиви по 5 елементів цілого типу в діапазоні $[-4, 18]$.
- b)** Переставити четвертий елемент першого масиву на четверте місце в другому масиві, перший масив стиснути.
- c)** Знайти результат помноження елементів першого масиву.
- d)** Вивести два початкових, два отримані масиви і значення результату помноження.
- 11. a)** Одновимірний масив з 10 елементів цілого типу в діапазоні $[50, 200]$.
- b)** Видалити з початкового масиву елементи з другого по сьомий, зрушуючи його елементи.
- c)** Обчислити суму елементів отриманого масиву.
- d)** Вивести початковий, отриманий масиви і значення суми.
- 12. a)** Два одновимірні масиви по 5 елементів цілого типу в діапазоні $[-80, -30]$.
- b)** Поміняти місцями треті елементи цих масивів.
- c)** Обчислити суму елементів першого масиву до і після перетворення.
- d)** Вивести початковий, отриманий масиви і значення сум.
- 13. a)** Одновимірний масив з 6 елементів цілого типу в діапазоні $[-10, 20]$.
- b)** Видалити з початкового масиву другий елемент і стиснути масив (значення видаленого елемента надрукувати).
- c)** Знайти мінімальний елемент перетвореного масиву.
- d)** Вивести початковий, отриманий масиви і значення мінімального елемента.
- 14. a)** Два одновимірні масиви по 5 елементів цілого типу в діапазоні $[16, 80]$.
- b)** переставити чотири останні елементи першого масиву на чотири перші елементи в другому масиві.
- c)** Знайти мінімальний елемент перетвореного другого масиву.
- d)** Вивести початковий, отриманий масиви і значення мінімального елемента.

15. a) Одновимірний масив з 8 елементів цілого типу в діапазоні [-5, 20].
 b) Замінити в початковому масиві два перші елементи одиницями.
 c) Обчислити суму елементів масиву, починаючи з третього.
 d) Вивести початковий, отриманий масиви і значення суми.
16. a) Одновимірний масив з 5 елементів цілого типу в діапазоні [-4, 100].
 b) Доповнити початковий масив, починаючи з третього номера, п'ятьма двійками, зрушуючи елементи масиву.
 c) Знайти максимальний елемент перетвореного масиву.
 d) Вивести початковий, отриманий масиви і значення максимального елемента.
17. a) Два одновимірні масиви по 5 елементів цілого типу в діапазоні [-10, 30].
 b) Доповнити другий масив, починаючи з другого номера, трьома останніми елементами першого масиву.
 c) Знайти максимальний елемент перетвореного другого масиву.
 d) Вивести два початкових, два отриманих масиви і значення максимального елемента.
18. a) Одновимірний масив з 10 елементів цілого типу в діапазоні [40, 120].
 b) Видалити з початкового масиву три перші елементи, зрушуючи елементи масиву, надрукувати їх значення.
 c) Знайти результат помноження елементів перетвореного масиву.
 d) Вивести початковий, отриманий масиви і значення результату помноження.
19. a) Одновимірний масив з 9 елементів цілого типу в діапазоні [-30, -1].
 b) Переставити шостий елемент початкового масиву на перше місце, зрушуючи його елементи, вивести його значення.
 c) Обчислити результат помноження елементів перетвореного масиву.
 d) Вивести початковий, отриманий масиви і значення результату помноження.
20. a) Одновимірний масив з 12 елементів цілого типу в діапазоні [-50, 50].
 b) Видалити з початкового масиву три перші елементи і чотири останніх, вивести значення елементів, що видаляються.
 c) Знайти максимальний елемент перетвореного масиву.
 d) Вивести початковий, отриманий масиви і значення суми.
21. a) Одновимірний масив з 10 елементів цілого типу в діапазоні [40, 70].
 b) Замінити п'ять останніх елементів початкового масиву числом 100.
 c) Знайти максимальний елемент перетвореного масиву і його порядковий номер у масиві.
 d) Вивести початковий, отриманий масиви, значення максимального елемента і його порядковий номер.
22. a) Одновимірний масив з 10 елементів цілого типу в діапазоні [-80, -50].
 b) Зрушити елементи початкового масиву на 5 значень до початку, заповнивши порожні місця одиницями.
 c) Обчислити суму елементів нового масиву.
 d) Вивести початковий, отриманий масиви і значення суми.
23. a) Одновимірний масив з 10 елементів цілого типу в діапазоні [0, 40].
 b) Доповнити початковий масив трьома одиницями, починаючи з п'ятого елемента, не зрушуючи елементи початкового масиву.
 c) Знайти мінімальний елемент перетвореного масиву.
 d) Вивести початковий, отриманий масиви і значення мінімального елемента.
24. a) Одновимірний масив з 10 елементів цілого типу в діапазоні [-17, 60].
 b) У кінець початкового масиву додати шість дев'яток.
 c) Знайти максимальний елемент перетвореного масиву.
 d) Вивести початковий, отриманий масиви і значення максимального елемента.

25. a) Два одновимірні масиви (по 5 елементів в кожному) з цілих чисел у діапазоні [-30, 18].
 b) Замінити два останні елементи першого масиву четвертим і шостим елементами першого масиву.
 c) Знайти мінімальний елемент до і після перетворень.
 d) Вивести початковий, отриманий масиви і значення мінімальних елементів.
26. a) Одновимірний масив із 7 елементів цілого типу в діапазоні [-5, 100].
 b) Інвертувати початковий масив (змінити знаки елементів на протилежні).
 c) Обчислити суму елементів отриманого масиву.
 d) Вивести початковий, отриманий масиви і значення суми.
27. a) Одновимірний масив з 10 елементів цілого типу в діапазоні [4, 46].
 b) Утворити новий масив, перенісши в нього елементи початкового масиву з 4 по 9 елементів включно.
 c) Обчислити суму елементів отриманого масиву.
 d) Вивести початковий, отриманий масиви і значення суми.
28. a) Одновимірний масив з 9 елементів цілого типу в діапазоні [-18, 42].
 b) Привласнити першому елементу масиву значення 0, решту елементів зрушити.
 c) Обчислити суму елементів отриманого масиву.
 d) Вивести початковий, отриманий масиви і значення суми.
29. a) Одновимірний масив з 5 елементів цілого типу в діапазоні [-32, -5].
 b) Поставити на перші три місця в початковому масиві -1, зрушуючи його елементи.
 c) Знайти мінімальний елемент перетвореного масиву.
 d) Вивести початковий, отриманий масиви і значення мінімального елемента.
30. a) Одновимірний масив з 8 елементів цілого типу в діапазоні [4, 36].
 b) Поміняти місцями четвертий і сьомий елементи початкового масиву.
 c) Знайти максимальний елемент перетвореного масиву.
 d) Вивести початковий, отриманий масиви, значення переставлених елементів і максимальний елемент.

Приклад виконання завдання

- a) Одновимірний масив з 10 елементів цілого типу в діапазоні [-20, 80].
 b) Другий масив скласти з першого шляхом перенесення в нього непарних елементів початкового масиву. Вивести отриманий масив на екран. Потім видалити з цього масиву третій за номером елемент, зрушуючи його елементи.
 c) Обчислити суму елементів початкового масиву і знайти мінімальний елемент перетвореного масиву.
 d) Вивести початковий, два отримані масиви і значення суми і мінімального елемента.

1) Текст програми з поясненнями

Sub ArrayOne()

' Масив First статичний з 10 елементами від 0 до 9

Dim First(9) As Integer

' Масив Second динамічний

Dim Second() As Integer

'Виконуємо пункт а) 'Заповнення початкового масиву випадковими числами

Randomize 'Запуск лічильника випадкових чисел

For i = 0 To 9

'Привласнити елементу масиву випадкове число від 1 до 100 із зрушенням на -20

First(i)= Int((100) * Rnd - 21)

Next i

'Заповнити рядок елементами масиву,

'перемежуючи їх пропусками, і вивести на екран початковий масив

```

For i = 0 To 9
FirstStr = FirstStr & First(i)& " "
Next i
MsgBox FirstStr,, "Початковий масив"
'Виконуємо пункт b) Підрахувати кількість непарних чисел для визначення розмірності
'масиву Second
For j = 0 To 9
    If (First(j) Mod 2) <> 0 Then
        до = до + 1
    End If
Next j

'Вивести на екран кількість непарних елементів
MsgBox до,, "Кількість непарних елементів"
'Визначити розмірність масиву Second елементів від 0 до k-1
ReDim Second(до - 1)
i = 0 'лічильник масиву Second обнулити
For j = 0 To 9
    If (First(j) Mod 2) <> 0 Then
        'Записати в масив Second непарне число з масиву First
        Second(i)= First(j)
        'лічильник масиву Second збільшується тільки для непарних чисел масиву First
        i = i + 1
    End If
Next j
'Заповнити рядок елементами масиву, перемежуючи їх пропусками, і вивести на 'екран
другий масив з непарними елементами
For i = 0 To - 1
    SecondStr = SecondStr & Second(i)& " "
Next i
MsgBox SecondStr,, "Перетворений масив"
'Якщо величина масиву Second більше 3, то викинути з нього елемент з
номером 3
If до > 3 Then
    'Починаючи з елемента з номером три привласнювати йому наступний елемент цього
ж масиву
    'Щоб не вийти за межі масиву, завершуємо' на передостанньому елементі.
    For i = 3 To - 2
        Second(i)= Second(i + 1)
    Next i
    ' Оскільки кількість елементів у масиві Second на одиницю зменшилося,
    ' змінюємо його розмірність на 1. Ключове слово Preserve не дозволить
    ' обнулити вже існуючі елементи масиву
    ReDim Preserve Second(до - 2)
    'Сформувати рядок для виведення стислого масиву на екран і вивести його на екран
    For i = 0 To - 2
        ThirdStr = ThirdStr & Second(i)& " "
    Next i
    MsgBox ThirdStr,, "Стислий масив"
Else

```

' Якщо в масиві Second менше 4 елементів, він стисненню не підлягає
 MsgBox "Масив не підлягає стисненню, оскільки його розмір менше 4", "Стислий масив"
 End If

'Виконуємо пункт з)

'Підраховуємо суму елементів початкового масиву

Sum = 0

For i = 0 To 9

'Додаємо до суми кожен елемент початкового масиву

Sum = Sum + First(i)

Next i

MsgBox Sum, "Сума елементів початкового масиву"

'Знаходимо мінімальний елемент

'Приймаємо за мінімальний елемент нульовий елемент масиву

Min = Second(0)

For i = 1 To - 2

'Починаючи з першого елемента масиву перевіряємо, якщо він менший

'мінімального, то записуємо його в мінімальне значення

If Second(i) < Min Then

Min = Second(i)

End If

Next i

MsgBox Min, "Мінімальний елемент стислого масиву"

'Вивести на екран все обчислене в програмі 'виведено на екран, тому наступну команду можна не виконувати, але виводячи на екран 'все одночасно легко перевірити правильність роботи всієї програми

MsgBox "Початковий масив: " & FirstStr & Chr(13) & _

"Перетворений масив: " & SecondStr & Chr(13) & _

"Стислий масив: " & ThirdStr & Chr(13) & _

"Сума елементів початкового масиву: " & Sum & Chr(13) & _

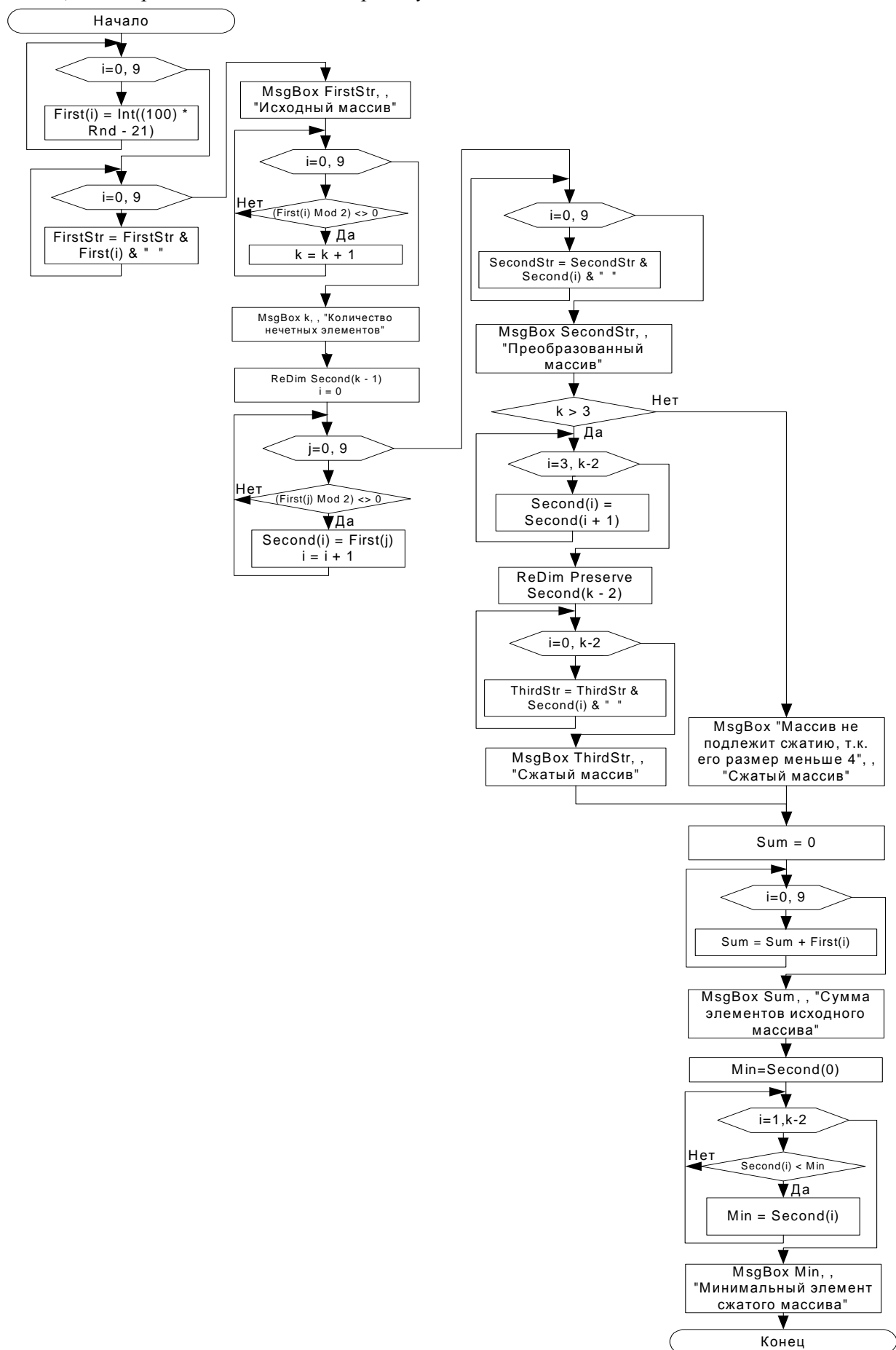
"Мінімальний елемент стислого масиву: " & Min & _

"Підсумки роботи програми"

End Sub

<i>Таблиця результатів тестування</i>	
Початковий X	30 56 -5 27 6 -15 0 43 78 -10
Перетворений Y	-5 27 -15 43
Стислий Y	-5 27 -15
s=	210
min=	-15

2) Створення блок-схеми алгоритму



1.5.6. Лабораторна робота № 6 «Двовимірні числові масиви»

Вкладені цикли

Цикли можна розміщувати усередині інших циклів точно так, як і це робиться з інструкціями If.Then. Розміщені таким чином цикли зазвичай називаються вкладеними. Вкладені циклічні структури можуть бути будь-якого типу (змішані цикли For і Do) і можуть розташовуватися на будь-якому рівні.

У кожного з таких циклів має бути унікальна змінна лічильника, якщо вона використовується в певній циклічній конструкції.

Багатовимірні масиви

Одновимірні масиви добре працюють з простими списками даних. Проте часто в програмах доводиться представляти інформацію, яка знаходиться в елементах таблиці, що мають рядки і стовпчики, — неначебто вона знаходилася в осередках Excel. Для цього необхідно використовувати багатовимірний масив.

На рис. 2 представлена найбільш поширена форма багатовимірного масиву — двовимірний масив.

Двовимірний масив		
Стовпчик 0	Стовпчик 1	
12.4	10.2	Рядок 0
2.5	1.2	Рядок 1
16.7	9.3	Рядок 2
8.18	88.1	Рядок 3
0.125	23.4	Рядок 4
21.45	12.1	Рядок 5
17.8	41.2	Рядок 6
62.37	55.4	Рядок 7
7818.01	86.1	Рядок 8
37.2	99.02	Рядок 9

Рис. 2. Двовимірний числовий масив розмірності 2 на 10 з нумерацією від 0 до 1 і від 0 до 9

Багатовимірні масиви називаються так через те, що у них більш за одну розмірність: довжина (число рядків), ширина (число стовпчиків). На рис. 1 у двовимірному масиву два стовпчики (номер 0 і номер 1) і 10 рядків (пронумеровані від 0 до 9), а елементів всього 20. Як і у випадку з одновимірними масивами, для доступу до елемента багатовимірних масивів використовується їх індексація, тобто для вказівки окремого елемента застосовують номер його стовпчика і номер рядка. Індексація двовимірного масиву багато в чому нагадує вказівку осередку аркуша Excel; перша розмірність масиву відповідає стовпчикам, а друга — рядкам.

Якщо у масиву на рис.1 ім'я NumTable, то наступна інструкція привласнює значення 10,2 (з першого ряду другого стовпчика масиву) змінній AnyNum:

AnyNum = NumTable(1,0)

Аналогічним чином наступна інструкція привласнює значення 2,5 елемента з другого ряду першого стовпчика масиву:

NumTable(1,0)=2,5

Зверніть увагу, що в обох виразах індекси масиву поміщені в круглі дужки і координати стовпчика і рядка відокремлені комами.

Можна створювати масиви більш ніж з двома розмірностями; фактично VBA дає можливість задавати аж до 60 розмірностей. Але робота з масивами 3, 4 і більшими розмірностями є копіткою й утомливою. Але, на щастя, у програмуванні використовуються в основному одно- і двовимірні масиви.

- Індеси багатовимірного масиву необхідно відокремлювати комами.

- Не відокремлюйте пропусками імена масивів та індекси: у програмі набирайте ім'я, дужки й індекси як одне слово.

Скласти блок-схему алгоритму і програму, що реалізовує наступне:

1. Сформувати випадковий двовимірний масив заданого розміру і діапазону.
2. Виконати задані перетворення.
3. Обчислити деякі величини.
4. Вивести масиви і значення обчислених величин на екран.

Варіанти завдань

- 1.a) Двовимірний масив розмірністю 5 на 8 з цілих елементів у діапазоні $[-30, 30]$.
b) Знайти максимальний елемент другого рядка.
c) Обчислити суму елементів третього стовпчика.
d) Вивести початкову матрицю, елементи третього стовпчика і значення максимального елемента й обчислену суму.
- 2.a) Матрицю розмірністю 6 на 5 з випадкових елементів дійсного типу, виведених з точністю до десятих, в діапазоні $[-80, 30]$.
b) Транспонувати початкову матрицю.
c) Знайти мінімальний елемент на головній діагоналі.
d) Вивести початкову і транспоновану матриці та мінімальний елемент.
- 3.a) Матрицю розмірністю 9 на 6 з випадкових цілих елементів у діапазоні $[-50, 50]$.
b) Знайти суму елементів шостого стовпчика і сьомого рядка.
c) Знайти мінімальну з двох сум.
d) Вивести на екран початкову матрицю, обидві суми і мінімальну з них.
- 4.a) Матрицю розмірністю 4 на 4 з випадкових елементів дійсного типу, виведених з точністю до сотих, у діапазоні $[-20, 20]$.
b) Зробити початкову матрицю симетричною щодо головної діагоналі.
c) Знайти результат помноження елементів головної діагоналі.
d) Вивести на екран початкову, симетричну матриці та результат помноження елементів.
- 5.a) Матрицю розмірністю 8 на 7 елементів дійсного типу, виведених з точністю до десятих, у діапазоні $[-60, 40]$.
b) Знайти елементи менші 0 і замінити їх нулями.
c) Порахувати кількість замінених елементів.
d) Вивести на екран початкову і перетворену матриці і кількість замінених елементів.
- 6.a) Матрицю розмірністю 6 на 8 з елементів цілого типу в діапазоні $[-80, 80]$.
b) З початкової матриці утворити підматрицю розміром 4 на 5, починаючи з другого рядка і другого стовпчика початкової.
c) Знайти суму елементів підматриці.
d) Вивести на екран початкову матрицю, підматрицю і суму елементів.
- 7.a) Масив розмірністю 5 на 8 з елементів цілого типу в діапазоні $[-50, 40]$.
b) Сформувати з початкової матриці вектор розмірністю 1 на 5.
c) Непарні елементи вектора знаходяться як мінімальні елементи відповідного рядка, а парні елементи - як максимальні елементи відповідних рядків.
d) Вивести на екран початкову матрицю й отриманий вектор.
- 8.a) Матрицю розмірністю 8 на 9 з елементів цілого типу в діапазоні $[-90, 80]$.
b) Сформувати з початкової матриці вектор розмірністю 9 на 1.
c) Елементами вектора є кількості позитивних елементів відповідних стовпчиків початкової матриці.
d) Вивести на екран початкову матрицю й отриманий вектор.
- 9.a) Матрицю розмірністю 5 на 6 з елементів дійсного типу з точністю до десятих у діапазоні $[-40, 40]$.
b) Знайти кількість позитивних елементів початкової матриці.
c) Елементи на головній діагоналі замінити одиницями.
d) Вивести на екран початкову, перетворену матриці та кількість позитивних елементів.

10. a) Матрицю розмірністю 10 на 8 з елементів цілого типу в діапазоні [-16, 43].
 b) З початкової матриці утворити вектор 1 на 4.
 c) Перший елемент вектора - сума елементів другого рядка матриці, другий - різниця елементів четвертого рядка, третій елемент - результат помноження елементів п'ятого стовпчика, четвертий – результат ділення елементів сьомого стовпчика.
 d) Вивести на екран початкову матрицю й отриманий вектор.
11. a) Матрицю розмірністю 8 на 6 з елементів дійсного типу з точністю до сотих у діапазоні [4, 16].
 b) Визначити кількість елементів, менших 10.
 c) Максимальний елемент матриці замінити цим числом.
 d) Вивести на екран початкову і перетворену матриці.
12. a) Матрицю розмірністю 6 на 6 з елементів цілого типу в діапазоні [-20, 20].
 b) Замінити місцями другий стовпчик і третій рядок.
 c) Для заміни використовувати додатковий вектор.
 d) Вивести на екран початкову, перетворену матриці та додатковий вектор.
13. a) Матрицю розмірністю 5 на 8 з елементів цілого типу в діапазоні [-50, 60].
 b) Змінити знаки елементів матриці на протилежні.
 c) Порахувати кількість нульових елементів.
 d) Вивести на екран початкову, перетворену матриці та кількість нулів.
14. a) Матрицю розмірністю 6 на 5 з елементів цілого типу в діапазоні [-100, 100].
 b) Підрахувати кількість від'ємних елементів.
 c) Замінити третій і четвертий рядки початкової матриці кількістю від'ємних елементів.
 d) Вивести на екран початкову і перетворену матриці.
15. a) Матрицю розмірністю 7 на 7 з елементів дійсного типу (з точністю до сотих) в діапазоні [0, 50].
 b) Зробити дзеркальне відображення щодо елементів третього стовпчика.
 c) Підрахувати кількість додатних елементів.
 d) Вивести на екран початкову, перетворену матриці, вектор обміну і кількість позитивних елементів.
16. a) Матрицю розмірністю 8 на 8 з елементів цілого типу в діапазоні [-50, 50].
 b) Підрахувати кількість елементів більше нуля і кількість елементів менше нуля.
 c) Максимальний елемент матриці замінити кількістю елементів більше нуля, а мінімальний - кількістю елементів менше нуля.
 d) Вивести на екран початкову, перетворену матриці.
17. a) Матрицю розмірністю 9 на 4 з елементів дійсного типу (з точністю до десятих) в діапазоні [-80, 40].
 b) Всі елементи матриці збільшити удвічі.
 c) Знайти суму елементів головної діагоналі.
 d) Вивести на екран початкову, перетворену матриці та суму елементів головної діагоналі.
18. a) Матрицю розмірністю 8 на 10 з елементів цілого типу в діапазоні [-40, 40].
 b) З початкової матриці отримати підматрицю розмірністю 8 на 5 елементів, починаючи з елемента X [1, 1].
 c) Знайти максимальний елемент підматриці.
 d) Вивести на екран початкову матрицю, підматрицю і максимальний елемент.
19. a) Матрицю розмірністю 5 на 6 з елементів цілого типу в діапазоні [-40, 50].
 b) Поміняти місцями елементи п'ятого рядка і п'ятого стовпчика.
 c) Знайти суму елементів вектора обміну.
 d) Вивести на екран початкову, перетворену матриці та суму елементів вектора обміну.
20. a) Матрицю розмірністю 4 на 10 з елементів дійсного типу (з точністю до десятих) в діапазоні [-20, 20].
 b) Знайти мінімальний елемент матриці.
 c) Помножити елементи другого і четвертого стовпчиків на мінімальний елемент.
 d) Вивести на екран початкову, перетворену матриці та мінімальний елемент.

21. a) Матрицю розмірністю 6 на 9 з елементів цілого типу в діапазоні [-60, 40].
b) Парним елементам матриці замінити знак на протилежний.
c) Знайти результат помноження парних елементів матриці.
d) Вивести на екран початкову, перетворену матриці і результат помноження.
22. a) Матрицю розмірністю 9 на 6 з елементів цілого типу в діапазоні [-30, 80].
b) Створити другу матрицю розмірністю 2 на 2 й заповнити її нулями.
c) Замінити елементи початкової матриці, починаючи з п'ятого рядка і четвертого стовпчика, елементами другої матриці.
d) Вивести на екран початкові матриці та перетворену матрицю.
23. a) Матрицю розмірністю 7 на 8 з елементів цілого типу у діапазоні [-50, 50].
b) Утворити другу матрицю розмірністю 7 на 7 і заповнити її випадковими числами в діапазоні від 10 до 20.
c) Відняти з першої матриці другу поелементно.
d) Вивести на екран початкову і перетворену матриці.
24. a) Матрицю розмірністю 5 на 6 з елементів дійсного типу з точністю до десятих в діапазоні [-10, 10].
b) Підрахувати кількість позитивних елементів.
c) Позитивні елементи замінити одиницями.
d) Вивести на екран початкову, перетворену матриці та кількість позитивних елементів.
25. a) Матрицю розмірністю 10 на 8 з елементів цілого типу в діапазоні [-50, 60].
b) Знайти суму елементів, для яких $X[i, j] > 0$.
c) Елементи, для яких $X[i, j] > 0$, зменшити в 10 разів.
d) Вивести на екран початкову, перетворену матриці та суму елементів.
26. a) Матрицю розмірністю 5 на 5 з елементів цілого типу в діапазоні [-50, 60].
b) Перетворити початкову матрицю в симетричну, залишивши без зміни елементи, що стоять над діагоналлю.
c) Знайти різницю елементів, що не змінили свого місця розміщення.
d) Вивести на екран початкову, перетворену матриці та різницю.
27. a) Матрицю розмірністю 8 на 8 з елементів цілого типу в діапазоні [-60, 50].
b) Перетворити початкову матрицю в симетричну, залишивши без зміни елементи, що стоять під головною діагоналлю.
c) Замінити елементи головної діагоналі нулями.
d) Вивести на екран початкову, перетворену матриці (до і після зміни головної діагоналі).
28. a) Матрицю розмірністю 5 на 8 з елементів цілого типу в діапазоні [-50, 60].
b) Транспонувати початкову матрицю.
c) Підрахувати кількість позитивних і негативних елементів матриці.
d) Вивести на екран початкову, перетворену матриці та кількість позитивних і негативних елементів окремо.
29. a) Матрицю розмірністю 7 на 8 з елементів цілого типу в діапазоні [10, 60].
b) Знайти мінімальний і максимальний елементи початкової матриці.
c) Перетворити початкову матрицю так, щоб максимальний елемент стояв на місці мінімального, а мінімальний на місці максимального.
d) Вивести на екран початкову, перетворену матриці.
30. a) Матрицю розмірністю 10 на 7 з елементів цілого типу в діапазоні [-60, 60].
b) Підрахувати кількість позитивних елементів у початковій матриці.
c) Всім позитивним елементам змінити знак на протилежний.
d) Вивести на екран початкову, перетворену матриці та кількість позитивних елементів у початковій матриці.

Приклад виконання завдання

a) Матрицю розмірністю 5 на 7 з випадкових елементів дійсного типу, заданих з точністю до сотих у діапазоні [-30, 40].

b) Знайти мінімальні елементи в нульовому, другому і четвертому рядках і утворити з них одновимірний масив.

c) Знайти суму мінімальних елементів і перетворити початковий масив так, щоб елементи другого стовпчика отриманої матриці були зменшені на отриману суму.

d) Вивести на екран початкову, отриману матриці, матрицю мінімальних елементів і їх суму.

1) Текст програми з поясненнями

{ Робота з двовимірними числовими масивами. }

Sub ArraySecond()

' Початковий Масив First статичний двовимірний (матриця) розмірністю 5 на 7
' з елементами, занумерованими від 0 до 4 і від 0 до 6

Dim First(4, 6) As Single

' Масив Help статичний допоміжний масив з нумерацією

' елементів від 0 до 2 для

' трьох мінімальних значень з рядків з номерами 0, 2, 4

Dim Help(2) As Single

' Перетворений масив Second статичний двовимірний (матриця) розмірністю 5 на 7
' з елементами, занумерованими від 0 до 4 і від 0 до 6

Dim Second(4, 6) As Single

' Виконуємо пункт a) Заповнення початкового масиву випадковими числами'

Randomize 'Запуск лічильника випадкових чисел

For i = 0 To 4

For j = 0 To 6

' Привласнити елементу масиву випадкове число від 1 до 70 із зрушенням на -31

First(i, j) = Round((70) * Rnd - 31, 2)

Next j

Next i

' Заповнити рядок елементами масиву, перемежаючи їх пропусками і початком нового
рядка, 'по зміні індексу і як лічильника рядка вивести на екран початковий масив

For i = 0 To 4

For j = 0 To 6

FirstStr = FirstStr & First(i, j) & " "

Next j

FirstStr = FirstStr & Chr(13)

Next i

MsgBox FirstStr, "Початковий масив"

' Виконуємо пункт b), тобто знаходимо мінімальне значення

' для рядків 0, 2, 4 і записуємо ці значення в масив Help'

' Для рядка 0

Help(0) = First(0, 0)

For j = 0 To 6

If Help(0) > First(0, j) Then Help(0) = First(0, j)

Next j

' Для рядка 2

Help(1) = First(2, 0)

For j = 0 To 6

If Help(1) > First(2, j) Then Help(1) = First(2, j)

Next j

```

' Для рядка 4
Help(2)= First(4, 0)
For j = 0 To 6
If Help(2)> First(4, j) Then Help(2)= First(4, j)
Next j

```

'Заповнити рядок HelpStr елементами масиву Help і вивести його на друк

```

For i = 0 To 2
    HelpStr = HelpStr & Help(i)& " "
Next i
MsgBox HelpStr,, "Масив мінімальних елементів"

```

'Виконуємо пункт с) Підраховуємо суму елементів масиву мінімальних елементів

```

Sum = 0
For i = 0 To 2
'Додаємо до суми кожен елемент початкового масиву
Sum = Sum + Help(i)
Next i
MsgBox Sum,, "Сума елементів початкового масиву"

```

'Заповнення перетвореними елементами другого масиву

```

For i = 0 To 4
For j = 0 To 6
If j = 2 Then
Second(i, j)= First(i, j) - Sum
Else: Second(i, j)= First(i, j)
End If
Next j
Next i

```

'Заповнити рядок елементами масиву, перемежаючи їх пропусками і початком нового рядка

```

'по зміні індексу і як лічильника рядків, і вивести на екран перетворений масив
For i = 0 To 4
For j = 0 To 6
SecondStr = SecondStr & Second(i, j)& " "
Next j
SecondStr = SecondStr & Chr(13)
Next i
MsgBox SecondStr,, "Перетворений масив"

```

*'Вивести на екран все отримане в програмі. Усе обчислене вже виведене на екран
'тому наступну команду можна не виконувати, 'але виводячи на екран все одночасно
' легко перевірити правильність роботи всієї програми*

```

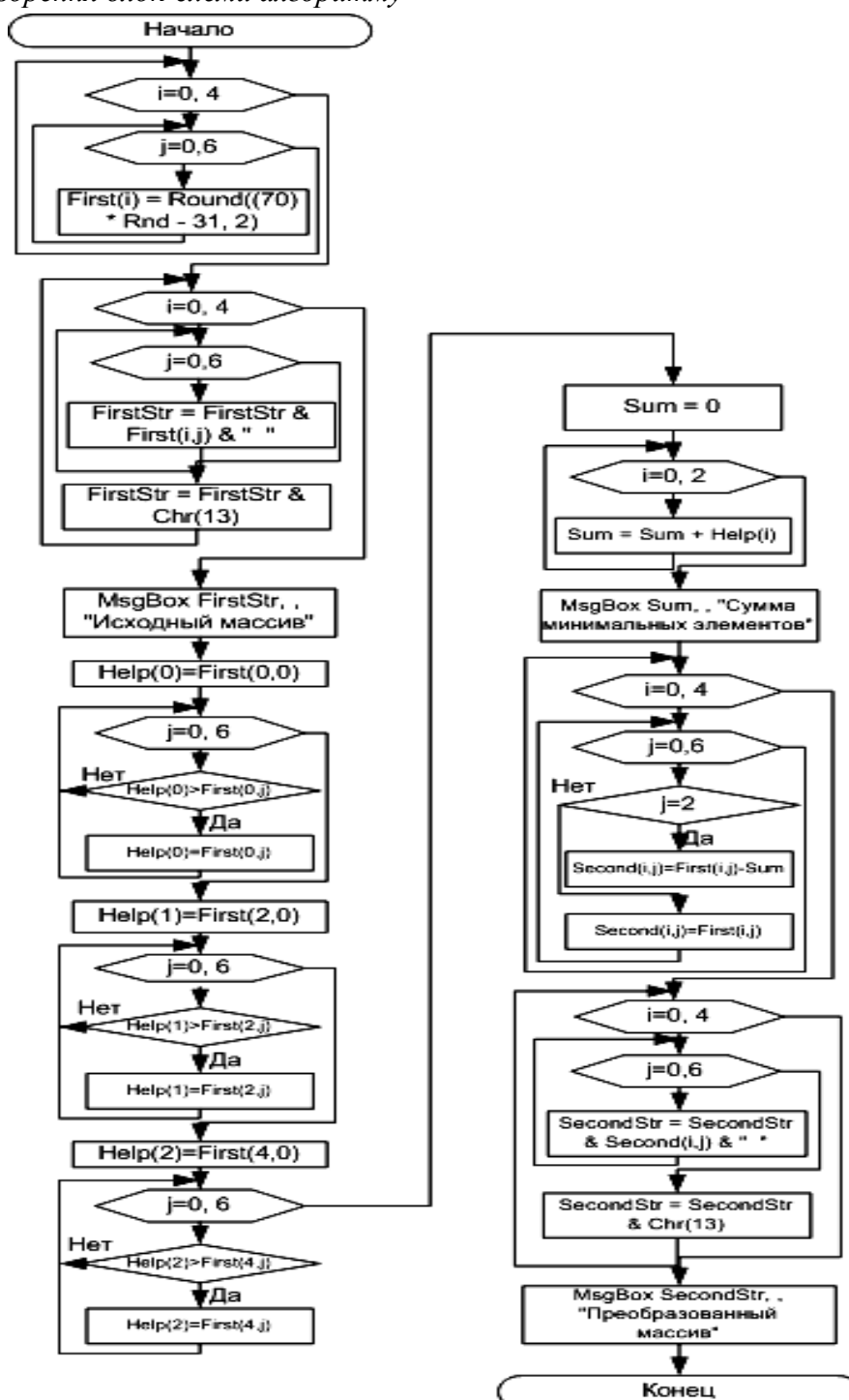
MsgBox "Початковий масив: " & Chr(13)& FirstStr & Chr(13)& _
    "Масив мінімальних елементів рядків 0, 2, 4: " & HelpStr & Chr(13)& Chr(13)& _
    "Сума мінімальних елементів рядків 0, 2, 4: " & Sum & Chr(13)& Chr(13)& _
    "Перетворений масив: " & Chr(13)& SecondStr _
    "Підсумки роботи програми"
End Sub

```

2) Таблиця результатів тестування

Початковий First	-15.96; 31.12; 32.42; -5.29; -16.00; 32.37; 14.74; -19.67; -8.60; -9.18; -21.13; -15.18; 4.16; -16.96; 25.62; -6.10; 2.61; 11.21; 20.65; -7.25; 28.91; 10.66; -4.35; 7.19; 21.54; 32.86; 26.18; -21.47; 7.44; -23.03; 22.58; -26.04; 12.11; 10.80; 28.01.
Перетворений Second	-15.96; 80.41; 32.42; -5.29; -16.00; 32.37; 14.74; -19.67; 40.69; -9.18; -21.13; -15.18; 4.16; -16.96; 25.62; 43.19; 2.61; 11.21; 20.65; -7.25; 28.91; 10.66; 44.94; 7.19; 21.54; 32.86; 26.18; -21.47; 7.44; 26.26; 22.58; -26.04; 12.11; 10.80; 28.01.
Масив Help	-16.00; -7.25; -26.04.
SUM	-49.29.

3) Створення блок-схеми алгоритму



1.5.7 Лабораторна робота № 7 «Процедури і функції» Структуризація програм VBA

Основною структурною одиницею при написанні коду VBA є *процедура* (іноді звана підпрограмою). Зазвичай процедура визначається як набір інструкцій (операторів), направлених на вирішення певної задачі. Розрізняють два види підпрограм: процедури і функції. При виконанні процедури не відбувається повернення ніякого значення в основну програму, тоді як функція завжди повертає певне значення. Процедура починається з ключового слова Sub, а функція - Function. Синтаксис підпрограм:

Процедура	Функція
Sub Ім'я процедури() Код процедури End Sub	Function Ім'я функції() Код функції Ім'яфункції = Вираз End Function

Якщо після імені підпрограми знаходяться порожні дужки, значить, у підпрограму не передаються ніякі параметри, тобто вона працює без параметрів.

Величина, яка повертається функцією, може бути використана багатьма способами - наприклад, як прапорець, що вказує на певний стан або як результат обчислень. Щоб функція повертала значення, *імені функції* потрібно привласнити певний вираз, як показано вище.

Зверніть увагу, що ім'я функції є і змінною, якою має бути привласнене значення. Якщо Вам не потрібно, щоб ваша підпрограма повертала значення, краще використовувати процедуру.

Виклик функцій і процедур

Виклик процедур і функцій можна здійснити з будь-якої процедури або функції. Наприклад, перша з нижче приведених процедур викликає другу, яка відкриває вікно повідомлення з текстом "Всім привіт".

```
Sub main()
    SecondSubroutine
End Sub

Sub SecondSubroutine()
    MsgBox "Привіт всім!"
End Sub
```

При виклику процедури (або функції) (наприклад, з іншої процедури), управління передається процедурі, що викликається (або функції). Потім послідовно виконуються оператори процедури (або функції), що викликається, після чого управління передається початковій (що викликає) процедурі.

Приклад використання підпрограм і функцій

Приведена нижче процедура викликає функцію GetLastName, аргументом якої є повне ім'я у формі Прізвище, Ім'я. Підпрограма передає функції аргумент "Петров, Іван", а функція GetLastName використовує функції VBA InStr Left\$ і повертає тільки прізвище.

```
Sub DisplayLastName()
    Dim StrName As String
    ' Виклик функції GetLastName
    StrName = GetLastName("Петров, Іван")
    MsgBox StrName
End Sub

Function GetLastName(FullName As String) As String
    Dim intLen As Integer
    intLen = InStr(FullName ",")
    GetLastName = Left$(FullName, intLen - 1)
End Function
```

Функція InStr визначає номер позиції, в якій знаходиться кома, а функція Left\$ повертає підрядок, починаючи з першого символу і кінчаючи символом, визначуваним виразом intLen - 1.

1. Введіть обидва приклади, знайдіть відмінності між ними.
2. Запустіть їх на виконання.
3. Поясніть отримані результати.

4. Завдання.

5. Змініть програму роботи з масивами, створену вами в роботі № 5, замінивши групи операторів введення і виведення масивів процедурами, а групи операторів знаходження максимального елемента, мінімального елемента і суми масиву функціями (ви маєте отримати не менше 2 процедур і не менше двох функцій).

Приклад виконання завдання

Відповідно до завдання з роботи 5 потрібно було ввести довільний масив розмірності 10, зробити низку перетворень, знайти суму масиву, його мінімальний елемент і вивести на екран. Причому низка дій (наприклад, виведення на екран) виконувалися кілька разів.

```
' Масив First статичний з 10 елементами від 0 до 9
Dim First(9) As Integer
' Масив Second динамічний
Dim Second() As Integer

Sub InputArray(Arr, Length)
'Заповнення початкового масиву випадковими числами
Randomize 'Запуск лічильника випадкових чисел
For i = 0 To Length - 1
' Привласнити елементу масиву випадкове число від 1 100 із зрушенням на -20
Arr(i)= Int((100) * Rnd - 21)
Next i
End Sub

Sub ArrayToScreen(Arr, Length, Title)
'Заповнити рядок елементами масиву,
'перемежуючи їх пропусками, і вивести на екран початковий масив
For i = 0 To Length - 1
Str = Str & Arr(i)& " "
Next i
MsgBox Str,, Title
End Sub

Function SummaOfArray(Arr, Length)
"Підрахуємо суму елементів початкового масиву
Sum = 0
For i = 0 To 9
'Додаємо до суми кожен елемент початкового масиву
Sum = Sum + Arr(i)
Next i
SummaOfArray = Sum
End Function

Function MinOfArray(Arr, Length)
'Знаходимо мінімальний елемент
'Приймаємо за мінімальний елемент нульовий елемент масиву
Min = Arr(0)
For i = 1 To Length - 1
'Починаючи з першого елемента масиву, перевіряємо, якщо він менший
'мінімального, то записуємо його в мінімальне значення
If Arr(i)< Min Then
Min = Arr(i)
End If
Next i
MinOfArray = Min
End Function

Sub Main()
```

```

' Виконуємо пункт а)
' Заповнення початкового масиву випадковими числами і виведення його на екран
InputArray First, 10
ArrayToScreen First, 10, "Початковий масив"

' Виконуємо пункт b)
' Підрахувати кількість непарних чисел для визначення розмірності масиву Second
For j = 0 To 9
    If (First(j) Mod 2) <> 0 Then
        до = до + 1
    End If
Next j
' Вивести на екран кількість непарних елементів
MsgBox до, "Кількість непарних елементів"

' Визначити розмірність масиву Second елементів від 0 до k-1
ReDim Second(до - 1)
i = 0 'лічильник масиву Second обнулити
For j = 0 To 9
    If (First(j) Mod 2) <> 0 Then
        ' Записати в масив Second непарне число з масиву First
        Second(i) = First(j)
        ' лічильник масиву Second збільшується тільки для непарних чисел масиву First
        i = i + 1
    End If
Next j
' Вивести на екран перетворений масив
ArrayToScreen Second, до, "Перетворений масив"

' Якщо величина масиву Second більше 3, то викинути з нього
' елемент з номером 3
If до > 3 Then
    ' Починаючи з елемента з номером 3 привласнювати йому наступний елемент
    ' цього ж масиву
    ' Щоб не вийти за межі масиву, закінчуємо на передостанньому елементі.
    For i = 3 To до - 2
        Second(i) = Second(i + 1)
    Next i
    ' Оскільки кількість елементів у масиву Second на одиницю зменшилося
    ' змінюємо його розмірність на 1. Ключове слово Preserve не дозволить
    ' обнулити вже існуючі елементи масиву
    ReDim Preserve Second(до - 2)

    ' Вивести на екран стислий масив
    ArrayToScreen Second, до - 1, "Стислий масив"
Else
    ' Якщо в масиві Second менше 4 елементів, він стисненню не підлягає
    MsgBox "Масив не підлягає стисненню, оскільки його розмір менше 4", "Стислий масив"
End If

' Виконуємо пункт c)
MsgBox SummaOfArray(First, Length), "Сума елементів початкового масиву"
MsgBox MinOfArray(Second, Length), "Мінімальний елемент стислого масиву"
End Sub

```

Контрольні питання

1. Що таке процедури і функції та для чого вони призначені?
2. Чим вони відрізняються один від одного?

2. САМОСТІЙНІ РОБОТИ

2.1 Операційна система Microsoft Windows

2.1.1 Самостійна робота № 1

Завдання

Самостійно зробіть налаштування робочого стола Вашого комп'ютера, змінивши шпалери робочого стола, вигляд і налаштування миші, сформувавши свою панель швидкого запуску завдань і ярлики робочого стола. Результат і поетапне налаштування відобразить у зошиті.

2.1.2 Самостійна робота № 2

Завдання

1. У кореневому каталозі диска (A, C, D) створити папку «Самостійна робота Ваше прізвище», наприклад, «Самостійна робота Іванова».
2. Створити малюнок за допомогою додатка Paint – умовно «Ваш автопортрет», зберегти його у файлі під ім'ям «Мій портрет» у папці «Самостійна робота Ваше прізвище».
3. Створити документ WordPad, записати в ньому три абзаци тексту не менш двох рядків кожний:
 - о про себе;
 - о про свої захоплення;
 - о цікаві випадки з Вами.
4. Зберегти цей документ під ім'ям «Про мене» у папці «Самостійна робота Ваше прізвище».
5. Вставити свій автопортрет з документа «Мій портрет» у документ «Про мене» і зберегти його під ім'ям «Про мене з портретом» у папці «Самостійна робота Ваше прізвище».

2.2 Текстовий процесор Microsoft Word

2.2.1 Самостійна робота № 1. Знайомство з текстовим редактором Microsoft Word

Завдання

Відкрийте Microsoft Word за допомогою меню кнопки «Пуск» *Пуск → Програми → Microsoft Word*.

1. Відкрийте по черзі всі команди меню і запишіть їх у зошит.
2. Запишіть у зошит у стовпчик назви кнопок «Стандартної панелі інструментів» і навпроти кожної з них – назву команди меню, яка дублює цю операцію.
3. Перегляньте документ у чотирьох різних режимах (кнопки режимів знаходяться в лівому нижньому кутку екрана).
4. Увімкніть режим показу знаків, які не друкуються (кнопка ¶ на панелі інструментів).
5. Увійдіть у налаштування панелі інструментів *Сервіс → Налаштування → Панелі інструментів*
6. Встановлюючи позначки, перегляньте всі можливі панелі інструментів.
7. Залишіть у робочому вікні тільки панелі «Стандартна» і «Форматування».
8. Зробіть переміщення по тексту усіма вивченим Вами способами.
9. Створіть новий документ шляхом клацання на кнопці «Створити». Зверніть увагу на назву цього документа.
10. Сформуйте текст шляхом копіювання фрагментів тексту самостійної роботи через буфер обміну. Отриманий Вами текст має містити фрагменти тексту в наступному порядку: Тема, Контрольні питання, Способи переміщення по тексту, Панелі інструментів, Способи виділення тексту.
11. Запишіть новий документ на знімний носій під ім'ям «Фрагменти тексту.doc»

2.2.2 Самостійна робота № 2. Форматування документів Microsoft Word

Завдання

1. Відкрийте документ «Фрагменти тексту.doc».
2. Встановіть режим розмітки сторінки (див. Меню Вигляд).
3. За допомогою пункту «Параметри сторінки → Поля» встановіть верхнє поле 2,5 см, а ліве – 3 см. Подивіться, як змінилися границі смуги набору тексту. Відповідь запишіть у зошит.
4. За допомогою лінійок встановіть величину правого поля 2 см, а нижнього – 4 см.
5. Збережіть документ із старим ім'ям (зберегти).

6. За допомогою пункту «Параметри сторінки → Розмір паперу» встановіть замість книжкової - альбомну орієнтацію. Зверніть увагу на те, як при цьому змінилися розміри полів. Відповідь запишіть у зошит.

7. За допомогою пункту «Параметри сторінки → Декілька сторінок → «Дзеркальні поля» подивіться, як змінилися параметри сторінки й назви полів. Відповідь запишіть у зошит.

8. За допомогою кнопки «Скасувати введення» поверніться до книжкової орієнтації.

9. Зберегти документ у каталог «Ваше Прізвище» під ім'ям «Форматування фрагментів тексту.doc».

10. В останньому абзаці встановіть такі параметри: відступ ліворуч – 0, відступ праворуч – 4 см, відступ першого рядка – 0.5 см, відступ попереду – 12 пт, міжрядковий інтервал – подвійний, вирівнювання – по лівому краю.

11. Виділіть перший абзац файлу «Форматування фрагментів тексту.doc» і помістіть його в буфер обміну.

12. Вставте скопійований абзац після зміненого (останнього) з нового рядка.

13. Цей абзац відформатуйте в такий спосіб: вирівнювання – по правому краю, лівий та правий відступи й відступ першого рядка – 0 см, відступ зверху – 6 пт, відступ знизу – 18 пт, міжрядковий інтервал – множник, значення – 0,8.

14. Починаючи з наступного порожнього рядка, напишіть свої плани на майбутній тиждень (наприклад, відвідати лекції, закінчити курсову роботу, піти в театр, купити нову річ та ін). Текст має складатися з декількох (трьох) речень, що створюють абзац. Бажано розписати план кожного дня.

15. Для символів першої речення вашого тексту встановіть наступні параметри шрифту: гарнітура – Bookman Old Style, розмір – 20, накреслення – напівжирний.

16. Для символів другого речення встановити гарнітуру – Courier New, розмір – 10, накреслення – підкреслений курсив, кольори – червоний, усі прописні.

17. Для символів третього речення: гарнітура – Comic Sans MS, розмір – 32, кольори – синій, виділення кольорами – жовтий.

18. Запишіть новий документ на знімний носій під ім'ям «Фрагменти тексту 2.doc»

2.2.3 Самостійна робота № 3. Створення маркованих і нумерованих списків

Завдання

1. Створіть список, що описує ваше місто, що складається з районів, які містять вулиці, на яких розташовані будинки (4-рівневий структури).

2. Для цього списку створіть свій багаторівневий список.

3. Збережіть документ з ім'ям (Багаторівневий список.doc).

4. Замініть стандартний маркер символом, який Вам сподобався.

5. Відкрийте додаток Microsoft Word.

6. Відкрийте файл, створений раніше (Фрагменти тексту2.doc).

7. Виділіть всі абзаци, крім «Контрольні питання» й «Тема» і вимкніть підкреслення.

8. Не знімаючи виділення, встановіть нумерований список.

9. Збережіть файл у вашому каталозі під ім'ям (Нумерований список.doc).

10. Замініть нумерований список маркованим.

11. Зберегти файл під ім'ям (Маркований список.doc).

12. Встановіть курсор в останній маркований абзац, натисніть «Enter» і в абзаці, що з'явився, напишіть висновки (що при цьому відбулося?).

13. Зберегти файл під ім'ям (Маркований список.doc).

2.2.4 Самостійна робота № 4. Прийоми роботи з таблицями.

Вставка символів та буквиці

Завдання

1. Створіть новий документ Word з ім'ям <Ваше ім'я>.doc і наберіть текст, заданий викладачем.
2. Після кожного п'ятого слова введіть символ «Стрілочка вліво», «Авторське право» й «Ключик».
3. Зробіть новий абзац, починаючи із другого речення, і зробіть для нього буквицю.
4. Виконаєте «Вирівнювання по ширині».
5. Виконайте перевірку правопису для набраного вами фрагмента тексту й виправте помилки.
6. Якщо помилок немає, внесіть їх самостійно й подивіться, як реагує Word.
7. Виконайте розміщення переносів (пункт меню «Розміщення переносів...» з меню «Мова»).
8. Між першим і другим абзацами за допомогою кнопки "Додати таблицю" вставте таблицю, що складається з 3 рядків й 5 стовпчиків.
9. За допомогою команди «Границі й заливання» зробіть зовнішню рамку подвійною, а межі між комірками – пунктирними.
10. Скопіюйте цю таблицю в буфер обміну і вставте її перед першим абзацом.
11. У скопійованій таблиці видаліть межі комірок за допомогою команди Автоформат...
12. Сховайте сітку скопійованої таблиці командою «Сховати сітку».
13. Після другого абзацу вставте таблицю з використанням вікна діалогу «Вставка таблиці». Таблиця повинна мати 5 рядків й 6 стовпчиків.
14. Після третього абзацу вставте таблицю за допомогою інструмента "Намалювати таблицю". Таблиця повинна мати 2 рядки й 10 стовпчиків.
15. Змініть останню комірку таблиці за допомогою «ластику» таким чином, щоб залишилося 5 стовпчиків різної ширини.
16. У зошиті опишіть способи переміщення по таблиці.
17. У зошиті опишіть способи виділення елементів таблиці (комірок, рядків, стовпчиків, ділянок).
18. У таблицю розмірністю 5 на 6 внесіть наступні дані:
 - у діапазон осередків B2:F2 внесіть назви предметів

Математика	Фізика	Мова	Інформатика	Фізкультура
------------	--------	------	-------------	-------------

 - у діапазон осередків A3:A5 внесіть імена друзів

Артур
Христина
Денис

 - об'єднайте діапазон B1:F1 в одну комірку і внесіть у неї текст «Найменування предметів»;
 - об'єднайте діапазон A1:A2 в одну комірку і внесіть у неї текст «Імена друзів».
19. Виконайте наступні елементи форматування:
 - о виконайте вирівнювання по центру для заповнених комірок;
 - о текст «Назви предметів» позначте як «Жирний»;
 - о для комірки з текстом «Імена друзів» виконайте вирівнювання по вертикалі (по центру).
20. Заповніть відповідні комірки оцінками.
21. Розфарбуйте текст і тло комірок у різні кольори.
22. Збережіть файл під ім'ям (Таблиця.doc).

2.2.5 Самостійна робота № 5. Прийоми роботи з об'єктами MS Word

Завдання

1. Викличте панель «Малювання» і запишіть у зошит назви кнопок цієї панелі, включаючи всі підменю.
2. Відкрийте файл «Таблиця.doc».
3. Після останньої таблиці намалюйте прямокутник, квадрат і коло.
4. З'єднайте центр кола і квадрата (коло зверху квадрата).

5. Зменшуючи або збільшуючи розміри фігур, зробіть, щоб коло було вписано у квадрат.
6. Перетягніть пряму так, щоб вона перекреслювала обидві фігури.
7. Згрупуйте фігури в одну.
8. Скопіюйте отриману фігуру в початок документа.
9. Збережіть документ під ім'ям «ВАШЕ ім'я1.doc».
10. Для фігури, що перебуває на початку документа, змініть розміщення відносно тексту (за текстом і перед текстом).
11. Розгрупуйте фігури, що перебувають наприкінці тексту.
12. Розфарбуйте їх у різні кольори і застосуйте заливання.
13. Поставте квадрат попереду круга. Що ви спостерігаєте?
14. За допомогою винесень підпишіть кожну фігуру.
15. Для фігури, що знаходиться на початку тексту, задайте параметри за допомогою пункту «Формат автофігури...» з контекстного меню.
16. Наприкінці документу створіть текстовий ефект фрагменту: «Ваше Прізвище Ім'я По батькові».
17. Витягніть на стандартну панель інструментів кнопку запуску редактора формул.
18. Наберіть довільну формулу.
19. Збережіть документ з ім'ям «Таблиця1.doc».

2.2.6 Самостійна робота № 6. Засоби підвищення швидкості роботи з документами MS Word

Завдання

1. Викличте панель інструментів «Автотекст».
2. Перегляньте й законспекуйте кнопки меню «Автотекст».
3. Відкрийте файл, створений минулого разу і збережіть його під ім'ям «Ваше ім'я6.doc».
4. Із цього файлу видаліть усі графічні об'єкти і вставлені символи.
5. Наберіть заданий фрагмент тексту (теоретичний розділ цієї роботи).
6. У набраному фрагменті тексту виберіть два ключових слова, тобто два поняття, які визначають суть цього фрагмента. Запишіть ключові слова в зошит.
7. Для кожного ключового слова створіть елемент автотексту.
8. Скопіюйте через буфер обміну набраний текст і вставте його після набраного тексту.
9. Між першим і другим частинами тексту вставте порожній рядок.
10. У порожній рядок вставте обидва ключові слова одне за іншим.
11. Вставте ключові слова перед першим абзацом.
12. Другому ключовому слову призначте комбінацію клавіш.
13. Перевірте, як вона працює.
14. Одному із ключових слів призначте кнопку на панелі інструментів.
15. Перевірте, як вона працює.
16. Збережіть файл під ім'ям «Ваше ім'я6.doc».

2.2.7 Самостійна робота № 7. Колонтитули, стилі, шаблони в MS Word

Завдання

1. Викличте вікно «Колонтитули» і запишіть у зошит назви кожної із кнопок, використовуючи для цього коротку підказку.
2. Відкрийте документ, створений минулого разу.
3. Створіть верхній колонтитул, який буде містити Ваше прізвище, поточну дату й час.
4. Прикрасьте верхній колонтитул за своїм розсудом.
5. У нижній колонтитул вставте номер сторінки.
6. Збережіть файл під ім'ям «<Ваше ім'я>-Колонтитул.doc».
7. Задайте відстань до верхнього колонтитула (1,5 см), ширина колонтитула (2 см).
8. Зробіть так, щоб кожен елемент верхнього колонтитула починався з нового рядка і текст колонтитулів був вирівняний по центру.

9. Збережіть зміни у файлі.
10. Натисніть кнопку «Стиль» і запишіть у зошит декілька стилів, що сподобалися, з їхньою назвою і параметрами.
11. Скопіюйте останній абзац вашого документа три рази.
12. За допомогою діалогового вікна «Створення стилю» створіть стиль абзацу з наступними параметрами: тип шрифту «Courier New», розмір – 15, кольори – жовтогарячий, вирівнювання по центру, напівжирний курсив, подвійна рамка навколо абзацу, мова російська з перевіркою орфографії.
13. Застосуйте створений вами стиль до першого із скопійованих абзаців.
14. На базі тексту другого скопійованого абзацу створіть стиль, в якому шрифт 8 розміру, червоного кольору із чорними мурахами.
15. Перегляньте всі наявні шаблони за допомогою засобів попереднього перегляду.
16. Назви переглянутих шаблонів запишіть у зошит.
17. Створіть шаблон залікової книжки.
18. Збережіть створений шаблон під своїм ім'ям у каталозі «Шаблони».
19. На підставі створеного вами шаблону створіть документ, що містить заповнену залікову книжку Іванова.
20. Збережіть файл під ім'ям «Ваше ім'я-Колонтитул.doc».

2.3 Power Point

2.3.1 Самостійна робота. Створення презентації

Створіть файл презентації на підставі самостійних робіт з Microsoft Word. Презентація має містити 7 слайдів з написами, малюнками та діаграмами, які ілюструють відповідну роботу, а також перший слайд із назвами усіх робіт і керуючими кнопками, які здійснюють перехід до кожного із 7 слайдів. Використайте в презентації різноманітні ефекти та стилі.

2.4 Основи електронних таблиць EXCEL

2.4.1 Самостійна робота № 1. Знайомство з Excel

Завдання

1. Запустіть Excel, клацнувши на відповідній піктограмі на робочому столі. Вивчіть можливість запуску Excel, використовуючи *Пуск → Програми → Excel*.
2. Вивчіть основні елементи вікна Excel, яке відкрилося. Укажіть на рядок заголовка, рядок меню, стандартну панель інструментів, панель інструментів форматування, вікно адреси комірки, рядок формул, смуги прокручування аркуша, рядок стану, панель задач.
3. Вивчіть рядок меню Excel. Законспекуйте вкладені меню кожної команди рядка меню. Зверніть увагу на активні й пасивні команди.
4. Вивчіть порожнє вікно, що відкрилося, книги. Зверніть увагу на основні об'єкти, з якими працює Microsoft Excel:
 - Робоча книга
 - Аркуш
 - Рядок
 - Столпчик
 - Комірка
 - Діапазон
 - Список.
5. Перейдіть на аркуш 2, використовуючи ярлички аркушів. Поверніться на Аркуш 1.
6. Змініть розмір аркуша, скориставшись кнопкою "Відновити".
7. Згорніть вікно книги, скориставшись кнопкою "Згорнути".
8. Розгорніть вікно книги до максимального розміру.
9. Послідовно зробіть активними комірки A1, A19, K19, K1, A1.
10. Установіть покажчик на будь-якій комірці та клацніть правою клавішею миші. Вивчіть і законспекуйте контекстне меню, яке відкрилося.

11. Установіть покажчик на будь-якій панелі інструментів і клацніть правою клавішею миші. Вивчіть й законспекуйте контекстне меню, яке відкрилося.
12. Вивчіть панель інструментів. Законспекуйте найбільше часто використовувані команди стандартної панелі інструментів.
13. Перемістіть стандартну панель інструментів до правого краю екрана, потягнувши мишею за вертикальну риску на початку панелі. Зробіть панель «висячою».
14. Виберіть меню **Вигляд** → **Панелі інструментів** → **Настроювання** й установіть на вкладці **Панелі інструментів** прапорці біля **Стандартна, Форматування та Границя**. Розташуйте панель **Границя**, яка з'явилася, у зручному для Вас місці екрана.
15. Виберіть вкладку **Параметри** відкритого вікна настроювання й законспекуйте всі вкладки цього вікна.
16. Видаліть кнопку **Вирізати** з панелі інструментів **Стандартна**. Виберіть меню **Вид** → **Панелі інструментів** → **Настроювання**. Виділіть на панелі інструментів кнопку **Вирізати** й перетягніть її за межі панелі.
17. Додайте кнопку **Вирізати** на панель інструментів (**Вид** → **Панелі інструментів** → **Настроювання** → **Команди** → **Виправлення**, вибрати кнопку й перетягнути її на панель інструментів; або за допомогою контекстного меню на панелі інструментів).
18. Додайте на панель інструментів кнопку **Очистити**.
19. Клацніть на кнопці **Довідка** рядка меню. Вивчіть вікно, що відкрилося, довідки. Отримайте довідку з уведення формул у таблицю (викличте помічника й уведіть у вікно для пошуку Формула, потім з переліку пропонуєваних посилань виберіть Створення формули).
20. Скопіюйте отриману довідку через буфер обміну у свій каталог у файл із ім'ям Довідка.doc. Закрийте вікно довідки.
21. Клацніть правою клавішею миші на Помічнику й виберіть **Параметри**. Установіть прапорці біля всіх опцій, крім «Відображувати тільки важливі поради». Клацніть на ОК. Спробуйте змінити помічника, вибравши іншого на вкладці **Колекція**.
22. Клацніть правою клавішею миші на **Помічнику** й виберіть **Мотор!** Відобразіть результат у зошиті.
23. За своїм варіантом виберіть предметну галузь і складіть у зошиті «Звіт продажів» і «Список товарів», аналогічних наведеним нижче.

Приклад «Звіту продажів» для предметної галузі «Комп'ютерна фірма»

	Фірма K&S					
		Січень	Лютий	Березень	Квітень	Травень
Статті доходу						
Комп'ютери		34890	45750	43780	51250	65740
Комплектуючі		10980	12550	15670	16780	11570
Периферія		13450	11650	10550	4870	4550
Дохід усього						
Витрати						
Реклама		750	1200	1450	250	900
Оренда		1000	1000	1000	1000	1000
Податки		2000	1500	2500	3000	5000
Загальні витрати						
Прибуток						

Приклад «Списку товарів» для предметної галузі «Молочний ларьок»

категорія	товар	модель	ціна (гот.)	кількість	дата надходження
йогурт	ананасовий	виолла	10.7	50	30.04.2010
йогурт	ананасовий	ромол	9.24	100	12.05.2010
йогурт	полуничний	старий майстер	52.8	20	24.04.2010
йогурт	полуничний	едем	38.3	30	24.04.2010
молоко	козяче	чумак	2.45	240	28.04.2010
майонез	маслиновий	фані	3.45	150	08.05.2010
йогурт	персиковий	королівський	24	32	01.05.2010
сир	Плавлений	ромол	1.25	30	12.05.2010
майонез	провансаль	заріччя	8.65	90	10.05.2010
масло	вершкове	фані	2.55	150	08.05.2010
сир	твердий	заріччя	2.3	100	10.05.2010
сир	твердий	ромол	2.1	50	12.05.2010
сир	твердий	фані	1.8	60	08.05.2010
молоко	пряжене	хжк	2.45	200	28.04.2010
масло	шоколадне	ромол	2.55	100	12.05.2010

Варіанти завдань

Варіант	Предметна галузь	Варіант	Предметна галузь
1	Молочна крамниця	16	Кондитерська крамниця
2	Хлібний магазин.	17	Вино-горілчаний магазин
3	Магазин «Соки – води»	18	Магазин «Овочі – фрукти»
4	Склад-магазин пило-матеріалів	19	Ковбасний магазин
5	Тютюнова крамниця	20	Книгарня
6	Аптечна крамниця	21	Іграшковий магазин
7	Магазин канцтоварів	22	Взуттєвий магазин
8	Ювелірний магазин	23	УМС-крамниця
9	Магазин оргтехніки	24	Господарський магазин
10	Крамниця із продажу побутової хімії	25	Магазин побутової техніки
11	Магазин верхнього одягу	26	Меблевий магазин
12	Автомагазин	27	Квітковий магазин
13	Магазин «Годинники»	28	Магазин із продажу шпалер
14	Магазин «Килими»	29	Магазин освітлювальних приладів
15	Магазин із продажу музичних інструментів	30	Косметична крамниця

Контрольні запитання

1. У чому складається основне призначення електронних таблиць?
2. Яким іншим способом можна запустити Microsoft Excel, якщо на робочому столі немає піктограми Excel?
3. На панелі інструментів відсутні деякі знайомі кнопки та присутні непотрібні. Як повернути звичний стан панелі?
4. Що відбудеться при натисканні кнопки **Скидання меню** налаштування панелі інструментів?
5. Як можна збільшити частину документа, яку ви бачите?
6. Назвіть основні об'єкти Microsoft Excel.
7. Яке розширення мають файли, створені в Microsoft Excel?
8. Що таке робоча книга?
9. Які типи аркушів Вам відомі?
10. Що таке робочий аркуш, із чого він складається?

11. Які позначки мають рядки, стовпчики таблиці?
12. Що являє собою комірка таблиці, яка інформація в ній може міститися?
13. Які константи може містити комірка?
14. Що означає адреса комірки?
15. Які стилі адресації використовуються в Microsoft Excel?
16. Як змінити стиль адресації комірок?
17. З яких елементів складається рядок формул?
18. Поясніть поняття: поточна (активна) робоча книга, що таке аркуш, що таке комірка?
19. Назвіть способи навігації по аркушах книги.
20. Як відобразити (сховати) ярлички аркушів?
21. Назвіть способи переміщення по робочому аркуші.
22. Які комбінації клавіш дозволяють переміщуватися по аркуші за допомогою клавіатури?
23. Як викликати Помічника, як можна змінити його вигляд?
24. Як активізувати неактивні команди підменю рядка меню?
25. Назвіть режими, у яких може перебувати Microsoft Excel?
26. Як активізувати закриту робочу книгу?
27. Якими способами може здійснюватися перехід між відкритими робочими книгами?
28. Як створити новий документ Microsoft Excel?
29. Як зберегти поточний документ?
30. Назвіть способи завершення роботи Microsoft Excel.

2.4.2 Самостійна робота № 2. Створення електронних таблиць.

Введення і редагування даних

Завдання

1. Запустіть Excel.
 2. Зробіть активною комірку B1. Уведіть текст *Продажі*.
 3. Зробіть активною комірку A5. Уведіть текст *Виконавець*.
 4. Зробіть активною комірку A6. Уведіть текст *Дата*, в C6 – поточну дату = *сьогодні()*.
- Зверніть увагу на те, що *сьогодні()* це функція, результатом якої є поточна дата.
5. Зробіть активною комірку C5. Уведіть Ваше прізвище, ім'я та по батькові.
 6. Збережіть набраний документ під ім'ям FIO-1.xls у Вашому каталозі.
 7. Відредагуйте текст у комірці B1, написавши *Звіт продажів*.
 8. Зробіть активною комірку C8. Уведіть текст *Січень*.
 9. Заповніть комірки D8-G8 за допомогою автозаповнення:
 - зробіть активною комірку C8;
 - установіть покажчик на маркер заповнення (у правому нижньому куті комірки);
 - натисніть ліву клавішу миші й, не відпускаючи, переміщуйте покажчик до комірки G8, де відпустите клавішу.
 10. В комірки A9, A10 уведіть відповідно 0 й 1.
 11. За допомогою автозаповнення заповніть діапазон A11-A20. Проаналізуйте отриманий результат.
 12. Очистіть діапазон A9-A20.
 13. У комірку B3 уведіть назву Вашої фірми.
 14. Відповідно до свого варіанта створіть таблицю «Звіт продажів», аналогічну наданій:
 - виділіть діапазон A9-H19, нажавши клавішу **Shift** і ліву клавішу миші на комірці A9 та, утримуючи їх, обведіть діапазон до комірки H19;
 - комірки C10-G12, C15-G17 заповніть довільними числами;
 - комірка A9 – *Статті доходу*;
 - комірка A13 – *Доход усього*;
 - комірка A14 – *Статті витрат*;
 - комірка A18 – *Загальні витрати*;
 - комірка A19 – *Прибуток*.

	A	B	C	D	E	F	G	H
1		Звіт продажів						
2								
3		Фірма K&S						
4								
5	Виконавець		Іванов І. І.					
6	Дата		02.11.01					
7								
8			Січень	Лютий	Березень	Квітень	Травень	
9	Статті доходу							
10	Комп'ютери		34890	45750	43780	51250	65740	
11	Комплектуючі		10980	12550	15670	16780	11570	
12	Периферія		13450	11650	10550	4870	4550	
13	Дохід усього							
14	Статті витрати							
15	Реклама		750	1200	1450	250	900	
16	Оренда		1000	1000	1000	1000	1000	
17	Податки		2000	1500	2500	3000	5000	
18	Загальні витрати							
19	Прибуток							

15. Збережіть набраний документ під ім'ям ФІО-2.xls (наприклад, Іванов -2.xls).

16. За допомогою автосуми введіть в H10 суму за 5 місяців:

- виділіть комірку H10;
- на стандартній панелі інструментів клацніть на кнопці *Автосума*;
- переконайтеся, що діапазон обраний правильно й натисніть *Enter*;
- якщо діапазон обраний неправильно, помістіть покажчик у комірку B10 й, не відпускаючи ліву клавішу, обведіть до G10, натисніть *Enter*.

17. Аналогічно п. 16 заповніть комірки H11:H12, H15:H17, C13:H13, C18:H18.

18. Знайдіть прибуток як різницю між доходом і витратою. Активізуйте комірку C19 й уведіть формулу C13-C18:

- уведіть знак «=»;
- клацніть на комірці C13;
- уведіть знак «-»;
- клацніть на комірці C18.

19. Альтернативний варіант – скористайтесь рядком формул і кнопками на ній.

20. Аналогічно п. 18 заповніть комірки D19-H19.

21. Наприкінці Ви маєте одержати документ, аналогічний наведеному нижче. Збережіть його.

	A	B	C	D	E	F	G	H
1		Звіт продажів						
2								
3		Фірма K&S						
4								
5	Виконавець		Іванов І. І.					
6	Дата		02.11.01					
7								
8			Січень	Лютий	Березень	Квітень	Травень	
9	Статті доходу							
10	Комп'ютери		34890	45750	43780	51250	65740	241410
11	Комплектуючі		10980	12550	15670	16780	11570	67550
12	Периферія		13450	11650	10550	4870	4550	45070
13	Дохід усього		59320	69950	70000	72900	81860	354030
14	Статті витрати							
15	Реклама		750	1200	1450	250	900	4550
16	Оренда		1000	1000	1000	1000	1000	5000
17	Податки		2000	1500	2500	3000	5000	14000
18	Загальні витрати		3750	3700	4950	4250	6900	23550
19	Прибуток		55570	66250	65050	68650	74960	330480

Контрольні запитання

1. Якими способами можна ввести дані в таблицю?
2. Якими способами можна здійснити підтвердження введення даних в комірку?
3. У чому відмінність введення даних у текстовому редакторі й табличному процесорі, зокрема в Microsoft Excel?
4. Які типи даних використовуються в Microsoft Excel?
5. Що таке діапазон комірок, як він позначається?
6. Чим відрізняється автозавершення й автозаповнення?
7. При автозавершенні й автозаповненні можуть брати участь значення, які перебувають не в суміжних комірках, чи ні?
8. Як здійснити вибір даних зі списку при введенні текстових даних?
9. Якими способами можна виконувати автозаповнення?
10. Що таке маркер заповнення, у яких випадках він використовується?
11. Що таке список автозаповнення?
12. Як створити новий список автозаповнення?
13. Як за допомогою автозаповнення ввести в комірки B19-G19 числа, починаючи з 0 і кратні 2?
14. Назвіть основні складові формули.
15. Що є головною особливістю при введенні формул?
16. Як відрізнити текст від формули?
17. Які види операторів можуть застосовуватися при побудові формул?
18. Для чого використовується оператор конкатенації?
19. Як позначаються операції об'єднання, перетинання діапазонів комірок?
20. Як увести формулу за допомогою рядка формул?
21. Яку функцію у формулі виконують круглі дужки?
22. Як одержати суму діапазону комірок?
23. Які вбудовані функції Excel вам відомі, як їх викликати?
24. Як використовується вікно побудови виражень при виклику функції?
25. Назвіть основні категорії вбудованих функцій Excel.
26. Що означають зовнішні посилання?
27. Що означають вилучені посилання?
28. Для чого використовуються тривимірні посилання?
29. Як позначається посилання у відкритій робочій книзі й у закритій?
30. Як можна відредагувати вміст комірки?

2.4.3 Самостійна робота № 3. Редагування таблиць. Формування робочих аркушів

Завдання

1. Запустіть *Excel* і відкрийте файл ФІО-2.xls.
2. Збільшіть ширину стовпчика А так, щоб у ньому містився весь текст:
 - установіть покажчик на правій межі заголовка стовпчика А (покажчик прийме вигляд стрілки «↔»);
 - натиснувши ліву клавішу миші, потягніть межу праворуч до потрібного розміру й відпустіть клавішу;
 - спробуйте змінити ширину стовпчика Н, скориставшись контекстним меню, викликаним на заголовку стовпчика.
3. Форматуйте рядок 8, виставивши назву місяців по центру комірок, наприклад:
 - викличте контекстне меню на заголовку 8-го рядка;
 - виберіть *Формат комірок*;
 - перейдіть на вкладку *Вирівнювання*;
 - у вікні *По горизонталі* виберіть *По центру* й клацніть *ОК*.
4. У комірку E5 уведіть текст "*Курс грн/\$*", а в комірку F5 – формулу, що відображує поточний курс (наприклад, =1/7,96).

5. У комірку I9 уведіть текст "У доларах", а в комірку I10 – формулу, що перераховує дані в долари за допомогою абсолютної адресації – $H10*\$F\5 ; у комірку J9 – текст "У середньому за 5 місяців в \$", а в комірку J10 – формулу для обчислення середнього; у комірку DО9 – «У середньому за 5 місяців у грн», у клітинку DО10 – відповідну формулу, скориставшись майстром формул: *Вставка* → *Формула*, у вікні формул вибрати СРЗНАЧ, виділити діапазон 310:G10, *ОК*.

6. Скопіювати вміст комірки I10 в комірку I11:I19:

- виділити комірку I10;
- установити покажчик у нижньому правому куті виділеної комірки й, натиснувши ліву клавішу миші, окреслити діапазон для копіювання, після чого відпустити клавішу.

7. Перегляньте вміст комірок I11:I19 і висновки запишіть.

8. Скопіюйте діапазон I10:I19 у діапазон DО10:DО19.

9. Установіть у стовпчиках I, J відповідний формат комірок – знак \$ перед значенням:

- виділіть діапазон I10:I19;
- виберіть *Формат комірок*;
- у вікні числових форматів виберіть *Грошовий*;
- установіть 2 відображуваних знаки після коми;
- у списку *позначення* виберіть значок \$;
- клацніть *ОК*.

10. Скопіюйте формат стовпчика I у стовпчик J, скориставшись кнопкою *Формат за зразком* на стандартній панелі інструментів.

11. Скопіюйте комірку J10 у діапазон J11:J19. Чому Ви одержали дані в такому поданні?

12. Форматуйте комірку з датою (36):

- використовуючи *Формат комірок* у списку числових форматів виберіть пункт *Дата*,
- у списку *Тип* виберіть подання дати у вигляді «1 грудня, 2010»;
- завершіть натисканням *ОК*.

Якщо дата не відображується повністю, збільшіть ширину стовпчика.

13. Змініте шрифт і кольори заголовка таблиці:

- виділіть комірку B1;
- виберіть *Формат осередків*;
- перейдіть на вкладку *Шрифт*;
- виберіть у відповідних вікнах *Times New Roman*, накреслення – напівжирний, розмір – 14, кольори – синій;
- перегляньте зразок і клацніть *ОК*.

14. Змініть шрифт у комірці, яка містить Ваше прізвище: виберіть *Century Gothic*, накреслення – напівжирний курсив, розмір – 12, кольори – червоний.

15. Виділіть рамкою інформацію про виконавця й дату:

- виділіть групу комірок A5-36;
- у контекстному меню виберіть *Формат комірок*;
- відкрийте вкладку *Границя*;
- виберіть тип *Зовнішні*, тип лінії – жирну одинарну;
- клацніть *ОК*.

16. Об'єднайте групу комірок B1:J1 в одну:

- виділіть групу комірок B1-J1;
- виберіть у меню *Формат*;
- відкрийте вкладку *Вирівнювання* й у графі *Відображення* ввімкніть опцію *Об'єднання комірок*;
- у вікні *По горизонталі* встановіть *По центру*;
- у вікні *По вертикалі* встановіть *По центру*;
- клацніть *ОК*.

Альтернативний варіант – виділіть групу комірок B1-J1, на стандартній панелі інструментів клацніть на кнопці *Об'єднати й помістити в центрі*; ще один варіант – скористатися контекстним меню.

17. Обведіть об'єднану комірку подвійною жирною рамкою.

18. Збільшіть висоту першого рядка в 2 рази:

- виберіть у меню *Формат* → *Рядок* → *Висота*;
- збільшіть число у вікні в 2 рази й клацніть *ОК*.

19. Збільшіть висоту 19-го рядка («Прибуток»):

- установіть покажчик на нижній границі заголовка;
- натиснувши ліву клавішу миші, потягніть границю вниз до потрібної висоти;
- відпустіть клавішу.

20. Збільшіть висоту рядків «Дохід усього», «Загальні витрати».

21. Поєднуючи різні групи комірок, міняючи формати комірок і виділяючи їх рамками, приведіть таблицю до вигляду, аналогічного наведеному вище.

22. Збережіть документ під ім'ям ФІО-3.xls і вийдіть із *Excel 2000*.

Контрольні запитання

1. Якими засобами можна виділити комірку?
2. Яким чином здійснити виділення із групи комірок з різним вмістом, наприклад, тільки формул?
3. Назвіть способи виділення декількох рядків (стовпчиків), одного рядка (стовпчика).
4. Як можна виділити кілька несуміжних стовпчиків?
5. Як змінити ширину стовпчика (укажіть кілька способів)?
6. Як змінити висоту рядка (укажіть кілька способів)?
7. Як видалити рядок або стовпчик?
8. Що відбудеться зі стовпчиком даних D, якщо стовпчик C буде вилучений?
9. Як додати рядок, стовпчик?
10. Як виконати приховання стовпчика (рядка)?
11. Як відобразити стовпчик, якщо він схований?
12. Як виділити групу несуміжних робочих аркушів?
13. Як виділити декілька аркушів, що ідуть підряд?
14. Чи можна виділити кілька відкритих робочих книг?
15. Чим відрізняється операція очищення комірок від видалення?
16. Як здійснити очищення комірок?
17. Як виконати видалення комірок?
18. Які числові формати використовуються в Excel?
19. Чи можна змінити шрифт і його накреслення у всій таблиці? Як це зробити?
20. Як виконується обрамлення комірок таблиці?
21. Які існують способи копіювання комірок?
22. Якими способами можна виконати переміщення комірок, у чому відмінність переміщення від копіювання?
23. У чому переваги копіювання комірок через буфер обміну?
24. При виконанні яких операцій і для чого користуються командою Спеціальна вставка?
25. Назвіть послідовність дій при копіюванні комірок з умовою, так щоб у результуючий об'єкт були записані не формули, що отримуються у вихідному об'єкті, а значення, обчислені за цими формулами.
26. Що таке відносна адресація?
27. Що таке абсолютна адресація?
28. Що означає змішана адресація?
29. Яку адресу потрібно ввести в комірку, якщо номер рядка змінюватися при копіюванні не повинен, а ім'я стовпчика може змінюватися?
30. Як виконати об'єднання декількох комірок в одну?

2.4.4 Самостійна робота № 4. Створення й налаштування діаграм в Excel

Завдання

1. Запустіть *Excel* і відкрийте Ваш файл *ФІО-5.xls*.


Побудова гістограми «Об'єми продажів комп'ютерів за місяцями»

2. Виділіть на аркуші *Звіт продажів* дані для побудови діаграми (наприклад, доходи по комп'ютерах, периферії й комплектуючих за 5 місяців, 310:G12).

3. Виберіть команду меню *Вставка* → *Діаграма* або клацніть *Майстер діаграм* стандартної панелі інструментів. На екрані з'явиться діалогове вікно Майстра діаграм.

4. У вікні, що відкрилося (крок 1 з 4), виберіть тип діаграми *Гістограма* й натисніть кнопку *Перегляд результату*. У вікні *Вигляд* виберіть вигляд діаграми «Звичайна гістограма».

5. Клацніть на кнопці *Далі*, щоб відкрити друге вікно *Майстра діаграм*.

6. Розкрийте вкладку *Ряд* й уведіть наступні дані. У списку рядів виберіть мишею *Ряд1*. У графі *Ім'я* (за допомогою кнопки ) уведіть назву ряду «Комп'ютери», комірка A10. Для *Ряд2* і *Ряд3* відповідно введіть *Комплектуючі* й *периферія*. Потім у графу «*Підписи осі X*» уведіть діапазон, що відповідає п'ятьом місяцям, 38:G8. Ви маєте одержати діаграму, подібну наведеній нижче.

7. Клацніть на кнопці *Далі*, щоб відкрити третє вікно *Майстра діаграм* (крок 3 з 4) – *Параметри діаграми*. По черзі відкриваючи всі шість вкладок, внесіть необхідні зміни: задайте назву діаграми, визначіть необхідні осі та задайте їхні назви, уведіть додаткові лінії сітки або зніміть їх взагалі, уведіть у поле діаграми підписи даних або таблицю даних. На полі праворуч перегляньте внесені зміни.

8. Клацніть на кнопці *Далі*, щоб відкрити третє вікно *Майстра діаграм* (крок 4 з 4) – *Розміщення діаграми*. У відповідному вікні встановіть перемикач і розмістіть діаграму на тому ж аркуші.

9. Клацніть на кнопці *Готово*, щоб завершити роботу над діаграмою. Діаграма має бути розміщена на поточному аркуші, затуляючи Вашу таблицю.

10. Помістивши курсор на порожньому місці всередині діаграми, потягніть її мишею вниз і розмістіть під Вашою таблицею.

11. Клацніть на порожньому місці усередині діаграми. Потягнувши за маркери-квадратики, що з'явилися, додайте діаграмі найбільш зручний розмір.

12. Додайте на панель інструментів панель *Діаграми*. Вивчіть і законспектуйте кнопки панелі *Діаграми*.

Побудова кругової діаграми по рекламі за 5 місяців

13. Викличте майстра побудови діаграм.

14. Задайте тип діаграми *Кругова*.

15. Перейдіть на другий крок і виберіть вкладку *Ряд*.

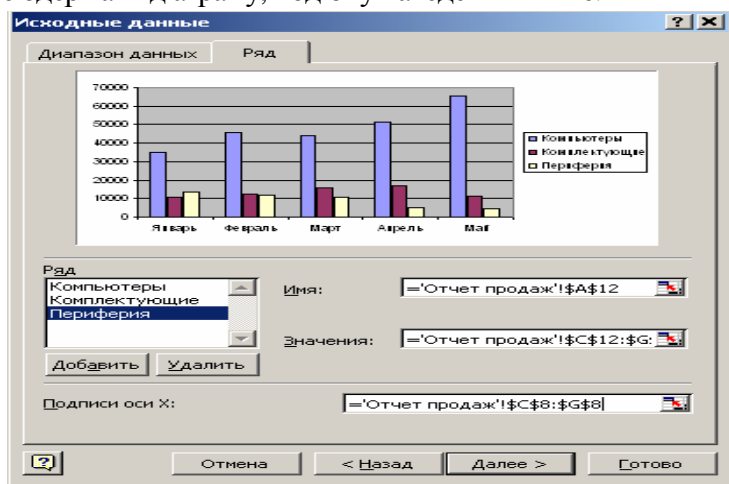
16. Заповніть це вікно: як ім'я ряду має бути *Реклама*; значеннями має бути рядок витрат на рекламу за 5 місяців; підписи до категорій заповнюються назвами місяців.

17. На сектори діаграми внесіть назви місяців, вставляючи усередину поля діаграми рамки з текстом.

18. Внесіть заголовок діаграми, наприклад, «Реклама». Двічі клацнувши по заголовку, змініть шрифт на напівжирний курсив.

19. По черзі виділяючи поле діаграми й кожний із секторів, змініть кольори отриманої діаграми.

20. Розташуйте кругову діаграму праворуч від гістограми.



Зміна діаграми

21. Змініть назву гістограми на «Дохід від продажу комп'ютерів»:
 - виділіть першу діаграму;
 - виберіть команду *Діаграма* → *Параметри діаграми*;
 - розкрийте вкладку *Заголовки* й уведіть нову назву діаграми в поле *Назва діаграми*;
 - клацніть *ОК*.
22. Спробуйте змінити назву осей діаграми, виконавши дії, аналогічні попередньому пункту.
23. Видаліть із гістограми ряд «Периферія».
24. Додайте в гістограму дані за останньою статтею Ваших доходів:
 - виділіть діапазон з даними (у нашому випадку – 312:G12);
 - захопивши його мишею, потягніть на область першої діаграми й відпустіть – з'являться 5 стовпчиків інших кольорів, що відповідають останній статті доходів.
25. Додайте в першу діаграму дані по третій статті Ваших доходів.
26. Змініть тип гістограми на об'ємний:
 - установіть покажчик миші на області діаграми й викличте контекстне меню;
 - виберіть опцію *Тип діаграми*;
 - у діалоговому вікні, що відкрилося, змініть тип на об'ємний;
 - клацніть на *ОК*.
27. Відформатуйте об'ємну діаграму, додавши їй гарного вигляду:
 - установіть покажчик миші на області діаграми й викличте контекстне меню;
 - у діалоговому вікні, що відкрилося, *Формат тривимірної* проекції змініть значення в полях *Піднесення*, *Поворот за допомогою кнопок зі стрілками*;
 - поекспериментуйте із прапорцями *Ізометрія* й *Автомасштаб*, а також із поворотними стрілками;
 - підібравши бажаний вигляд, клацніть на кнопці *Застосувати*;
 - клацніть на *ОК*.
28. На підставі даних про витрати по категоріях за 3 місяці (лютий, березень, квітень), на окремому аркуші побудуйте об'ємну діаграму, вибравши в якості стовпчиків циліндри. Внесіть у нову діаграму заголовок, підписи осей і мітки значень.
29. Відформатуйте елементи об'ємної діаграми:
 - виберіть на області елемента з контекстного меню *Формат рядів даних*;
 - у діалоговому вікні, що відкрилося, *Формат ряду* даних відкрийте вкладку *Фігура*;
 - виберіть фігуру, що сподобалася Вам;
 - клацніть на *ОК*.
30. Збережіть Ваш документ під ім'ям ФІО-6.xls, вийдіть із *Excel*.

Контрольні запитання

1. З якою метою застосовується створення діаграм?
2. Якими способами можна створити діаграму?
3. Де можна розмістити створену діаграму?
4. Чи можна змінити розміщення діаграми після її створення?
5. Які типи діаграм вам відомі?
6. Чи можна в побудованій діаграмі змінити її тип?
7. Як видалити дані з побудованої діаграми?
8. Що таке ряд даних у діаграмі?
9. Як додати у створену діаграму новий ряд даних?
10. Як видалити ряд даних з діаграми?
11. Де відображуються імена рядів даних після побудови діаграми?
12. Як можна змінити порядок даних на побудованій діаграмі?
13. Що таке легенда діаграми?
14. Як змінити текст легенди?

15. Як додати підписи значень у діаграму?
16. Які ви знаєте способи зміни підписів діаграми?
17. Що таке маркери даних, де вони відображаються?
18. Як додати в діаграму таблицю даних?
19. Якими способами можна відредагувати заголовки діаграми та її осей?
20. Як видалити з області побудови діаграми сітку, додати її?
21. Що таке лінія тренда, як її відобразити на діаграмі?
22. Як вирівняти текст назви осі діаграми знизу нагору?
23. Як можна змінити ціну ділення осі значень?
24. Як змінити кольори й вигляд елементів діаграми?
25. Як перемістити діаграму з аркуша з даними на окремий аркуш?
26. Як внести в діаграму прізвище автора?
27. Як змінити вигляд об'ємної діаграми?
28. Яким чином можна змінити розмір створеної діаграми?
29. Як створити свій власний формат діаграми?
30. Чи можна створити нову діаграму без використання майстра діаграм?

2.5 Мова програмування VBA – Visual Basic for Applications

2.5.1 Розгалужені обчислювальні процеси

Складові логічні вирази

Завдання 1: Дано 3 числа А, В, С. Визначити та надрукувати:

1. Взаємно-зворотні числа.
2. Рівні числа.
3. Супротивні числа.
4. Числа, добуток двох з яких дорівнює третьому.
5. Числа, сума двох з яких дорівнює третьому.
6. Числа, різниця двох з яких дорівнює третьому.
7. Від'ємні числа.
8. Додатні числа.
9. Числа, що кратні двом.
10. Числа, що кратні трьом.
11. Парні числа.
12. Непарні числа.
13. Кількість від'ємних чисел.
14. Кількість додатних чисел.
15. Кількість парних чисел.
16. Кількість рівних чисел.
17. Кількість чисел > 1 .
18. Кількість чисел, розташованих в діапазоні $(-2, +15)$.
19. Кількість чисел < 1 .
20. Кількість чисел $= 0$.
21. Числа, що кратні п'яти.
22. Чи можна з відрізків А, В, С побудувати трикутник?
23. Чи складають вони арифметичну прогресію?
24. Чи виконується умова $A > B > C$?
25. Чи виконується умова $A < B < C$?

2.5.2 Циклічні обчислювальні процеси

2.5.2.1 Регулярні цикли

Обчислення кінцевих сум та добутків

Завдання

1. Дано натуральне число N і дійсне число x. Обчислити	$S = \sum_{k=1}^N \left(k^3 * \sum_{i=1}^k (k - i^2) \right).$
2. Дано натуральне число N і дійсне число f. Обчислити	$P = \prod_{i=1}^N \frac{1}{\sum_{k=0}^i (f + k)}.$
3. Дано натуральне число N. Обчислити	$S = \sum_{k=1}^N (-1)^{k-1} * \prod_{m=1}^{2k} \cos \frac{m+1}{2k}.$
4. Дано натуральне число N. Обчислити	$S = \sum_{k=1}^N k * (k+1) * \dots * (k+k).$
5. Дано натуральне число N. Обчислити	$S = \sum_{k=1}^N \frac{\sum_{i=1}^N \sin(0.01 * k * i)}{k!}.$
6. Дано натуральне число N. Обчислити	$S = \sum_{k=1}^N \left(k^3 * \sum_{i=1}^k (k - i^2) \right).$
7. Дано натуральне число N. Обчислити	$S = \sum_{i=1}^N \prod_{j=1}^i \frac{j!}{i!}.$
8. Дано натуральне число N та дійсне число x. Обчислити	$S = \sum_{i=1}^N \prod_{k=1}^N \frac{i+x}{k}.$
9. Дано натуральне число N та дійсне число x. Обчислити	$S = \sum_{i=1}^N \frac{x}{(N+i)!}.$
10. Дано натуральне число N та дійсне число x. Обчислити	$S = \sum_{i=1}^N (-1)^{i+1} \frac{x^{2i}}{i!}.$
11. Дано натуральне число N. Обчислити	$S = \sum_{k=1}^N \prod_{m=1}^{2k} \sin \frac{m\pi}{2k+1}.$
12. Дано l, n – будь-які числа. Обчислити	$S = \prod_{a=1}^5 \left(\frac{\sum_{i=1}^l \prod_{j=1}^n (i+j)^a}{a!} \right).$
13. Дано натуральне число N. Обчислити	$D = \sum_{k=1}^m \sin kx + \prod_{i=1}^m \left(\cos x + \sum_{j=1}^n \frac{1}{i + \sin^2 jx} \right).$
14. Дано натуральне число N. Обчислити	$F = \prod_{i=1}^{12} \frac{\sum_{j=1}^i \sum_{k=1}^j (k=i)^2}{(i+5)!}.$
15. Обчислити	$T = \frac{\prod_{i=1}^{30} \sum_{j=1}^l (b+c)^{j+i}}{(m+l)!-l!}.$

Продовження табл.

16. Обчислити	$F = \frac{\sum_{i=1}^{10} \prod_{j=1}^3 (a+b)^{i=j}}{a!}.$
17. Обчислити	$S = \prod_{c=1}^5 \left(\frac{\sum_{i=1}^m \prod_{j=1}^n (i+j)^c}{c!} \right).$
18. Обчислити	$F = \frac{\prod_{i=1}^{30} \sum_{j=1}^n (b+c)^{i+j}}{(m+n)!-n!}.$
19. Дано натуральне число N і дійсне число x. Обчислити	$S = \sum_{i=0}^n \frac{x^{2i+1}}{(2i+1)!}.$
20. Дано натуральне число N і дійсне число x. Обчислити	$S = \sum_{i=0}^n \frac{x^{2i}}{(2i)!}.$
21. Дано натуральне число N і дійсне число x. Обчислити	$S = \sum_{i=0}^n (-1)^i \frac{x^{2i+1}}{(2i+1)!}.$
22. Дано натуральне число N і дійсне число x. Обчислити	$S = \sum_{i=0}^n (-1)^i \frac{x^{2i}}{(2i)!}.$
23. Дано натуральне число N і дійсне число x. Обчислити	$S = \sum_{i=0}^n \frac{(x-1)^{2i+1}}{(2i+1)(x+1)^{2i+1}}.$
24. Дано натуральне число N і дійсне число x. Обчислити	$S = \sum_{i=0}^n \frac{(x+1)^{2i}}{2i(x-1)^{2i}}.$
25. Дано натуральне число N і дійсне число x. Обчислити	$S = \sum_{i=1}^n \frac{(x+1)^{2i-1}}{(2i-1)(x-1)^{2i-1}}.$

2.5.2.2 Ітераційні цикли

Завдання 1. Обчислити із заданою точністю EPS

1. Обчислити $\ln(1+x)$ за наближеною формулою:

$$\ln(1+x) = x - \frac{x^2}{2} + \frac{x^3}{3} - \dots + (-1)^{n-1} \frac{x^n}{n}.$$

2. Обчислити інтегральну функцію $\Phi(x)$, використовуючи наближену формулу ($x>0$):

$$\Phi(x) = x \sqrt{\frac{2}{\pi}} \left(1 - \frac{(x/2)^2}{1!3} + \frac{(x/2)^4}{2!5} - \frac{(x/2)^6}{3!7} + \dots \right)$$

3. Обчислити $\ln 2$ за наближеною формулою:

$$\ln 2 = 1 - \frac{1}{2} + \frac{1}{3} - \frac{1}{4} + \dots \pm \frac{1}{n}.$$

4. Обчислити $\sin x$ за рекурентною формулою (у радіанах):

$$U_{k+1} = -U_k \frac{x^2}{2k(2k+1)}; Y_{k+1} = Y_k + U_{k+1}, \quad k = 1, 2, \dots; \quad Y_1 = U_1 = x.$$

5. Обчислити $\cos x$ за наближеною формулою (у радіанах):

$$U_{k+1} = -U_k \frac{x^2}{(2k-1)2k}; Y_{k+1} = Y_k + U_{k+1}, k = 1, 2, \dots; Y_1 = U_1 = 1$$

6. Обчислити константу e за наближеною формулою:

$$e = 1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \dots + \frac{1}{n!}, \text{ де } n - \text{натуральне число.}$$

7. Обчислити $\arctg(x)$ за наближеною формулою:

$$\arctg(x) = x - \frac{x^3}{3} + \frac{x^5}{5} - \dots + (-1)^{m-1} \frac{x^{2m-1}}{2m-1}.$$

8. Обчислити $\arcsin x$ за наближеною формулою:

$$\arcsin x = x + \frac{x^3}{2*3} + \frac{1*3*x^5}{2*4*5} + \frac{1*3*5*x^7}{2*4*6*7} + \dots + \frac{1*3*5...(2n-1)x^{2n+1}}{2*4*6...(2n)(2n+1)},$$

де $|x| < 1$.

9. Обчислити $\arccos x$ за наближеною формулою:

$$\arccos x = \frac{\pi}{2} - \left(x + \frac{x^3}{2*3} + \frac{1*3*x^5}{2*4*5} + \frac{1*3*5*x^7}{2*4*6*7} + \dots + \frac{1*3*5...(2n-1)x^{2n+1}}{2*4*6...(2n)(2n+1)} \right),$$

де $|x| < 1$.

10. Обчислити гіперболічний косінус за наближеною формулою:

$$\cosh x = 1 + \frac{x^2}{2!} + \frac{x^4}{4!} + \frac{x^6}{6!} + \dots + \frac{x^{2n}}{(2n)!},$$

де $|x| < \infty$.

11. Обчислити гіперболічний синус за наближеною формулою:

$$\sinh x = x + \frac{x^3}{3!} + \frac{x^5}{5!} + \frac{x^7}{7!} + \dots + \frac{x^{2n+1}}{(2n+1)!},$$

де $|x| < \infty$.

12. Обчислити $\ln x$ за наближеною формулою:

$$\ln x = 2 \left[\frac{(x-1)}{x+1} + \frac{(x-1)^3}{3(x+1)^3} + \frac{(x-1)^5}{5(x+1)^5} + \dots + \frac{(x-1)^{2n+1}}{(2n+1)(x+1)^{2n+1}} \right],$$

де n – натуральне число.

13. Обчислити $y = a^x$, де

$$a = 1 + \frac{1}{2} + \frac{1}{4} + \dots + \frac{1}{2^n},$$

де n – натуральне число.

14. Обчислити a^x за наближеною формулою:

$$a^x = e^{x \ln a} = 1 + \frac{x \ln a}{1!} + \frac{(x \ln a)^2}{2!} + \frac{(x \ln a)^3}{3!} + \dots$$

15. Знайти суму ряду $\frac{x-1}{x} + \frac{(x-1)^2}{2x^2} + \frac{(x-1)^3}{3x^3} + \dots + \frac{(x-1)^n}{nx^n} + \dots$

16. Обчислити число π з точністю 10^{-6} , користуючись рядом Грегорі:

$$\frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \dots + (-1)^{n+1} \cdot \frac{1}{n-1} + \dots$$

17. Обчислити число π з точністю 10^{-6} , користуючись добутком Валліса:

$$\frac{\pi}{2} = \frac{2}{1} * \frac{2}{3} * \frac{4}{3} * \frac{4}{5} * \frac{6}{5} * \frac{6}{7} * \dots \frac{(2n)^2}{(2n-1)(2n+1)} \dots$$

18. Обчислити $\operatorname{arcsinh} x$ за наближеною формулою:

$$\operatorname{arcsinh} x = x - \frac{1 * x^3}{2 * 3} + \frac{1 * 3}{2 * 4 * 5} x^5 - \frac{1 * 3 * 5}{2 * 4 * 6 * 7} x^7 + \dots + (-1)^n \frac{1 * 3 * 5 \dots (2n-1)^2}{2 * 4 * 6 \dots 2n(2n+1)} x^{2n+1} + \dots$$

19. Написати програму обчислення суми ряду:

$$S = 1 - \frac{x^2}{2} + \frac{1 * 3 x^2}{2 * 4} - \frac{1 * 3 * 5 x^3}{2 * 4 * 6} + \frac{1 * 3 * 5 * 7 * x^4}{2 * 4 * 6 * 8} + \dots$$

20. Написати програму обчислення e^x за формулою:

$$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots + \frac{x^n}{n!} + \dots$$

21. Обчислити $(1+x)^m$ за наближеною формулою:

$$(1+x)^m = 1 + mx + \frac{m(m-1)}{1 * 2} x^2 + \dots + \frac{m(m-1) \dots (m-n+1)}{1 * 2 * \dots * n} x^n$$

22. Обчислити $\sin x$ за наближеною формулою (у радіанах):

$$\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \dots + (-1)^n \frac{x^{2n+1}}{(2n+1)!},$$

де n – натуральне число.

23. Обчислити $\cos x$ за наближеною формулою (у радіанах):

$$\cos x = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \dots + (-1)^n \frac{x^{2n}}{(2n)!},$$

де n – натуральне число.

24. Обчислити значення

$$P(x): P(x) = 1 - x + \frac{x^2}{2} - \frac{x^3}{3} + \dots - (-1)^n \frac{x^n}{n}$$

де n – натуральне число.

25. Знайти суму ряду $\frac{1}{\sqrt{1 * 2 * 3 * 4 + 1}} + \frac{2}{\sqrt{2 * 3 * 4 * 5 + 1}} + \dots + \frac{k}{\sqrt{k(k+1)(k+2)(k+3) + 1}}$

де k – натуральне число.

Завдання 2.

1. Обчислити вірогідність появи випадкової події рівно m разів в n незалежних дослідах за формулою Бернуллі: $P_n(m) = C_n^m p^m q^{n-m}$, де $C_n^m = \frac{n!}{(n-m)!m!}$; $q = 1-p$;

n, m – цілі додатні числа, p – дійсне додатне число з інтервалу $(0;1)$.

2. Обчислити інтегральну функцію $\Phi(x)$, використовуючи наближену формулу ($x > 0$):

$$\Phi(x) = x \sqrt{\frac{2}{\pi}} \left(1 - \frac{(x/2)^2}{1!3} + \frac{(x/2)^4}{2!5} - \frac{(x/2)^6}{3!7} + \dots \right).$$

3. Обчислити $\ln 2$ за наближеною формулою

$$\ln 2 = 1 - \frac{1}{2} + \frac{1}{3} - \frac{1}{4} + \dots \pm \frac{1}{n}.$$

4. Обчислити $F \left[3.8405 \cdot 10^{-3} \sum_{i=1}^n \sqrt{i!} \right]$, де $F[]$ – символ виділення цілої частини числа,

яке знаходиться у квадратних дужках; $i = 1, 2, \dots, n$.

5. Обчислити $y = \frac{1}{m} \sum_{n=1}^m k^n$, де $n = 1, 2, \dots, m$; $k = \text{const}$.

6. Обчислити константу e за наближеною формулою:

$$e = 1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \dots + \frac{1}{n!},$$

де n – натуральне число.

7. Визначити число размещень за формулою: $A_n^m = \frac{n!}{(n-m)!}$.

8. Обчислити $\arcsin x$ за наближеною формулою:

$$\text{Arcsin} x = x + \frac{x^3}{2 \cdot 3} + \frac{1 \cdot 3 \cdot x^5}{2 \cdot 4 \cdot 5} + \frac{1 \cdot 3 \cdot 5 \cdot x^7}{2 \cdot 4 \cdot 6 \cdot 7} + \dots + \frac{1 \cdot 3 \cdot 5 \dots (2n-1) x^{2n+1}}{2 \cdot 4 \cdot 6 \dots (2n)(2n+1)},$$

де $|x| < 1$.

9. Обчислити $\arccos x$ за наближеною формулою:

$$\text{Arccos} x = \frac{\pi}{2} - \left(x + \frac{x^3}{2 \cdot 3} + \frac{1 \cdot 3 \cdot x^5}{2 \cdot 4 \cdot 5} + \frac{1 \cdot 3 \cdot 5 \cdot x^7}{2 \cdot 4 \cdot 6 \cdot 7} + \dots + \frac{1 \cdot 3 \cdot 5 \dots (2n-1) x^{2n+1}}{2 \cdot 4 \cdot 6 \dots (2n)(2n+1)} \right),$$

де $|x| < 1$.

10. Обчислити гіперболічний косінус за наближеною формулою:

$$\text{ch} x = 1 + \frac{x^2}{2!} + \frac{x^4}{4!} + \frac{x^6}{6!} + \dots + \frac{x^{2n}}{(2n)!}, \text{ де } |x| < \infty.$$

11. Обчислити гіперболічний синус за наближеною формулою:

$$\text{sh} x = x + \frac{x^3}{3!} + \frac{x^5}{5!} + \frac{x^7}{7!} + \dots + \frac{x^{2n+1}}{(2n+1)!},$$

де $|x| < \infty$.

12. Обчислити $\ln x$ за наближеною формулою:

$$\ln x = 2 \left[\frac{(x-1)}{x+1} + \frac{(x-1)^3}{3(x+1)^3} + \frac{(x-1)^5}{5(x+1)^5} + \dots + \frac{(x-1)^{2n+1}}{(2n+1)(x+1)^{2n+1}} \right],$$

де n – натуральне число.

13. Обчислити $y = a^* a$, где

$$a = 1 + \frac{1}{2} + \frac{1}{4} + \dots + \frac{1}{2^n},$$

де n – натуральне число.

14. Обчислити a^x за наближеною формулою: $a^x = e^{x \ln a} = 1 + \frac{x \ln a}{1!} + \frac{(x \ln a)^2}{2!} + \frac{(x \ln a)^3}{3!} + \dots$

15. Знайти суму ряду $\frac{x-1}{x} + \frac{(x-1)^2}{2x^2} + \frac{(x-1)^3}{3x^3} + \dots + \frac{(x-1)^n}{nx^n} + \dots$

16. Обчислити число π з точністю 10^{-6} , користуючись рядом Грегорі:

$$\frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \dots + (-1)^{n+1} \cdot \frac{1}{n+1} + \dots$$

17. Обчислити число π з точністю 10^{-6} , користуючись добутком Валліса:

$$\frac{\pi}{2} = \frac{2}{1} * \frac{2}{3} * \frac{4}{3} * \frac{4}{5} * \frac{6}{5} * \frac{6}{7} \dots \frac{(2n)^2}{(2n-1)(2n+1)} \dots$$

18. Обчислити arcsch за наближеною формулою:

$$\operatorname{arcsch} x = x - \frac{1 * x^3}{2 * 3} + \frac{1 * 3}{2 * 4 * 5} x^5 - \frac{1 * 3 * 5}{2 * 4 * 6 * 7} x^7 + \dots + (-1)^n \frac{1 * 3 * 5 \dots (2n-1)^2}{2 * 4 * 6 \dots 2n(2n+1)} x^{2n+1} + \dots$$

19. Написати програму обчислення суми ряду

$$S = 1 - \frac{x^2}{2} + \frac{1 * 3 x^2}{2 * 4} - \frac{1 * 3 * 5 x^3}{2 * 4 * 6} + \frac{1 * 3 * 5 * 7 * x^4}{2 * 4 * 6 * 8} + \dots$$

20. Написати програму обчислення e^x за формулою:

$$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots + \frac{x^n}{n!} + \dots$$

21. Обчислити появу випадкової події рівно n разів за формулою Пуассона:

$$P(n) = \frac{\lambda^n}{n!} e^{-\lambda}, \text{ де } n = 0, 1, 2, \dots; \lambda > 0; n! = 1 * 2 * 3 \dots n.$$

22. Обчислити $\sin x$ за наближеною формулою (у радіанах):

$$\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \dots + (-1)^n \frac{x^{2n+1}}{(2n+1)!},$$

де n – натуральне число.

23. Обчислити $\cos x$ за наближеною формулою (у радіанах):

$$\cos x = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \dots + (-1)^n \frac{x^{2n}}{(2n)!},$$

де n – натуральне число.

24. Визначити число сполучень за формулою:

$$C_n^m = \frac{n!}{(n-m)!m!},$$

де n, m – натуральні числа, $m \leq n$.

25. Знайти суму ряду

$$\frac{1}{\sqrt{1 * 2 * 3 * 4 + 1}} + \frac{2}{\sqrt{2 * 3 * 4 * 5 + 1}} + \dots + \frac{k}{\sqrt{k(k+1)(k+2)(k+3) + 1}},$$

де k – натуральне число.

3. КОНТРОЛЬНІ РОБОТИ

3.1 ОС WINDOWS

У папці **comps** перебувають дані щодо комп'ютерів деякого класу. Завдання полягає в структуруванні цієї інформації. Практично необхідно створити систему папок і файлів для однієї з машин за Вашим варіантом (завдання 1) і зібрати в одному файлі дані по всіх машинах відносно однієї з підсистем, наприклад, відео (завдання 2), також за Вашим варіантом. Для файлу завдання 2 необхідно організувати посилання у вигляді значка, який слід вибрати з папки **icons**.

Варіант завдання вказується викладачем.

Порядок виконання роботи

1. У папці "Ваша група" створити папку "**Ваше прізвище**", наприклад, "Іванов".
2. У папці "Ваше прізвище" створити папку "**Ім'я комп'ютера**" (наприклад, ITGSH9), який слід взяти з файлу з описом комп'ютера за Вашим варіантом (папка comps).
3. У папці "Ім'я комп'ютера" створити однорівневу систему папок: "Відеосистема", "Мережа", "Системна плата", "Зберігання даних".
4. У кожній із зазначених у п. 3 папок створити текстовий файл (Блокнот) з аналогічною папці назвою і скопіювати в нього дані для відповідної підсистеми з файлу з описом комп'ютера за Вашим варіантом.
5. У папці "Ваше прізвище" створити текстовий файл (Блокнот) з назвою підсистеми за Вашим варіантом (наприклад, мережа). Записати в нього дані для Вашої підсистеми всіх машин, перерахованих у папці comps. Кожен опис слід позначати ім'ям комп'ютера.
6. Створити ярлик для отриманого в п. 5 файлу. Ім'я ярлика – Ваше прізвище, назва підсистеми (наприклад, Іванов мережа). Розташування – у папці "Ваше прізвище". Значок ярлика вибрати довільно з папки **icons**.
7. Результуючу папку зберегти на знімному носії.
8. Результати роботи оформити у вигляді звіту та здати роботу викладачеві.

Варіанти

Варіант	Файл опису комп'ютера	Підсистема
1	Машина 1	Мережа
2	Машина 2	Зберігання даних
3	Машина 4	Відео
4	Машина 5	Системна плата
5	Машина 6	Мультимедіа
6	Машина 8	Мультимедіа
7	Машина 9	Системна плата
8	Машина 11	Відео
9	Машина 12	Зберігання даних
10	Машина 14	Мережа

3.2 Текстовий процесор Microsoft Word

Написати реферат за одною з поданих тем. Реферат має містити не менше ніж 3 розділи, титульний лист, зміст, список використаних джерел. Основний текст реферату має бути форматованим за допомогою стилю, який створений користувачем на базі звичайного стилю; йому треба привласнити ім'я – «Ваше прізвище». Для назв розділів слід створити стиль «Ваше прізвище1». Зміст треба створити автоматично, використовуючи стилі.

Оформити реферат, використовуючи графіку та малюнки за Вашим розсудом.

Теми рефератів

1. Передача, перетворення, зберігання й використання інформації в техніці.
2. Мова як спосіб подання інформації, двійкова форма подання інформації, її особливості й переваги.
3. Принципи роботи з даними і командами у комп'ютері.

4. Принцип автоматичного виконання програм в ЕОМ.
5. Операційні системи сімейства UNIX.
6. Побудова й використання комп'ютерних моделей.
7. Телекомунікації, телекомунікаційні мережі різного типу, їхнє призначення й можливості.
8. Мультимедійні технології.
9. Роль інформатики в житті суспільства.
10. Інформація в спілкуванні людей.
11. Підходи до оцінки кількості інформації.
12. Історія розвитку ЕОМ.
13. Сучасний стан електронно-обчислювальної техніки.
14. Класи сучасних ЕОМ.
15. Шкідливий вплив комп'ютера. Засоби захисту.
16. Суперкомп'ютери і їхнє застосування.
17. Ноутбук – пристрій для професійної діяльності.
18. Кишенькові персональні комп'ютери.
19. Основні типи принтерів.
20. Сканери та програмне забезпечення розпізнавання символів.
21. Мережа Інтернет і кіберзлочинність.
22. Криптографія.
23. Комп'ютерна графіка на ЕОМ.
24. WWW. Історія створення й сучасність.
25. Проблеми створення штучного інтелекту.
26. Використання Інтернету в маркетингу.
27. Пошук інформації в Інтернеті. Web-індекси, Web-каталоги.
28. Системи електронних платежів.
29. Комп'ютерна грамотність та інформаційна культура.
30. Пристрої введення інформації.

3.3 Power Point

Зробіть презентацію до Вашого реферату. Презентація має складатися не менш ніж з 5 слайдів, які мають бути оформлені за вашим розсудом для найбільш наочного подання вашої теми з використанням різних ефектів, малюнків та діаграм.

3.4 Мова програмування VBA – Visual Basic for Applications

3.4.1 Лінійні обчислювальні процеси

Завдання

1.

$$f = z^3 + z + 1; k = z - 35$$

$$z = a^2 \cos(x + 1) + 2 \sin(y^2 + 5)$$

$$x = 0,25 + t + \cos^3(0,25 + t)$$

t – будь-яке

$$a = 14S + 1$$

$$S = 9,6$$

y – будь-яке

2.

$$p = n + f^2 + f$$

$$y = p^2 + c + m^2 f$$

$$f = 3c^3 + 5 + \cos^3(3c^3 + 5) + \sin(3c^3 + 5)$$

$$n = m^7 + \sqrt{m + 1}$$

$$m = db + \frac{\sqrt{b^2 + 7}}{15}$$

d, b, c – будь-які, $a = 16$

3.

$$x = 0,05 + \sin(a^2 yz) - 6 - z^5$$

$$y = \sin(a^2 bz) + |a^2 bz| + \frac{a^2 bz}{7}$$

$$b=0,02; a - \text{будь-яке}$$

$$z = 100 - \frac{189}{|b-a|}$$

$$z = 3p + 8p^2 + p^6$$

$$p = 0,6$$

5.

$$S = at^2 + 1$$

$$p = 6,2; z = b \sin x + \ln |b \sin x|$$

$$x = y^3 + py^2 + Sy$$

$$t, a - \text{будь-яке}$$

$$f = \sqrt{\sin S + xa} + |p-1 + \sin S|$$

$$y = a^5 - a + 7; b = \sin(S)$$

7.

$$m = 3,2a + 6,45b - c$$

$$k = m^3 + a - b$$

$$a - \text{будь-яке}$$

$$t = a^2 + mb - k$$

$$c = -6,66$$

$$b = a^3 + a - 1 + \ln |a^3 + a - 1|$$

$$s = m + kt$$

9.

$$b = 10s + s^2 + 1$$

$$x = 0,7 + t - \cos(0,7 + t)$$

$$s = -3,4; m = y + z$$

$$t - \text{будь-яке}$$

$$Q = z + z^5 + 3; z = b - 4$$

$$y = b^3 \cos(x-1) - 4 \sin(b+5)$$

11.

$$x - \text{будь-яке}; a - \text{будь-яке}$$

$$n = am^2 + 47a^2m^5$$

$$m = xa - \cos^3(xa)$$

$$t = -3,33$$

$$p = (n+m) + z^3 + \sin(m+n-z)$$

$$z = t^3 + x^2 + a; s = p^3 + 5$$

13.

$$i = 0,8j + 25k$$

$$p = 2 \sin(s+1) + \sqrt{s+1} + \ln(s+1)$$

$$S = abc; a, b - \text{будь-які}$$

$$j = 3,45; k = 12j - j^7 + 5$$

$$c = 6,31 - j$$

$$f = 100 + \frac{14}{i+j+k}$$

4.

$$S = Vt; k = at + 5$$

$$a = \sqrt[3]{S+25} + \sin(S+25) + |S^5 - 1 + S|$$

$$t - \text{будь-яке}; V = \frac{u^2 + w^2}{5}$$

$$u = t^7 + 3; w = 3,41$$

6.

$$n = a^3 + \sqrt{a^3 - 1}$$

$$C = 0,5f^3 + f + 1; Q = S^3 + n - 5$$

$$b = 25 - a; f - \text{будь-яке}; a = -3,1$$

$$r = n + a + \sin(n+a) - \sqrt{n+a}$$

$$S = C + rb + r^2a$$

8.

$$a, b, c - \text{будь-які}; d = 0,35$$

$$n = m^3 + \sqrt{am} + \sin(am)$$

$$m = b - 0,65d$$

$$z = 3(c^2 + 1) + |c^2 + 1|$$

$$y = x + c - mz$$

$$x = n + z$$

10.

$$d = f^2 + a + b$$

$$f = 30a + 22b + c^5$$

$$t = a^2 + f^3 + d$$

$$c = 105$$

$$b = a^7 + \ln(a+1) + \sin(a^7 + \ln(a+1))$$

$$Q = f + dt; a - \text{будь-яке}$$

12.

$$A = 0,654; d, t, c - \text{будь-які}$$

$$z = d^3 + 6$$

$$y = z - d + \sqrt{d^2 + 1} + d^2 + 1$$

$$x = z - A$$

$$u = |tc| - \cos y + l$$

$$l = \sin Atc$$

14.

$$p = z^3 + az^2 + bz + c$$

$$a - \text{будь-яке}$$

$$b = 0,12Q - 1; f = 154$$

$$z = a - t^5 + \sin |a - t^5|$$

$$c = 0,34f - f^3 + f^5$$

$$Q = \sin(f+1) + t$$

$$t = a + \sqrt{|a+s|}$$

15.

$$a = \frac{b^3 + x^{z+k}}{0.63y^2 + 1}$$

$$x = 2z^5 + y; \quad f = \sin(c)$$

$$y = z + \sqrt[3]{f}$$

c – будь-яке

$$k=3; \quad b=4; \quad z=14$$

17.

$$S = a \cos^2 x + \sqrt{\cos^2 x}$$

$$a = \ln |t-1| + e^{t-1}$$

$$x = c - \sqrt[5]{t-1}$$

$$c=0,45t - \text{будь-яке}$$

19.

$$y = \frac{2m^2 + \sqrt[3]{bx^2} + \sin^2 x}{\ln(2x-a) + e^{2x}} + 1,67c$$

$$z = 5b^2 + \ln y + \cos x - \sqrt{c^x}$$

$$s = \sqrt{y+2z} + tgx$$

$$a, b, c, m - \text{будь-які} \quad x=5$$

21.

$$y = \frac{at^2 + 2t \sin^2 x + 1,35}{tgx - \ln a - 16,2 \ln x}$$

$$z = \frac{2ab + 3t - \sin x}{\arctg |5-y|}$$

$$a, t - \text{будь-які} \quad x=5,2$$

23.

$$y = \frac{5ax\sqrt[3]{c+x} - \ln x}{a+x+c - \sin^2 x}$$

$$r = \frac{y}{2x} - \ln \frac{a+b\sqrt{x}}{c - \sin x}$$

$$s = \frac{r}{x} + \frac{\sqrt{r}}{2x} - \frac{\sqrt{y}}{3r}$$

$$a, b, c - \text{будь-які} \quad x=35,2$$

25.

$$y = \frac{ab^2 + \sqrt[3]{a+b} - 0,8c}{\sin 2x + 0,92}$$

$$z = \sqrt{a} + 2bc + |y| - 0,65 \ln x + e^{-x}$$

$$a, b, c - \text{будь-які} \quad x=17,6$$

16.

$$c = \sqrt{t+1}$$

$$y = \frac{0,2z^x + 0,5x}{\ln^2 |c-b^k|}; \quad x=0,83$$

$$z = 2c^3 + b; \quad b = 0,7c^2 + f$$

$$f, t, k - \text{будь-які}$$

18.

$$z = \frac{4(x^y + y^k)}{\sqrt{\ln^2 c + b^4}}$$

$$y = 6,2 \quad b = \frac{3c^3}{y} \quad c = \left| \frac{2d^3}{0,2y} \right|$$

$$d, k - \text{будь-які} \quad x=5$$

20.

$$z = \frac{mb^2 + \sqrt[3]{c} - \ln a + e^{2x}}{m-3 + \ln \sqrt{x} + \sqrt{x}}$$

$$s = \frac{\ln(a+z - \sin x)}{\sqrt[5]{a + \ln(x+c)}},$$

$$a, b, c, m - \text{будь-які} \quad x=5$$

22.

$$r = \frac{c + \ln x - e^x}{\arctg(b+x)} - \frac{\sqrt{x}}{\lg x}$$

$$s = \frac{\sqrt{x}}{\sin x} + \frac{\ln(e^x + 2^x)}{|\sin^2 2x|}$$

$$b, c - \text{будь-які} \quad x=15,2$$

24.

$$y = (e^{-x_1} + e^{-x_2})/2$$

$$z = (a\sqrt{x_1} - b\sqrt{x_2})/c$$

$$x_1 = (b + \sqrt{|b^2 - 4ac|})/2a$$

$$x_2 = (b - \sqrt{|b^2 - 4ac|})/2a$$

$$a, b, c - \text{будь-які}$$

3.4.2 Розгалужені обчислювальні процеси

3.4.2.1 Прості логічні вирази

Завдання:

№	ЗАВДАННЯ
1	$mf = \begin{cases} x-1, & \text{якщо } x < 1 \\ ax+b, & \text{якщо } x=1, \text{ при } a=12,4; b=56,1; x \text{ – довільне значення} \\ 1+b/a, & \text{якщо } x > 1 \end{cases}$
2	$mf = \begin{cases} x+1, & \text{якщо } x < 1 \\ a-x+b, & \text{якщо } x=1, \text{ при } a=0,001; b=5,1; x \text{ – довільне значення} \\ x-(b/a), & \text{якщо } x > 1 \end{cases}$
3	$ff = \begin{cases} 1-\cos x-y, & \text{якщо } y < 0 \\ (ay+b)/2, & \text{якщо } y=0, \text{ при } a=0,2; b=0,01; x,y \text{ – довільне значення} \\ a+1, & \text{якщо } y > 0 \end{cases}$
4	$mf = \begin{cases} 1-a \cos x, & \text{якщо } x < 1 \\ ax+b, & \text{якщо } x=1, \text{ при } a=0,001; b=5,1; x \text{ – довільне значення} \\ x+(b/a), & \text{якщо } x > 1 \end{cases}$
5	$xf = \begin{cases} \log(ay+1), & \text{якщо } y < 2 \\ (ay+b)/2, & \text{якщо } y=0, \text{ при } a=0,2; b=0,01; y \text{ – довільне значення} \\ a+1, & \text{якщо } y > 0 \end{cases}$
6	$mf = \begin{cases} x+1, & \text{якщо } x < 1 \\ ax+b, & \text{якщо } x=1, \text{ при } a=0,001; b=5,1; x \text{ – довільне значення} \\ x+(b/a), & \text{якщо } x > 1 \end{cases}$
7	$wf = \begin{cases} b-yx, & \text{якщо } y > 1 \\ \sin(b-y), & \text{якщо } y=1 \\ (x-y)/2, & \text{якщо } y < 1, \text{ при } b=0,15; x=2, y \text{ – довільне значення} \end{cases}$
8	$ff = \begin{cases} c-ba, & \text{якщо } a > 0 \\ a-((b/2)+1), & \text{якщо } a=0, \text{ при } b=1,2; c=1,71; a \text{ – довільне значення} \\ a-1, & \text{якщо } a < 0 \end{cases}$
9	$ff = \begin{cases} 0,5y-1, & \text{якщо } y=1 \\ (1-y)/a, & \text{якщо } y > 1, \text{ при } a=18,12; y \text{ – довільне значення} \\ (1+y), & \text{якщо } y < 1 \end{cases}$
10	$ff = \begin{cases} (1-x)/2, & \text{якщо } x > 3 \\ \sin x-2, & \text{якщо } x=3 \\ (1-x)/(a-b), & \text{якщо } x < 3, \text{ при } a=5,5; b=1,2; x \text{ – довільне значення} \end{cases}$

$$11 \quad kf = \begin{cases} 1-xy, & \text{якщо } y < 3 \\ 1+xy, & \text{якщо } y=3, \text{ при } x=0, 812; y \text{ – довільне значення} \\ 1-2xy, & \text{якщо } y > 3 \end{cases}$$

$$12 \quad yf = \begin{cases} (x+2)/3, & \text{якщо } x < 2 \\ x-a, & \text{якщо } x=2 \\ (x-2)/(a-1), & \text{якщо } x > 2, \text{ при } a=7,21 \text{ } x \text{ – довільне значення} \end{cases}$$

$$13 \quad sf = \begin{cases} 1-y, & \text{якщо } y=1 \\ (1+y)/a, & \text{якщо } y < 1, \text{ при } a=0,412; y \text{ – довільне значення} \\ 1+3,5y, & \text{якщо } y > 1 \end{cases}$$

$$14 \quad gf = \begin{cases} (1-x), & \text{якщо } x < 0 \\ (1+2x)/(a-2), & \text{якщо } x > 0, \text{ при } a=6,31; x \text{ – довільне значення} \\ 1+x, & \text{якщо } x=0 \end{cases}$$

$$15 \quad ff = \begin{cases} ((a-b)/2+x), & \text{якщо } x > 1 \\ x \log(a+b), & \text{якщо } x=1 \\ ((a+b)x)/a, & \text{якщо } x < 1, \text{ при } a=0,7; b=0,31 \text{ } x \text{ – довільне значення} \end{cases}$$

$$16 \quad sf = \begin{cases} 1-x, & \text{якщо } x=0 \\ (1+xb)/a, & \text{якщо } x < 0, \text{ при } a=7,2; b=0,001; x \text{ – довільне значення} \\ (1+x)/0,5, & \text{якщо } x > 0 \end{cases}$$

$$17 \quad xf = \begin{cases} (m-1,5y)/2, & \text{якщо } y < 2 \\ mb-y, & \text{якщо } y=2 \\ y+mb, & \text{якщо } y > 2, \text{ при } b=0,2; m=5 \text{ } y \text{ – довільне значення} \end{cases}$$

$$18 \quad sf = \begin{cases} 3,5x-15,2y, & \text{якщо } y > 2 \\ 8,1x+\sin(y-1), & \text{якщо } y=2 \\ (1,2x-5,1y)/a, & \text{якщо } y < 2, \text{ при } a=0,1 \text{ } x, y \text{ – довільне значення} \end{cases}$$

$$19 \quad qf = \begin{cases} 3,1a-x, & \text{якщо } x=1 \\ 5,2b-x, & \text{якщо } x > 1, \text{ при } a=5,2; b=1,6; x \text{ – довільне значення} \\ x-1, & \text{якщо } x < 1 \end{cases}$$

$$20 \quad rf = \begin{cases} 3,1x+b, & \text{якщо } x > 3 \\ (x-1)/2, & \text{якщо } x < 3, \text{ при } b=3,21; x \text{ – довільне значення} \\ (x+b), & \text{якщо } x=3 \end{cases}$$

$$21 \quad cf = \begin{cases} (x-1)/(a-1), & \text{якщо } x=5 \\ (x+1)/a, & \text{якщо } x > 5, \text{ при } a=642,1 \text{ } x \text{ – довільне значення} \\ (x-1), & \text{якщо } x < 5 \end{cases}$$

22	$yf = \begin{cases} (1-x)/(c+1), & \text{якщо } x > 0 \\ (1-x)b, & \text{якщо } x < 0, \text{ при } b=1,2; c=6,62; x - \text{довільне значення} \\ (1-x)c, & \text{якщо } x = 0 \end{cases}$
23	$yf = \begin{cases} (1+x)b, & \text{якщо } x=10 \\ b-x, & \text{якщо } x < 10, \text{ при } b=6,531 \text{ } x - \text{довільне значення} \\ b+x, & \text{якщо } x > 10 \end{cases}$
24	$ff = \begin{cases} y-1, & \text{якщо } y > 1 \\ y+b, & \text{якщо } y = 1 \\ ((y+1)b)/(a-b), & \text{якщо } y < 1, \text{ при } a=0,2; b=0,3 \text{ } y - \text{довільне значення} \end{cases}$
25	$af = \begin{cases} x+b, & \text{якщо } b < 0 \\ 0,5x-b, & \text{якщо } b > 0, \text{ при } x=3,12; b,y - \text{довільні значення} \\ ux+b, & \text{якщо } b=0 \end{cases}$

3.4.2.2 Складові логічні вирази

Завдання

1	$z = \begin{cases} \sin(x+y), & \text{якщо } x > 1 \text{ або } y > 1; \\ x+y , & \text{якщо } x < -1 \text{ і } x > y; \\ e^{x-y}, & \text{якщо } 0 \leq x \leq 1, \\ \cos(x+1), & \text{в інших випадках.} \end{cases} \quad x, y - \text{довільні значення}$
2	$y = \begin{cases} x^3 , & \text{якщо } x < ab \text{ і } x < -2; \\ \sin(x^a + b), & \text{якщо } x > ab \text{ або } x > 2; \\ a+b+1, & \text{якщо } 0 < x < 1 \text{ і } x = ab; \\ x - (a+b)^2, & \text{в інших випадках.} \end{cases} \quad x, a, b - \text{довільні значення}$
3	$f = \begin{cases} c+x, & \text{якщо } 1 \leq x \leq 3; \\ c-x, & \text{якщо } 3 < x \leq 10 \text{ і } c > 0; \\ c ^x, & \text{якщо } x > 10; \\ x, & \text{в інших випадках.} \end{cases} \quad c, x - \text{довільні значення}$
4	$w = \begin{cases} e^v, & \text{якщо } v > x \text{ або } v > y; \\ (x+y), & \text{якщо } v = 10 \text{ і } y > x > 15; \\ v^{x+y}, & \text{якщо } v = x; \\ 1, & \text{в інших випадках.} \end{cases} \quad v, x, y - \text{довільні значення}$
5	$t = \begin{cases} s+2z, & \text{якщо } s+z = 2,8 \text{ або } s+z = 1; \\ s-z^2, & \text{якщо } s+z > 5 \text{ і } z > 0; \\ s+z , & \text{якщо } 3 < s+z \leq 5; \\ a+s, & \text{в інших випадках.} \end{cases} \quad a, s, z - \text{довільні значення}$

$$6 \quad y = \begin{cases} a+b+c, & \text{якщо } a+b > c; \\ (a+b)-c, & \text{якщо } a+b < c \text{ і } c > 5; \\ |\sin(b)|, & \text{якщо } a+b = c; \\ a^{bc} & \text{в інших випадках} \end{cases} \quad a, b, c - \text{довільні значення}.$$

$$7 \quad z = \begin{cases} 1, & \text{якщо } y < 0; \\ 2, & \text{якщо } 0 \leq y < 5; \\ 3, & \text{якщо } 5 \leq y < 10; \\ 4, & \text{якщо } y \geq 10. \end{cases} \quad y - \text{довільне значення}$$

$$8 \quad s = \begin{cases} t-1, & \text{якщо } 0 < zt < 5; \\ z, & \text{якщо } zt > 10 \text{ або } zt < -3; \\ t, & \text{якщо } zt = 0; \\ 0, & \text{в інших випадках.} \end{cases} \quad z, t - \text{довільні значення}$$

$$9 \quad m = \begin{cases} \sin^2(x) + \cos(x), & \text{якщо } x > 0; \\ \frac{x+y}{a^2+2} + \sin\left(\frac{x+y}{a^2+2}\right), & \text{якщо } x \leq 0 \text{ і } a > 0; \\ x + |a-y|, & \text{в інших випадках.} \end{cases} \quad x, y, a - \text{довільні значення}$$

$$10 \quad n = \begin{cases} |m+c|, & \text{якщо } m > 10 \text{ або } m = 4; \\ m-c, & \text{якщо } m < 3; \\ m, & \text{якщо } 5 < m < 8; \\ mc, & \text{в інших випадках.} \end{cases} \quad m, c - \text{довільні значення}$$

$$11 \quad x = \begin{cases} t+q, & \text{якщо } t > 10 \text{ і } t = q; \\ t+1, & \text{якщо } t \leq -10; \\ t, & \text{якщо } t = 5; \\ 1+ tq & \text{в інших випадках.} \end{cases} \quad t, q - \text{довільні значення}$$

$$12 \quad k = \begin{cases} x+y^2+2, & \text{якщо } x = y+2; \\ x+y+2, & \text{якщо } x > y+2 \text{ і } y = 3; \\ \sin(x) + \sin(2), & \text{якщо } 0 < x < y+2; \\ x + \ln|y+2y| & \text{в інших випадках.} \end{cases} \quad x, y - \text{довільні значення}$$

$$13 \quad z = \begin{cases} 2k, & \text{якщо } k < 2; \\ 3k, & \text{якщо } 2 \leq k \leq 6; \\ 4k, & \text{якщо } 6 < k < 10; \\ 5k, & \text{якщо } k \geq 10. \end{cases} \quad k - \text{довільне значення}$$

$$14 \quad p = \begin{cases} t+a, & \text{якщо } t > 0 \text{ або } a > 0; \\ t-a, & \text{якщо } t < 0 \text{ і } t < 0; \\ |a| & \text{в інших випадках.} \end{cases} \quad a, t - \text{довільні значення}$$

15	$g = \begin{cases} \ln(a^2 + 1), & \text{якщо } 15 \leq a < 20; \\ 2a + 7, & \text{якщо } 10 \leq a < 15; \\ 1 - ab, & \text{якщо } 8 < a < 10; \\ \frac{4a}{7(a+b)^2 + 5}, & \text{якщо } 0 \leq a < 8 \text{ і } a > b; \\ a + b & \text{в інших випадках.} \end{cases}$	a, b - довільні значення
16	$z = \begin{cases} 25y + a, & \text{якщо } y > a \text{ і } a > 10; \\ y + 3a, & \text{якщо } a < y < 20; \\ \cos(ay), & \text{якщо } y = a; \\ 1 & \text{в інших випадках.} \end{cases}$	a, y - довільні значення
17	$v = \begin{cases} z^5 + z + 1, & \text{якщо } 1 < z^5 < 3; \\ \cos(z^5 + z + 1), & \text{якщо } 3 \leq z^5 < 7; \\ \ln(z^5 + z + 1), & \text{якщо } 7 \leq z^5 < 10; \\ \frac{1}{z^4 + z^2 + 1}, & \text{якщо } z^5 \geq 10 \text{ або } z^5 \leq 1. \end{cases}$	z - довільне значення
18	$x = \begin{cases} t^3 + 5, & \text{якщо } t < -2; \\ t - 1, & \text{якщо } -2 \leq t < 0; \\ \sin(t), & \text{якщо } 0 \leq t < 5; \\ \ln(t), & \text{якщо } t \geq 5. \end{cases}$	t - довільне значення
19	$y = \begin{cases} a + z, & \text{якщо } a + z > 0 \text{ і } z < 0; \\ az, & \text{якщо } z > 5 \text{ або } a + z < -3; \\ a - x, & \text{в інших випадках.} \end{cases}$	a, z, x - довільні значення
20	$f = \begin{cases} \frac{1}{(ay)^2 + 1}, & \text{якщо } y = -4; \\ 2xy, & \text{якщо } xy > 0 \text{ і } x > 0; \\ \frac{y}{x^4 + 1}, & \text{в інших випадках.} \end{cases}$	a, y, x - довільні значення
21	$a = \begin{cases} b^3, & \text{якщо } b > 0 \text{ і } c > 0 \text{ або } b > 10; \\ b + c, & \text{якщо } b < 0 \text{ і } c = 0; \\ \frac{1}{(bc)^2 + 5}, & \text{якщо } b = 0 \text{ або } -5 < c < -1; \\ \frac{1}{(b+1)^2 + 3}, & \text{в інших випадках.} \end{cases}$	c - довільні значення
22	$y = \begin{cases} a^3 + x, & \text{якщо } a < x < 10 \text{ і } a < 10; \\ a + \sin(x), & \text{якщо } 10 < x \leq 15 \text{ і } x = a; \\ \ln ax , & \text{якщо } x = 20 \text{ або } 25 < a \leq 30; \\ e^{x-a}, & \text{в інших випадках.} \end{cases}$	a, x - довільні значення

$$23 \quad p = \begin{cases} 1+z, & \text{якщо } z=0 \text{ і } y=0; \\ z+y, & \text{якщо } zy < 0; \\ 25 & \text{в інших випадках} \end{cases} ; z, y - \text{довільні значення}$$

$$24 \quad y = \begin{cases} a+d+c, & \text{якщо } a+d > c; \\ (a+d)-c, & \text{якщо } a+d < c \text{ і } c > 5; \\ |\sin(d)|, & \text{якщо } a+d = c; \\ a^{dc} & \text{в інших випадках} \end{cases} . \quad a, d, c - \text{довільні значення}$$

$$25 \quad z = \begin{cases} y, & \text{якщо } y < 0; \\ 2y, & \text{якщо } 0 \leq y < 5; \\ 3y & \text{якщо } 5 \leq y < 10; \\ 4y, & \text{якщо } y \geq 10. \end{cases} \quad y - \text{довільне значення}$$

3.4.3 Циклічні обчислювальні процеси

3.4.3.1. Регулярні цикли

Табулювання функцій однієї та декількох змінних

Завдання: Протабулювати функцію і визначити суму обчислених значень

Варіант	Функція	Вихідні дані				
		a	b	xn	xk	h
1	2	3	4	5	6	7
01	$Y = \frac{\arctg bx}{1 + \sin^2 x}$	-		1.35	6.5	0.8
02	$Y = 5 \sqrt{\frac{a+bx}{\ln^2 x}}$	19.6	7.8	14.6	34.8	6
03	$Y = \frac{a \ln^2 x}{b + \sqrt{x}}$	1.38	-1.26	60	100	10
04	$Y = \frac{\sin^2 x}{\sqrt{x} + bx}$	-	1.68	1.2	2.4	0.2
05	$Y = \frac{\ln^2(x-b)}{a\sqrt{x}}$	0.36	5.5	10	50	6
06	$Y = \frac{e^{ax} + b}{1 + \cos^2 x}$	0.9	1.85	0	1.2	0.15
07	$Y = \frac{a + \sqrt[3]{x}}{\sin^2 bx}$	1.24	0.67	10.2	12.4	0.45
08	$Y = \frac{a\sqrt{x} - bx}{\ln^3 x}$	2.8	0.45	40	60	4.5
09	$Y = \frac{\sqrt{ax-b}}{\lg^3 x}$	20.2	7.65	3.5	4	0.1
10	$Y = e^{-x^2} \frac{a+bx}{\ln^2 x}$	4.6	2.5	0.75	1.8	0.3
11	$Y = \frac{\lg^2 ax - b}{e^{ax}}$	0.55	0.78	4.2	5.8	0.25
12	$Y = \frac{\arctg bx}{1 + \sqrt[5]{ax}}$	7.38	0.3	9	12	0.35

Продовження табл.

Варіант	Функція	Вихідні дані				
		a	b	xп	xк	h
1	2	3	4	5	6	7
13	$Y = \frac{\sin^3 ax}{ax + b}$	0.28	1.35	1.2	7.5	0.5
14	$Y = \frac{e^{-bx}}{b + \cos^3 ax}$	0.9	0.66	2.3	8.9	1.3
15	$Y = \frac{\ln^2 \sqrt{x}}{a\sqrt{x}}$	0.85	-	17.2	24.6	2
16	$Y = \frac{\arctg(a^3 + x^3)}{\sqrt{a^3 + x^3}}$	1.16	-	0.25	1.28	0.33
17	$Y = \frac{1 + \sqrt{bx}}{0.5 + \sin^2 ax}$	0.4	10.8	0.84	1.25	0.15
18	$Y = \frac{a - e^{bx}}{\ln^2 x}$	1.28	0.03	12.6	34.9	7.6
19	$Y = \frac{(a + bx)^{2.5}}{1.8 + \cos^3 ax}$	0.25	0.68	11.6	15.8	0.6
20	$Y = \frac{b + \sin^2 ax}{e^{-x/2}}$	1.6	1.24	0.2	1.4	0.35
21	$Y = \frac{\sin^2 x - a}{bx}$	1.8	0.34	6.44	9.1	0.25
22	$Y = \frac{atg^2 x}{b + 0.7x}$	0.44	2.28	6.5	7.3	0.12
23	$Y = \frac{\ln(a^2 - x)}{b \sin^2 x}$	3.2	0.45	0.6	1.5	0.2
24	$Y = \frac{a - \sqrt{bx}}{1 + \cos^2 x}$	17.6	10.45	1.9	3.8	0.3
25	$Y = \frac{\ln^2(a + x)}{(a + x)^2}$	8.24	-	14.9	24.8	1.5

3.4.4. Обробка списків та масивів

3.4.4.1. Пошук та вибір елементів масивів

Завдання

Масив X містить N елементів. Скласти схему алгоритму та програму введення значень елементів масиву та:

1. Визначення номерів та друкування значень елементів X(i) з ключовою відзнакою X(i)>22 або X(i)<5.
2. Визначення кількості елементів з ключовою відзнакою X(i) > -12 і X(i) < 3.
3. Визначення суми елементів з ключовою відзнакою X(i) > 25 або X(i) < -3.
4. Визначення добутку елементів з ключовою відзнакою 12 ≤ X(i) ≤ 33.
5. Визначення середнього арифметичного елементів масиву з ключовою відзнакою -2 ≤ X(i) ≤ 13.
6. Визначення різниці між добутками від'ємних та додатних елементів.
7. Визначення максимального елемента.
8. Визначення мінімального елемента.
9. Визначення кількості елементів, які більші середнього арифметичного.

10. Визначення кількості елементів, які менші середнього арифметичного.
11. Перевірки упорядкованості масиву по збільшенню.
12. Перевірки упорядкованості масиву по зменшенню.
13. Визначення різниці між максимальним та мінімальним елементами.
14. Визначення суми елементів з ключовою відзнакою $X(i) > 56$ або $X(i) < -13$ розташованих на парних місцях.
15. Визначення добутку елементів з ключовою відзнакою $1 \leq X(i) \leq 23$ розташованих на непарних місцях.
16. Визначення середнього арифметичного елементів масиву розташованих на парних місцях.
17. Визначення максимального елемента з непарним індексом.
18. Визначення максимального елемента з парним індексом.
19. Визначення кількості елементів з непарними індексами, більшими середнього арифметичного усіх елементів.
20. Визначення кількості елементів з парними індексами, меншими середнього арифметичного усіх елементів.
21. Визначення суми елементів з ключовою відзнакою $-3,14 \leq X(i) \leq 6,28$, розташованих на непарних місцях.
22. Визначення добутку елементів з ключовою відзнакою $\sqrt[5]{3} \leq X(i) \leq \frac{\sqrt{3}}{2}$, розташованих на парних місцях.
23. Формування масиву Z з елементів X(i), ключова відзнака яких $X(i) > 25$ або $X(i) < -3$ або $X(i) = 0$.
24. Формування масиву Y з елементів X(i), ключова відзнака яких $1 < X(i) < 125$, розташованих на парних місцях.
25. Формування масиву D з елементів X(i), ключова відзнака яких $X(i) < 2$ і $X(i) > -20$ або $X(i) = 5$.

3.4.4.2. Формування масиву

Завдання

1. Задані масиви $\{x_i\}, \{y_i\}, i = 1, 10$ та довільні змінні a, b
Сформувані масив $\{c_i\}, i = 1, 10$ та надрукувати його елементи

$$c_i = \frac{a \sin x_i + b \cos y_i}{x_i^2 + 1}$$

2. Заданий масив $\{x_i\}, i = 1, n$, де n- довільна змінна
Сформувані масив $\{d_i\}, i = 1, n$ та надрукувати його елементи

$$d_i = \begin{cases} x_i^2, & \text{якщо } x_i < 0 \\ |x_i|, & \text{якщо } x_i \geq 0 \end{cases}$$

3. Заданий масив $\{z_k\}, k = 1, 15$
Сформувані масив $\{d_k\}, k = 1, 15$ та надрукувати його елементи

$$d_k = \begin{cases} z_k + 1, & \text{якщо } z_k < 20 \\ \ln z_k^5, & \text{якщо } z_k > 20 \\ z_k^2, & \text{якщо } z_k = 20 \end{cases}$$

4. Задані масиви $\{x_i\}, \{a_i\}, i = 1, 12$
Сформувані масив $\{f_i\}, i = 1, 12$ та надрукувати його елементи

$$f_i = \ln x_i + 2 \sum_{j=1}^8 a_j$$

5. Сформувані масив $\{z_j\}$ з додатніх елементів масиву $\{y_k\}, k = 1, 15$

6. Задані масиви $\{a_i\}, i = 1, 10, \{b_j\}, j = 1, 12$

Сформувати масив $\{c_i\}, i = 1, 10$ та надрукувати його елементи

$$c_i = \frac{a_i + \prod_{k=1}^7 b_k}{\sum_{k=1}^9 b_k}$$

7. Задані масиви $\{x_i\}, \{y_i\}, i = 1, k$, де k - довільна змінна

Сформувати масив $\{z_i\}, i = 1, k$ та надрукувати його елементи

$$z_i = \frac{\sin x_i + \cos y_i}{x_i^2 + y_i^2}$$

8. Заданий масив $\{t_i\}, i = 1, n$, де n - довільна змінна

Сформувати масив $\{r_i\}, i = 1, n$ та надрукувати його елементи

$$r_i = \begin{cases} t_i^{32}, & \text{якщо } t_i < 10 \\ |t_i + 1|, & \text{якщо } t_i \geq 10 \end{cases}$$

9. Заданий масив $\{s_k\}, k = 1, 18$

Сформувати масив $\{q_k\}, k = 1, 18$ та надрукувати його елементи

$$q_k = \begin{cases} s_k^5 + 1, & \text{якщо } s_k < 100 \\ \lg s_k^5, & \text{якщо } s_k > 100 \\ s_k^2, & \text{якщо } s_k = 100 \end{cases}$$

10. Задані масиви $\{x_i\}, \{a_i\}, i = 1, 12$

Сформувати масив $\{f_i\}, i = 1, 12$ та надрукувати його елементи

$$f_i = \ln a_i + 2 \sum_{j=1}^8 x_j$$

11. Сформувати масив $\{x_j\}$ з від'ємних елементів масиву $\{y_k\}, k = 1, 25$

12. Задані масиви $\{a_i\}, i = 1, n, \{b_j\}, j = 1, m$ n, m - довільні змінні.

Сформувати масив $\{c_i\}, i = 1, m$ та надрукувати його елементи

$$c_i = \frac{b_i + \prod_{k=1}^n a_k}{\sum_{k=1}^n a_k}$$

13. Заданий масив $\{z_k\}, k = 1, 15$

Сформувати масив $\{d_k\}, k = 1, 15$ та надрукувати його елементи

$$d_k = \begin{cases} z_k + \prod_{j=1}^{10} z_j, & \text{якщо } z_k < 20 \\ \ln z_k^5, & \text{якщо } z_k > 20 \\ z_k^2 - \sum_{j=3}^8 z_j, & \text{якщо } z_k = 20 \end{cases}$$

14. Заданий масив $\{x_i\}, i = 1, n$, де n - довільна змінна

Сформувати масив $\{d_i\}, i = 1, n$ та надрукувати його елементи

$$d_i = \begin{cases} x_i^2 + \sum_{k=2}^5 x_k, & \text{якщо } x_i < 0 \\ |x_i| - \prod_{j=1}^4 x_j, & \text{якщо } x_i \geq 0 \end{cases}$$

15. Задані масиви $\{a_i\}, i = 1, 12, \{b_j\}, j = 1, 15, \{c_k\}, k = 1, 8, \{x_n\}, n = 1, 7$

Сформувати масив $\{f_i\}, i = 1, 12$ та надрукувати його елементи

$$f_i = \frac{\sum_{k=1}^{11} b_k + \prod_{k=2}^6 x_k c_k}{a_i}$$

16. Заданий масив $\{t_i\}, i = 1, n$, де n - довільна змінна

Сформувати масив $\{r_i\}, i = 1, n$ та надрукувати його елементи

$$r_i = \begin{cases} t_i^3, & \text{якщо } \sum_{k=1}^6 t_k < 10 \\ |t+1_i|, & \text{якщо } \sum_{k=1}^6 t_k \geq 10 \end{cases}$$

17. Заданий масив $\{x_i\}, i = 1, m$, де m - довільна змінна

Сформувати масив $\{y_i\}, i = 1, m$ та надрукувати його елементи

$$y_i = \begin{cases} \prod_{j=1}^8 x_j^2 + \sum_{k=2}^5 x_k - x_i, & \text{якщо } x_i < 3 \\ |x_i| - \prod_{j=1}^4 x_j - \sum_{k=3}^6 x_k, & \text{якщо } x_i \geq 3 \end{cases}$$

18. Сформувати масив $\{x_j\}$ з від'ємних і масив $\{z_i\}$ з додатних елементів масиву $\{y_k\}, k = 1, 25$

19. Задані масиви $\{x_i\}, \{y_i\}, i = 1, 10$ та довільні змінні a, b

Сформувати масив $\{c_i\}, i = 1, 10$ та надрукувати його елементи

$$c_i = \frac{a \sin \prod_{k=2}^7 x_k + b \cos \sum_{n=1}^5 y_n}{x_i^2 + 1}$$

20. Заданий масив $\{s_k\}, k = 1, 18$

Сформувати масив $\{q_k\}, k = 1, 18$ та надрукувати його елементи

$$q_k = \begin{cases} s_k^5 + 1 + \sum_{i=1}^5 s_i, & \text{якщо } s_k < 200 \\ \lg s_k^5 + \sum_{i=2}^7 s_i, & \text{якщо } s_k > 200 \\ s_k^2 - \prod_{i=3}^{10} s_i, & \text{якщо } s_k = 200 \end{cases}$$

21. Задані масиви $\{b_i\}, \{a_i\}, i = 1, 20$

Сформувати масив $\{c_i\}, i = 1, 20$ та надрукувати його елементи

$$c_i = \lg a_i + 2 \sum_{j=1}^8 x_j a_j$$

22. Заданий масив $\{z_k\}, k = 1, n$, n – довільна змінна

Сформувати масив $\{d_k\}, k = 1, n$ та надрукувати його елементи

$$d_k = \begin{cases} z_k + \prod_{j=2}^6 z_j, & \text{якщо } z_k < 0 \\ \ln z_k^5, & \text{якщо } z_k > 0 \\ z_k^2 - \sum_{j=1}^{10} z_j, & \text{якщо } z_k = 0 \end{cases}$$

23. Задані масиви $\{a_i\}, i = 1, 2, \dots, 7$, $\{b_j\}, j = 1, 5$, $\{c_k\}, k = 1, 8$, $\{x_n\}, n = 1, 7$

Сформувати масив $\{f_j\}, j = 1, 5$ та надрукувати його елементи

$$f_j = \frac{\sum_{k=1}^7 a_k + \prod_{k=1}^7 x_k c_k}{b_j}$$

24. Задані масиви $\{x_i\}, \{y_i\}, i = 1, 20$

Сформувати масив $\{z_i\}, i = 1, 20$ та надрукувати його елементи

$$z_i = \begin{cases} x_i - \prod_{j=1}^5 y_j, & \text{якщо } \sum_{k=1}^5 x_k \leq \prod_{j=1}^5 y_j \\ \sum_{k=1}^5 x_k + y_i, & \text{якщо } \sum_{k=1}^5 x_k > \prod_{j=1}^5 y_j \end{cases}$$

25. Сформувати масив $\{x_j\}$ з від'ємних, масив $\{z_i\}$ з додатніх і масив $\{a_n\}$ з нульових елементів масиву $\{y_k\}, k = 1, 25$

3.4.4.3 Сортювання масивів

Завдання:

Задані списки (масиви) числових значень з довільною кількістю елементів.

Згідно з умовою задачі виконати над даними списками операції та вивести списки на друк до і після операцій.

1. Перевірити на рівність списки L1 і L2
2. Визначити, чи входить список L1 до L2.
3. Перевірити, є хоча б в одному списку L1 чи L2 однакові елементи;
4. Додати в кінець списку L1 чи L2 новий елемент.
5. Додати в кінець списку L1 всі елементи списку L2;
6. Вставити в список L1 за першим входженням заданого елемента всі елементи списку L2.
7. "Перевернути" список L1 - тобто змінити порядок проходження в ньому елементів на протилежний;
8. У списку L2 виконати упорядкування за зростанням.
9. Додати в кінець списку L1 всі елементи списку L2.
10. Перевірити на нерівність списки L1 і L2;
11. Визначити, чи входить список M1 у M2 і навпаки.
12. Додати в початок списку L1 чи L2 новий елемент.
13. Перевірити, чи є в списку однакові елементи;
14. Додати в початок списку L1 всі елементи списку L2.
15. У списку L2 виконати упорядкування за зростанням, а у списку L1 - за убутанням.
16. У списку L виконати упорядкування за зростанням та за убутанням.
17. Додати новий елемент в упорядкований за зростанням список на відповідне місце.
18. Підрахувати кількість однакових елементів списку
19. Підрахувати кількість однакових елементів списків L1 та L2
20. Підрахувати, скільки разів кожний елемент входить у список
21. Додати в кінець списку L1 елементи списку L2, які більші заданого елемента;
22. Додати в початок списку L1 елементи списку L2, які більші заданого елемента
23. Додати в кінець списку L1 всі елементи списку L2, а в кінець списку L2 всі елементи списку L1;
24. Додати новий елемент в упорядкований за убутанням список на відповідне місце
25. Додати в початок списку L1 всі елементи списку L2

3.4.4.4 Двовимірні масиви

Завдання

Заданий двовимірний масив числових значень з довільною кількістю рядків і стовпців. Згідно з завданням виконати обробку даних у цьому масиві:

- 1) обчислити середнє арифметичне значення від'ємних елементів у кожному рядку;
- 2) визначити суму від'ємних значень у кожному стовпці.
- 3) обчислити середнє арифметичне значення в кожному рядку з парним номером
- 4) визначити кількість та суму від'ємних значень у кожному стовпці.
- 5) знайти мінімальне значення на побічній діагоналі масиву;
- 6) обчислити середнє арифметичне додатних елементів, розташованих під головною діагоналлю масиву.
- 7) знайти максимальне додатне значення на головній діагоналі масиву;
- 8) обчислити добуток додатних елементів, розташованих під головною діагоналлю масиву.
- 9) визначити число елементів першого рядка, що за значенням менше, ніж середнє арифметичне в цьому ж рядку;
- 10) знайти мінімальне і максимальне значення на побічній діагоналі масиву.
- 11) визначити число елементів кожного рядка, що за значенням більше, ніж середнє арифметичне в цьому ж рядку;
- 12) знайти мінімальне значення на головній діагоналі масиву.
- 13) обчислити середнє арифметичне значення додатних елементів у кожному рядку;
- 14) обчислити середнє арифметичне від'ємних елементів, розташованих над головною діагоналлю масиву.
- 15) обчислити середнє арифметичне додатних елементів, розташованих над головною діагоналлю масиву.
- 16) обчислити і надрукувати добуток кожного рядка;
- 17) обчислити і надрукувати суму кожного рядка;
- 18) знайти і надрукувати середнє значення кожного рядка;
- 19) знайти різницю між мінімальним і максимальним значенням елементів на побічній діагоналі масиву.
- 20) знайти різницю між мінімальним і максимальним значенням елементів масиву.
- 21) підрахувати кількість додатних та від'ємних елементів масиву.
- 22) обчислити і надрукувати добуток в непарних рядках;
- 23) обчислити і надрукувати кількість нульових елементів у парних рядках;
- 24) обчислити і надрукувати кількість ненульових елементів у непарних стовпцях;
- 25) знайти різницю між середнім арифметичним елементів у парних та непарних стовпцях.

Літературні джерела

1. Информатика. Базовый курс / Под ред. С.В. Симоновича - СПб: Издательство «Питер», 2000. – 640 с.: ил.
2. Васильев А., Андреев А. VBA в Office 2000: учебный курс. – СПб.: Питер, 2001.– 432 с.: ил.
3. Гарнаев А.Ю. Excel, VBA, Internet в экономике и финансах. – СПб.: БХВ-Петербург, 2001. – 816 с.: ил.
4. Гарнаев А.Ю. MS Excel 2002: разработка приложений. – СПб.: БХВ-Петербург, 2003. – 769 с.: ил.
5. Гарнаев А.Ю. Использование MS Excel и VBA в экономике и финансах. – СПб.: БХВ-Санкт-Петербург, 1999. – 336 с.: ил.
6. Долженков В.А., Колесников Ю.В. Microsoft Excel 2000. – СПб.: БХВ-Петербург, 1999. – 1088 с.: ил.
7. М. Коттингхэм Excel 2000: руководство разработчика: Пер. с англ. – К.: Издательская группа BHV, 2000. – 704 с.
8. Матросов А.В. и др. MS Office XP: разработка приложений / Матросов А.В., Новиков Ф.А., Усаров Г.Е., Харитонов И.А. / Под ред. Ф.А. Новикова. – СПб.: БХВ-Петербург, 2003. – 944 с.: ил.
9. Саймон, Джинжер. Программирование в Excel: наглядный курс создания интерактивных электронных таблиц. : Пер. с англ. – М. : Издательский дом «Вильямс», 2002. – 336 с.: ил.
10. Самоучитель программирования на VBA в Microsoft Office / Хорев В.Д. – К.: Юниор, 2001. – 296 с., ил.
11. Свиридова М.Ю. Создание презентации в Power Point – М.: Академия, 2010. – 224 с.: ил.
12. Сетевые операционные системы / В.Г. Олифер, Н.А. Олифер. – СПб.: Питер. 2002. – 544 с.: ил.
13. Союз освітніх сайтів – <http://allbest.ru/union/>
14. Эндрю Питоньяк. Open Office.org pro. Автоматизация работы – ООО "Книга по Требованию", 2009. – 496 с.
15. Харрис, Метью. Освой самостоятельно программирование для Microsoft Excel 2000 за 21 день.: Пер. с англ. : Уч. пос. – М.: Издательский дом “Вильямс”, 2000. – 880 с.: ил.
16. Методические указания для выполнения лабораторных работ по курсу «Экономическая информатика» (для студентов 1, 2 курсов дневной формы обучения бакалавров направления 6.030504 - «Экономика предприятий», 6.030509 – «Учет и аудит»). Сост. Б.И. Погребняк, А.В. Белогурова, Е.В Кузьмичева, Н.О. Манакова. – Харьков: ХНАГХ, 2008. – 86 с.
17. Дядюн С.В., Костенко О. Б., Меркулов В. С., Філіпенко І. Г., Шумеев В.В. Збірник задач та прикладів з обчислювальної техніки та програмування. Навчальний посібник, Х.: ХНАМГ, 2001. - 56с.

Навчальне видання

Методичні вказівки
до лабораторних, практичних,
самостійних та контрольних робіт
з дисципліни

«Інформатика та комп'ютерна техніка»

*(для студентів I курсу денної та заочної форм навчання за напрямом
підготовки 6.170202 «Охорона праці»)*

Укладачі **ДЯДІОН** Сергій Васильович
ПАН Микола Павлович
БЛОГУРОВА Ганна Вікторівна

Відповідальний за випуск *О. Б. Костенко*

Редактор *С. В. Тимошук*

Комп'ютерне верстання *К. А. Алексанян*

План 2012, поз. 432М

Підп. до друку 06.07.2012р.

Друк на ризографі.

Зам. №

Формат 60x84/16

Ум. друк. арк. 6,6

Тираж 50 пр.

Видавець і виготовлювач:
Харківський національний університет
міського господарства імені О. М. Бекетова,
вул. Революції, 12, Харків, 61002
Електронна адреса: rectorat@ksame.kharkov.ua
Свідоцтво суб'єкта видавничої справи:
ДК № 4064 від 12.05.2011р.