

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

МЕТОДИЧНІ ВКАЗІВКИ
ДО ВИКОНАННЯ РОЗРАХУНКОВО-ГРАФІЧНОЇ РОБОТИ
з дисципліни

«ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ В БУДІВНИЦТВІ
ТА БАЗИ ДАНИХ»

*(для студентів 3 курсу денної форми навчання за напрямом
6.060101 – Будівництво спеціальності «Промислове та цивільне
будівництво»)*

Методичні вказівки до виконання розрахунково-графічної роботи з дисципліни «Інформаційні технології в будівництві та бази даних» (для студентів 3 курсу денної форми навчання за напрямом 6.060101 – Будівництво спеціальності «Промислове та цивільне будівництво») / Харків. нац. ун-т міськ. госп-ва ім. О. М. Бекетова; уклад. : Н. О. Манакова, Н. В. Макогон. – Харків : ХНУМГ ім. О. М. Бекетова, 2016. – 40 с.

Укладачі: Н. О. Манакова,

Н. В. Макогон

Методичні вказівки складені відповідно до вимог кредитно-модульної системи організації навчального процесу та узгоджені з орієнтовною структурою змісту навчальної дисципліни, рекомендованою Європейською кредитно-трансферною Системою (ECTS).

Рецензент: канд. фіз.-мат. наук, доц. О. Б. Костенко

Рекомендовано кафедрою прикладної математики і інформаційних технологій протокол № 1 від 31.08.2015 р.

ЗМІСТ

| | |
|---|----|
| ВСТУП..... | 4 |
| 1.ТЕМАТИКА ЗАВДАНЬ..... | 5 |
| 2.ПЛАНУВАННЯ РОЗРОБОК..... | 5 |
| 2.1. Етапи виконання роботи..... | 5 |
| 2.2. Графік виконання роботи..... | 6 |
| 3.СТРУКТУРА І ЗМІСТ РОБІТ..... | 6 |
| 3.1. Структура записки пояснення..... | 6 |
| 3.2. Оформлення основних розділів..... | 7 |
| 4.ФОРМА КОНТРОЛЮ І ЗВІТНОСТІ..... | 12 |
| 4.1. Поточний проміжний контроль..... | 12 |
| 4.2. Захист курсової роботи..... | 13 |
| 5.ЗАВДАННЯ НА ПРОГРАМУВАННЯ..... | 13 |
| 5.1. Алгоритм циклічного процесу. Масиви..... | 13 |
| 5.2. Вкладені цикли. Двовимірні числові масиви..... | 28 |
| СПИСОК ЛІТЕРАТУРИ..... | 40 |

ВСТУП

Головними цілями виконання навчальної роботи з дисципліни «Інформаційні технології в будівництві та бази даних» є закріплення, поглиблення і узагальнення студентами знань, отриманих під час навчання, і вироблення у них уміння самостійно застосовувати ці знання для творчого вирішення конкретних практичних завдань.

При виконанні роботи студент підтверджує володіння такими основними навиками і прийомами, як:

- рішення і алгоритмізація обчислювальних завдань, що зустрічаються в інженерній практиці;

- програмування на мові високого рівня з урахуванням структурного підходу;

- створення простих додатків в середовищі візуального програмування;

- оформлення текстової документації супроводу інформаційних розробок і програмних продуктів.

Курсова робота виконується з використанням програмних засобів офісного призначення і візуального програмування.

Завдання курсової роботи – створення програмного додатку прикладного характеру, виконаного за індивідуальним завданням.

1. ТЕМАТИКА ЗАВДАНЬ

Індивідуальні завдання на розробку програмного продукту повинні:

–відповідати рівню підготовки студентів і ліміту часу, відведеному на виконання роботи;

–містити завдання, вирішення яких базується на обчислювальних алгоритмах і структурному програмуванні;

–допускати можливість варіювання значень початкових даних, що відбивається на організації інтерфейсу користувача;

–допускати можливість творчого підходу до дизайну інтерфейсу додатку;

–передбачати необхідність літературного огляду додаткових джерел інформації, що характеризують сучасний розвиток інформаційних технологій;

–мати орієнтацію на освоєння персонального комп'ютера при вирішенні завдань, що часто зустрічаються в інженерній практиці.

2. ПЛАНУВАННЯ РОЗРОБОК

2.1. Етапи виконання роботи

Робота над індивідуальним завданням включає виконання декількох етапів. Етапи мають певну трудомісткість. У таблиці представлена орієнтовна трудомісткість (час на виконання) окремих етапів роботи у відсотках від всього об'єму.

| № етапу | Найменування етапу | Трудо- місткість | Наростаю- чий об'єм |
|---------|---|---------------------|------------------------|
| 1. | Складання календарного плану | 5 % | 5 % |
| 2. | Реферат | 10 % | 15 % |
| 3. | Розробка структури презентації і створення її базових елементів | 10 % | 25 % |
| 4. | Доведення презентації і оформлення її мультимедійними ефектами | 10 % | 35 % |
| 5. | Аналіз умови завдання, її математичне рішення | 10 % | 45 % |

| | | | |
|-----|---|------|-------|
| 6. | Складання алгоритму рішення | 10 % | 55 % |
| 7. | Розробка структури програмного застосування, опис змінних | 10 % | 65 % |
| 8. | Програмна реалізація розрахункової частини | 10 % | 77 % |
| 9. | Налагодження, компіляція і тестування додатку | 10 % | 95 % |
| 10. | Оформлення звітної документації | 5 % | 90 % |
| 11. | Захист курсової роботи | 5 % | 100 % |

2.2. Графік виконання роботи

Плановим календарним початком виконання роботи є дата видачі завдань на підгрупу.

Плановим календарним терміном виконання 10-го етапу може бути дата не пізніше за перший день залікового тижня семестру (більш ранні терміни виконання вітаються).

Плановані календарні терміни виконання проміжних етапів повинні бути розміщені пропорційно їх об'єму в межах початкової і кінцевої дати.

Календарний план оформляється документально і затверджується викладачем на початку семестру (див. далі структуру робіт).

3. СТРУКТУРА І ЗМІСТ РОБІТ

3.1. Структура записки пояснення

Записка пояснення є невід'ємним документом курсової роботи. Документ записки пояснення повинен бути як в електронному вигляді (файл), так і в паперовому (відомості в обох формах повинні бути ідентичні).

Всі роботи відбиваються в записці пояснення. Не дивлячись на те, що самі роботи можуть відбуватися в різний час, розділи записки пояснення слідує в строго певному порядку. Не допускається відсутність яких-небудь розділів.

Записка пояснення включає *логічні частини першого рівня:*

ТИТУЛЬНИЙ АРКУШ
БЛАНК ЗАВДАННЯ
ЗМІСТ
ВСТУП
РОЗДІЛ 1. РЕФЕРАТ НА ТЕМУ «...»
РОЗДІЛ 2. БАЗОВИЙ ПЛАН ПРЕЗЕНТАЦІЇ.
РОЗДІЛ 3. ХАРАКТЕРИСТИКА ПРОЦЕСУ СТВОРЕННЯ
ПРЕЗЕНТАЦІЇ
РОЗДІЛ 4. АЛГОРИТМІЗАЦІЯ ОБЧИСЛЮВАЛЬНИХ ПРОЦЕСІВ
РОЗДІЛ 5. АНАЛІЗ ДАНИХ
РОЗДІЛ 6. СТРУКТУРА ПРОГРАМНОГО ЗАСТОСУВАННЯ
РОЗДІЛ 7. ТЕСТУВАННЯ
ВИСНОВОК
ЛІТЕРАТУРА

Структура підрозділів другого рівня визначається студентом самостійно.

3.2. Оформлення основних розділів

Оформлення титульного аркуша

Титульний аркуш повинен включати наступні реквізити:

- головне міністерство або відомство
- найменування ВНЗ
- найменування факультету
- найменування кафедри, на якій виконується робота
- найменування виду роботи
- найменування навчальної дисципліни
- авторські дані (*Прізвище І.Б. студента, номер групи*)
- інформація про керівника (*Прізвище І.Б. викладача*)
- відмітка про результат захисту (проставляється викладачем)
- місто і рік виконання роботи.

Титульний аркуш **1 стор.** Номер сторінки на титульному аркуші не проставляється

Оформлення бланку завдання

Приводиться текст формулювання завдання. Перераховуються підзадачі, що відображають етапи виконання роботи. Приводиться тема реферату.

Завдання включає необхідний календарний план виконання роботи, а також результати темпів реального виконання роботи.

Завдання оформляється студентом і затверджується викладачем на початку виконання роботи (1 тиждень після видачі завдання). Результати поточного виконання роботи візуються викладачем і є основою для поточного контролю і атестації студентів.

При виконанні менше 30 % роботи до календарного терміну планового захисту індивідуальне завдання підлягає заміні.

Об'єм завдання 1 стор.

Оформлення змісту

Зміст повинен відображати структуру записки пояснення не більше двох рівнів (розділи, підрозділи). Зміст повинен бути генерований автоматичними засобами текстового редактора. Нагадаємо, що автоматична генерація змісту можлива за умови правильного форматування всього тексту.

Об'єм змісту не більше 1 стор.

Оформлення вступу

Вступ повинен відображати мету і завдання виконання конкретної роботи. Приводиться аналіз зв'язку тематики роботи з суміжними природничо-науковими і технічними дисциплінами і галузями знань, можливі області застосування розробки. Указуються основні прийоми, інструменти, програмні засоби рішення задачі.

Об'єм вступу 0,5-1 стор.

Оформлення Розділу 1. Реферат на тему «...»

В даному випадку реферат – це не самостійний документ, а тільки розділ в цілому документі записки пояснення. Тому його форматування повинно відповідати стилю всієї записки пояснення.

Даний розділ включає літературний огляд і реферування (опис ключових питань) матеріалів однієї з тем, пов'язаних з сучасним станом інформаційних технологій. Дана робота, розширюючи кругозір студента,

направлена на придбання навиків структуризації описового матеріалу із заданої тематики, виділення головної і допоміжної, істотної і неістотної інформації.

У рефераті на задану тему, як правило, повинно бути відбито:

- визначення поняття, винесеного в заголовок теми;
- позначені передумови виникнення, цілі існування, області застосування аналізованих об'єктів, понять;
- перераховані основні характеристики, ознаки, властивості даного поняття взагалі;
- приведені різновиди реалізації об'єкту аналізу і основні характеристики, ознаки, властивості різновидів виконання зокрема, що показують достоїнства і недоліки;
- приведені рекомендації по використанню;
- приведені перспективи розвитку даного поняття.

Тематика повинна бути розкрита комплексно. Для скорочення об'єму тексту раціонально використовувати структуру конспекту, табличне представлення порівняльної інформації і т.п. Одностороннє освітлення теми не є рефератом і є серйозним недоліком роботи.

Літературні джерела інформації повинні бути приведені в списку використаної літератури. Джерела інформації, отримані через Internet повинні характеризуватися повною адресою доступу.

Увага! Студентські реферати, поміщені в поширених збірках, як правило, носять істотні недоліки:

- одностороннє освітлення теми (по суті – доповідь, а не реферат), що приводить до необхідності проглянути велику кількість таких рефератів;
- не сучасні відомості;
- безграмотне форматування тексту.

Індивідуальна тема реферату вибирається студентом і затверджується керівником разом з календарним планом завдання спочатку семестру.

Об'єм реферату 3-5 стор.

Оформлення Розділу 2. Базовий план презентації

Приводиться опис презентації згідно написаного реферату. Інформація повинна бути *необхідною і достатньою для освоєння теми стороннім користувачем*. Опис плану включає наступну інформацію:

–тему реферату;
–межі обхвату теми по вертикалі і горизонталі;
–опис послідовності викладу із структуризацією по підрозділах, а також з вказівкою номерів і кількості слайдів що ілюструє кожен підрозділ;

–загальна характеристика презентації (кількість слайдів, малюнків, таблиць, об'єм займаний файлів без анімації і з нею.).

Об'єм даного розділу записки пояснення 2-4 стор.

Оформлення Розділу 3. Характеристика процесу створення презентації

У даному розділі приводиться опис дій, які були виконані при створенні презентації:

- як і де розміщувалися малюнки, таблиці, графіки
- які елементи управління будуть доступні користувачеві
- як і де виникатимуть анімаційні і звукові ефекти

Приводиться ілюстрація зразкового зовнішнього вигляду презентації.

Об'єм даного розділу записки пояснення 1 стор.

Оформлення Розділу 4. Алгоритмізація обчислювальних процесів

У записці пояснення приводиться блок-схема алгоритму рішення задачі у вигляді графічного об'єкту. Алгоритм повинен відповідати вимогам і правилам алгоритмізації. Команди в блоках алгоритму повинні відповідати мові системи команд вирішувача. Рисунок алгоритму повинен відповідати правилам створення графічних об'єктів в текстовому редакторові.

Блок-схема представляє графічний запис алгоритму програми у вигляді послідовності геометричних фігур-блоків. Блок відображає деякий крок рішення задачі і містить пояснення дій на цьому кроці. Кожній формі блоку відповідає певне призначення. Блоки зв'язані між собою стрілками, які вказують послідовність виконання кроків алгоритму. Деякі певні зв'язки між блоками називають структурами. При побудові будь-якої програми використовують послідовну структуру, умовну структуру і структуру повторення. Форми і правила побудови схем визначені стандартом. Основні блоки і логічні структури, що управляють, приведені в навчальному посібнику лабораторного практикуму і в конспекті лекцій.

Об'єм даного розділу записки пояснення 1 стор.

Оформлення Розділу 5. Аналіз даних

Приводиться опис всіх змінних, які використовувалися при розробці програми з вказівкою їх імен, смислового змісту, типу даних, області видимості і так далі.

Необхідно обґрунтовано призначити і привести в записці всебічну характеристику для кожної змінної.

Об'єм даного розділу записки пояснення 1-2 стор.

Оформлення Розділу 6. Структура програмного застосування

У даному розділі приводяться зведення по реалізації програмного застосування в середовищі візуального програмування. При цьому використовуються методичні вказівки лабораторного практикуму.

Приводиться опис форм (якщо вони використовуються), модулів і так далі, їх призначення, зв'язки між ними.

Приводиться опис *об'єктів*, які застосовані при розробці додатку з вказівкою їх імен, значень їх важливих властивостей, призначених автором. Також приводиться опис *подій*, пов'язаних з цими елементами, що управляють. Приводиться опис *процедур*.

У даному розділі на окремій сторінці приводиться **лістинг** (роздрук) програмної коди рішення задачі без додаткових пояснень. При необхідності пояснення можна привести на інших сторінках записки.

Відлагоджений проект підлягає компіляції. Програмна розробка повинна містити авторські дані розробника.

Об'єм даного розділу записки пояснення 2-4 стор.

Оформлення Розділу 7. Тестування

Для підтвердження вірності розрахунків дане завдання необхідно реалізувати в іншому середовищі. Найшвидше це можна зробити в середовищі Excel. Необхідно переконатися в збігу розрахунків в різних реалізаціях при однакових початкових даних. Збережіть електронні документи тестування завдання в альтернативному середовищі для демонстрації на захисті.

У записці пояснення приводиться опис конкретних значень початкових даних, які застосовувалися для тестування, і відповідних

значень результату роботи додатку по обробці вибраних початкових даних і результатів виконання в альтернативному середовищі.

Представлення опису можливе в табличній формі і/або із застосуванням ілюстрацій.

Об'єм розділу записки пояснення 1-2 стор.

Оформлення висновку

У висновку даються короткі висновки про виконану роботу (по етапах її виконання), також дається коротка характеристика застосованих методів і використаного прикладного програмного забезпечення. Також слід вказати можливі подальші шляхи розвитку і розширення розробки, можливості розширення круга завдань і тому подібне.

Об'єм даного розділу записки пояснення 0,5-1 стор.

Оформлення списку літератури

Указується список літератури, використаної студентом при виконанні курсової роботи. При використанні матеріалів, які опубліковані в *Internet*, вказується адреса сайту, сторінки.

Допоміжна література для виконання курсової роботи підбирається студентом самостійно. Література включає довідкові матеріали по роботі з прикладним програмним забезпеченням офісного характеру, а також по роботі в конкретному середовищі програмування вибраної мови високого рівня.

Посилання на матеріали, за якими складений реферативний розділ, обов'язкові.

Об'єм даного розділу записки пояснення 0,5-1 стор.

4. ФОРМА КОНТРОЛЮ І ЗВІТНОСТІ

4.1. Поточний проміжний контроль

Робота виконується поетапно і підлягає проміжному контролю ходу виконання. Реальний хід виконання відбивається викладачем в бланку завдання і служить підставою для атестації студента.

Основою позитивної атестації служить своєчасне і правильне виконання робіт, завізоване в бланку.

Для проміжного контролю результати виконання етапу повинні бути оформлені на аркушах формату А4 (можна рукописний) і вважаються

чернетками. Програмні розробки здаються на дискеті підписаної автором. Матеріали для проміжного контролю комплектуються в теку або пластиковий файл, обов'язково прикладається бланк завдання. Чернетки з відмітками викладача бажано зберігати.

4.2. Захист курсової роботи

До захисту курсова робота представляється у вигляді записки пояснення на 20-25 сторінках (А4). У текст записки пояснення включаються ілюстрації презентації і друк програмної коди. Записка пояснення представляється до захисту **в зброшурованому вигляді** і з пронумерованими сторінками (титульний лист вважається першою сторінкою, але номер сторінки на ній не проставляється). Шрифт тексту 12 - 14 пт. Поля документа не менше 20 мм і не більше 25 мм з кожного боку. Кожен розділ записки пояснення починається з нової сторінки.

Обов'язковим є розміщення роботи, що включає програмні розробки, повну презентацію і записку пояснення, в дистанційному курсі.

Виконана робота підлягає захисту. За наслідками захисту проставляється оцінка.

При захисті студент демонструє тестування розробки; відповідає на питання, що стосуються математичного методу рішення задачі, алгоритмізації, методів і понять, використовуваних при програмній реалізації; освітлює тему реферату.

Час, що виділяється на захист роботи, – 15-20 хв.

5. ЗАВДАННЯ НА ПРОГРАМУВАННЯ

5.1. Алгоритм циклічного процесу. Масиви.

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 так далі. Масив дає можливість зберігати і обробляти всі свої елементи, використовуючи одну змінну. Крім того, що скорочується загальна кількість імен змінних, іншою істотною перевагою є те, що для обробки елементів в 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.

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.

Кожна «цеглинка» даних з масиву називається його елементом. У масиві на рисунку 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 і відомий як нумерація з нижньою межею — 0).

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 |

Рисунок 1 – Одновимірний масив з 10 елементів типу Double з нумерацією від 0 до 9

Щоб дістати доступ до окремого елемента масиву, вказується ім'я масиву, за яким слідує номер елемента, чий вміст треба узяти або змінити (цей номер називається індексом). Індекс завжди треба брати в дужки.

Наприклад, якщо масив на рисунку 1 називається NumArray, то в наступній інструкції змінної AnyNum привласнюється число 55,4.

$$\text{AnyNum} = \text{NumArray}(7)$$

У цій інструкції число 7 — це індекс масиву; зверніть увагу, що він поміщений в 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):

$$\text{NumArray}(7) = 12$$

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 індеси масивів потрібно використовувати цілі числа.
- Не відокремлюйте пропусками імена масивів і індеси: у програмі набирайте ім'я, дужки і індеси як одне слово.

Статичні і динамічні масиви.

Зазвичай 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 рисунку 1 її треба виділити стільки, щоб поміщалось 10 чисел типу Double.

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 витратиться даремно (різниця між найбільшою і найменшою кількістю елементів). Для такого роду масивів в 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 відокремлюються комою) і так далі, до тієї кількості розмірностей яке вам потрібно.

Вираз **Subscripts** має такий синтаксис:

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

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

Якщо у виразі **Subscripts** знаходиться *lower to*, то це значно полегшуватиме розуміння програм і знаходитиме в них помилки, що зробить програми зрозумілишими і надійнішими. Використання *lower To*

також дає можливість вказати початкове значення індексу, яке відрізняється 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 в VBA тип даних: Currency, Double, String і так далі. Якщо type буде пропущений, то у всіх елементів масиву буде тип Variant. При ініціалізації елементів числових масивів елементи набувають значення 0, а в строкових — значення порожнього рядка.

Завдання на обробку одновимірних масивів.

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 елементу початкового масиву, пересунувши його елементи;
- c)* Обчислити суму елементів отриманого масиву;
- d)* Вивести масив до i після зміни i значення суми.
4. *a)* Одновимірний масив з 8 елементів цілого типу в діапазоні [16, 40];
- b)* Переставити четвертий елемент масиву на останнє місце, пересунувши елементи масиву;
- c)* Обчислити суму елементів масиву, починаючи з п'ятого;
- d)* Вивести масив до i після зміни i 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, пересунувши 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 управо, 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 добуток елементів першого масиву;
d) Вивести два початкових, два отримані масиви і значення добутку.
11. *a)* Одновимірний масив з 10 елементів цілого типу в діапазоні $[50, 200]$;
b) Видалити з початкового масиву елементи з другого по сьомий, пересунувши 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, пересунувши 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, пересунувши елементи масиву, надрукувати їх значення;

c) Знайти добуток елементів перетвореного масиву;

d) Вивести початковий, отриманий масиви і значення добутку.

19. *a)* Одновимірний масив з 9 елементів цілого типу в діапазоні $[-30, -1]$;

b) Переставити шостий елемент початкового масиву на перше місце, пересунувши його елементи, вивести його значення;

- c)* Обчислити добуток елементів перетвореного масиву;
- d)* Вивести початковий, отриманий масиви і значення добутку.
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 максимального елементу.
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 в масиві;
- d)* Вивести початковий, отриманий масиви, значення максимального елементу і його порядковий номер.
22. *a)* Одновимірний масив з 10 елементів цілого типу в діапазоні [-80, -50];
- b)* Пересунути 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 елементу, елементи початкового масиву не пересувати;
- c)* Знайти мінімальний елемент перетвореного масиву;
- d)* Вивести початковий, отриманий масиви і значення мінімального елементу.
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 в 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 з 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 елементи 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 посунути;

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, пересунувши 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

31. 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 по номеру елемент, пересунувши його елементи;
- c) Обчислити суму елементів початкового масиву і знайти мінімальний елемент перетвореного масиву;
- d) Вивести початковий, два отримані масиви, значення суми і мінімального елементу.

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 a)
```

```
' Заповнення початкового масиву випадковими числами
```

```
Randomize 'Запуск лічильника випадкових чисел
```

```
For i = 0 To 9
```

```
' Привласнити ел-ту масиву випадкове число від 1 до 100 сдвнув на -20
```

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

```
Next i
```

```
' Заповнити рядок елементами масиву
```

```

'перемежаючи 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 розмір масиву Second елементів від 0 до k-1
ReDim Second(до - 1)
і = 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 для непарних чисел масиву
First
і = і + 1
End If
Next j
'Заповнити рядок елементами масиву перемежаючи їх пропусками і
вивести на
'екран другий масив з непарними елементами
For і = 0 To до - 1
SecondStr = SecondStr & Second(i)& " "
Next і
MsgBox SecondStr,, "Перетворений масив"

```


'Якщо величина масиву Second більше 3 то викинути елемент з номером 3

If до > 3 Then

'Починаючи з елемента з номером 3 привласнювати йому наступний елемент масиву

'Щоб не вийти за межі масива закінчуємо

'на передостанньому елементі.

For i = 3 To до - 2

Second(i) = Second(i + 1)

Next i

'Оскільки кількість елементів в масиві Second на одиницю зменшилася

'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

For i = 0 To до - 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 до - 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 отримане в програмі все обчислене в програмі вже один раз '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

```

2) *Таблиця результатів тестування.*

| | |
|----------------|-------------------------------|
| 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 |

3) *Створення блок-схеми алгоритму.*

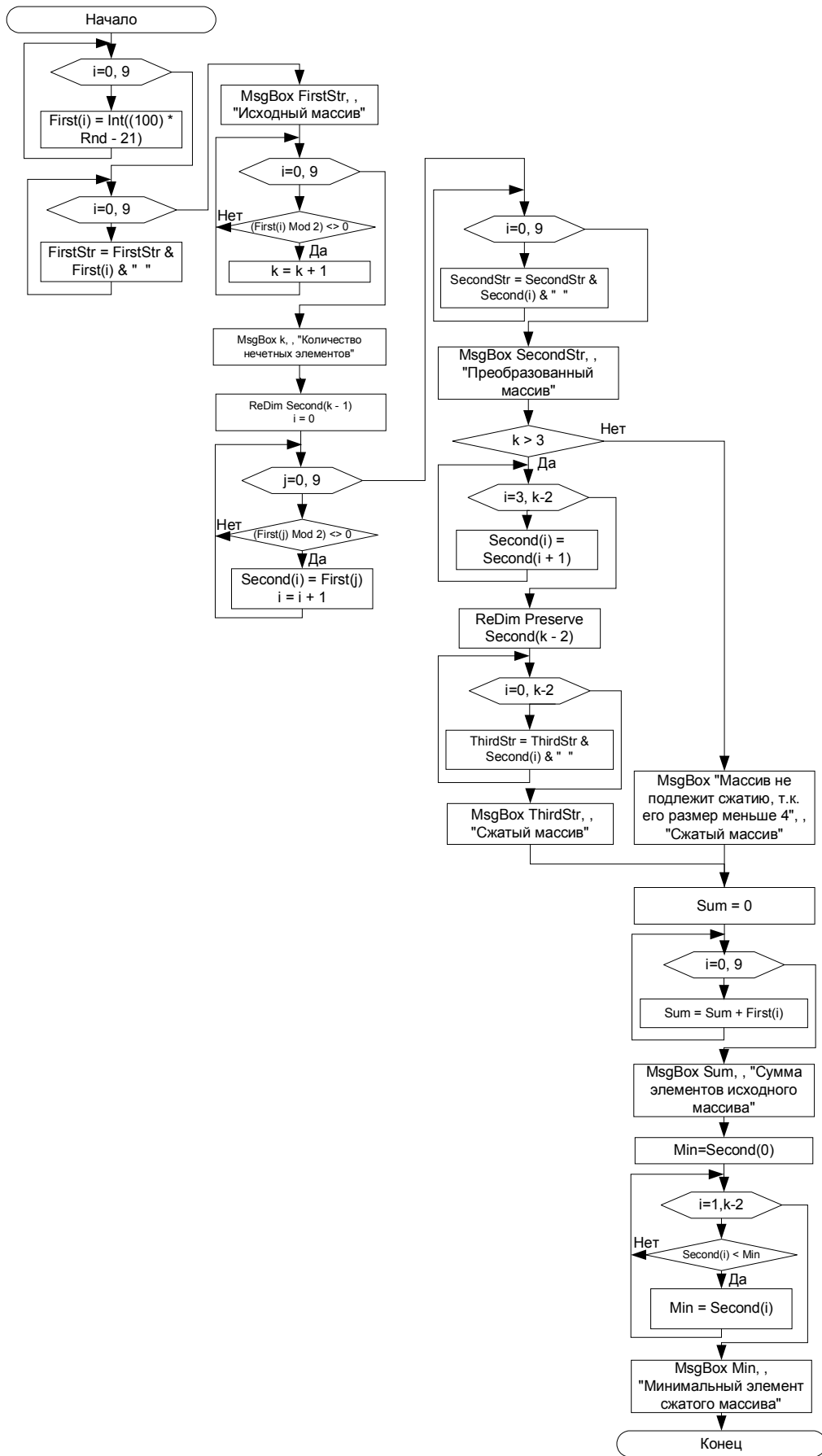


Рисунок 2 – Блок схема алгоритму

5.2. Вкладені цикли. Двовимірні числові масиви.

Багатовимірні масиви

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

На рисунку 3 представлена найбільш поширена форма багатовимірного масиву — двовимірний масив.

Двовимірний масив

| Стовпець 0 | Стовпець 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 |

The diagram shows a 2D array with 10 rows and 2 columns. The columns are labeled 'Стовпець 0' and 'Стовпець 1'. The rows are labeled 'Рядок 0' through 'Рядок 9'. Two callout boxes are present: one pointing to the cell at row 0, column 1 with the text 'Елемент (0,1)', and another pointing to the cell at row 3, column 0 with the text 'Елемент (3,0)'.

Рисунок 3 – 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 стовпців). На рисунку 3 у двовимірному масиву два стовпці (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 стовпця і номер рядка. Індексація двовимірного масиву багато в чому нагадує вказівку осередку листа Excel; перша розмірність масиву відповідає рядкам, а друга — стовпцям.

Якщо у масиву на рисунку 3 ім'я NumTable, то наступна інструкція привласнює значення 10,2 (з другого ряду першого стовпця масиву) змінній AnyNum:

$$\text{AnyNum} = \text{NumTable}(1,0)$$

Аналогічним чином наступна інструкція привласнює значення 2,5 елементу з другого ряду першого стовпця масиву:

$$\text{NumTable}(1,0)=2,5$$

Зверніть увагу, що в обох виразах індекси масиву поміщені в круглі дужки, і координати стовпця 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 більш розмірностей є кропіткою і утомливою. Але на щастя, в програмуванні використовуються в основному одно- і двовимірні масиви.



•Індекси багатовимірного масиву необхідно відокремлювати комами.

•Не відокремлюйте пропусками імена масивів і індекси: у програмі набирайте ім'я, дужки і індекси як одне слово.

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

- a) Сформувати випадковий двовимірний масив 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) Двовимірний масив розмірністю 5 на 8 з цілих елементів в діапазоні $[-30, 30]$;
 - b) Знайти максимальний елемент другого рядка;
 - c) Обчислити суму елементів третього стовпця;
 - d) Вивести початкову матрицю, елементи третього стовпця і значення максимального елемента і 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 і мінімальний елемент.
3.
 - a) Матрицю розмірністю 9 на 6 з випадкових цілих елементів в діапазоні $[-50, 50]$;
 - b) Знайти суму елементів шостого стовпця 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, в діапазоні $[-20, 20]$;
 - b) Зробити початкову матрицю симетричною щодо головної діагоналі;
 - c) Знайти добуток елементів головної діагоналі;
 - d) Вивести на екран початкову, симетричну матриці і добуток елементів.
5.
 - a) Матрицю розмірністю 8 на 7 з елементів дійсного типу, виведених з точністю до десятих, в діапазоні $[-60, 40]$;
 - b) Знайти елементи менші 0 і замінити їх нулями;
 - c) Злічити 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 нову матрицю розміром 4 на 5, починаючи з другого рядка і другого стовпця початкової;

- c)* Знайти суму елементів нової матриці;
- d)* Вивести на екран початкову матрицю, перетворену матрицю 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 і 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 стовпців початкової матриці;
- d)* Вивести на екран початкову матрицю і отриманий вектор.
9. *a)* Матрицю розмірністю 5 на 6 з елементів дійсного типу з точністю до десятих в діапазоні [-40, 40];
- b)* Знайти кількість позитивних елементів початкової матриці;
- c)* Елементи на головній діагоналі замінити одиницями;
- d)* Вивести на екран початкову, перетворену матриці і 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 – добуток елементів п'ятого стовпця, четвертий – добуток елементів сьомого стовпця;
- d)* Вивести на екран початкову матрицю і отриманий вектор.
11. *a)* Матрицю розмірністю 8 на 6 з елементів дійсного типу з точністю до сотих в 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 стовпець і третій рядок;
- c)* Для заміни використовувати додатковий вектор;
- d)* Вивести на екран початкову, перетворену матриці і 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)* Злічити кількість нульових елементів;
- d)* Вивести на екран початкову, перетворену матриці і кількість нулів.
14. *a)* Матрицю розмірністю 6 на 5 з елементів цілого типу в діапазоні [-100, 100];
- b)* Підрахувати кількість негативних елементів;
- c)* Замінити третій і четвертий рядки початкової матриці кількістю негативних 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 стовпця;
- c)* Підрахувати кількість позитивних 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 більших за нуль і кількість елементів менших нуля;
- c)* Максимальний елемент матриці замінити кількістю елементів більших за нуль, а мінімальний – кількістю елементів менших 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 і суму елементів головної діагоналі.

18. *a)* Матрицю розмірністю 8 на 10 з елементів цілого типу в діапазоні [-40, 40];

b) З початкової матриці отримати нову матрицю розмірністю 8 на 5 елементів, починаючи з елемента $X[1, 1]$;

c) Знайти максимальний елемент підматриці;

d) Вивести на екран початкову матрицю, нову матрицю і максимальний елемент.

19. *a)* Матрицю розмірністю 5 на 6 з елементів цілого типу в діапазоні [-40, 50];

b) Поміняти місцями елементи p 'ятого рядка і p 'ятого стовпця;

c) Знайти суму елементів вектора обміну;

d) Вивести на екран початкову, перетворену матриці і 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 стовпців на мінімальний елемент;

d) Вивести на екран початкову, перетворену матриці і 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 добуток парних елементів матриці;

d) Вивести на екран початкову, перетворену матриці і добуток.

22. *a)* Матрицю розмірністю 9 на 6 з елементів цілого типу в діапазоні [-30, 80];

b) Утворити другу матрицю розмірністю 2 на 2 і заповнити її нулями;

c) Замінити елементи початкової матриці, починаючи з p 'ятого рядка і четвертого стовпця, елементами другої матриці;

- d)* Вивести на екран початкові матриці і перетворену матрицю.
23. *a)* Матрицю розмірністю 7 на 8 з елементів цілого типу в діапазоні [-50, 50];
- b)* Утворити другу матрицю розмірністю 7 на 8 і заповнити її випадковими числами в діапазоні від 10 до 20;
- c)* Відняти з першої матриці другу поелементно;
- d)* Вивести на екран початкові 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 і 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 і 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 місцеположення;
- d)* Вивести на екран початкову, перетворену матриці і 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 і 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 та перетворену матриці.

30. a) Матрицю розмірністю 10 на 7 з елементів цілого типу в діапазоні [-60, 60];

- b) Підрахувати кількість позитивних елементів в початковій матриці;
- c) Всім позитивним елементам змінити знак на протилежний;
- d) Вивести на екран початкову, перетворену матриці і кількість позитивних елементів в початковій матриці.

ПРИКЛАД ВИКОНАННЯ ЗАВДАННЯ

31. a) Матрицю розмірністю 5 на 7 з випадкових елементів дійсного типу, заданих з точністю до сотих в 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 стовпця 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) розмір 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 розмірністю 5
на 7
' з елементами занумерованими від 0 до 4 і від 0 до 6
Dim Second(4, 6) As Single
' Виконуємо пункт а) 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 70 із зрушенням
на -31
    First(i, j)= Round((70) * Rnd - 31, 2)
  Next j
Next i
' Створити рядок елементів масиву перемежаючи їх пропусками і з
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 3)
  ' 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

```

‘Створити рядок елементів масиву перемежуючи їх пропусками і з нового рядка

‘по зміні індексу і як лічильника рядка вивести на екран преобр. 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

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

«Масив мін. Елементів рядків 0,2,4: « & HelpStr & Chr(13)& Chr(13)&

«Сума мін. 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

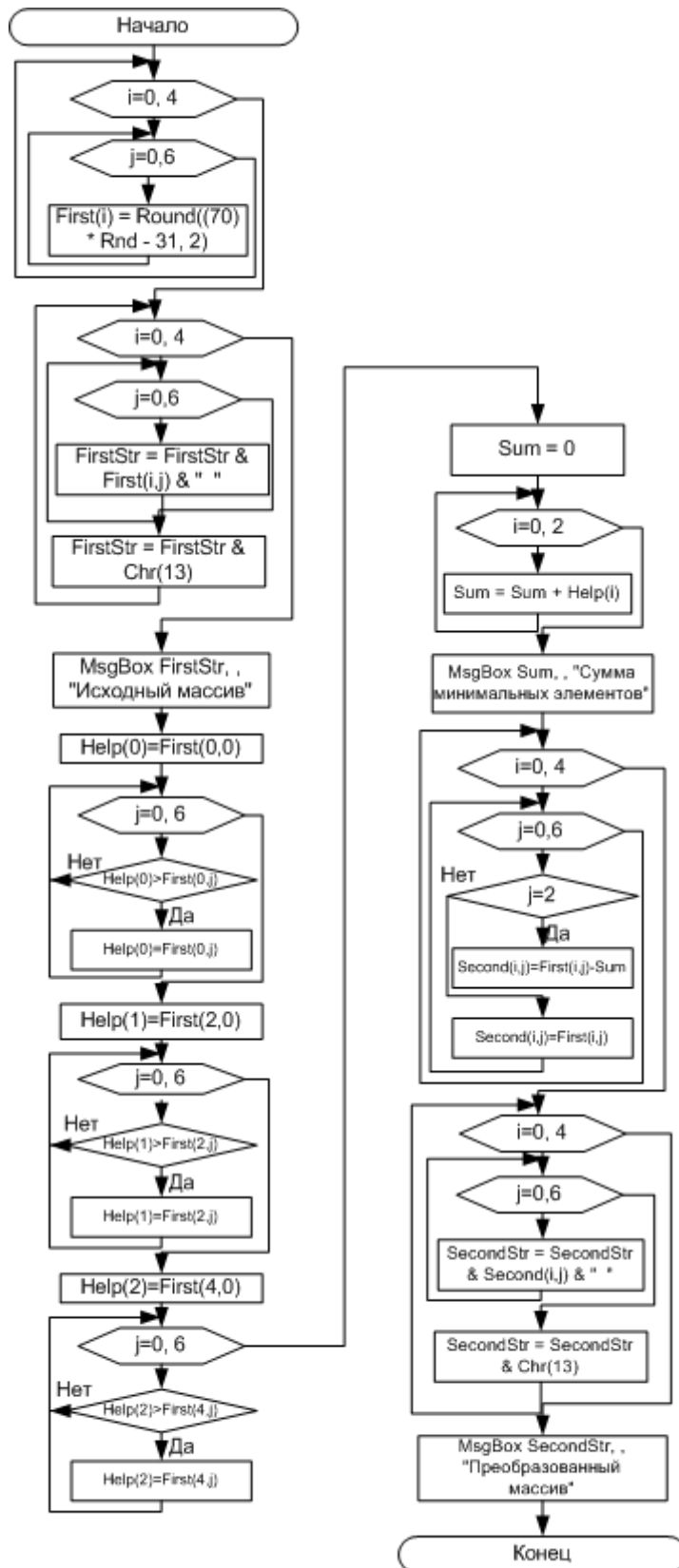


Рисунок 4 – Блок схема алгоритму

СПИСОК ЛІТЕРАТУРИ

1. Васильев А. VBA в Office 2000: учебный курс. / А. Васильев, А. Андреев. – Санкт-Петербург : Питер, 2001. – 432 с.: ил.
2. Гарнаев А.Ю. Использование MS Excel и VBA в экономике и финансах. / А. Ю. Гарнаев. – Санкт-Петербург : БХВ-Санкт-Петербург, 1999. – 336 с.: ил.
3. Долженков В. А. Microsoft Excel 2000. / В. А. Долженков, Ю. В. Колесников. – Санкт-Петербург : БХВ-Петербург, 1999. – 1088 с.: ил.
4. Коттингхэм М. Excel 2000: руководство разработчика: / М. Коттингхэм. Пер. с англ. – Киев : Издательская группа ВНУ, 2000. – 704 с.
5. MS Office XP: разработка приложений / А. В. Матросов, Ф. А. Новиков, Г. Е. Усаров, И. А. Харитоновна. / Под ред. Ф. А. Новикова. – Санкт-Петербург : БХВ-Петербург, 2003. – 944 с.: ил.
6. Саймон, Джинжер. Программирование в Excel: наглядный курс создания интерактивных электронных таблиц. : Пер. с англ. – Москва : Издательский дом «Вильямс», 2002. – 336 с.: ил.
7. Хорев В.Д. Самоучитель программирования на VBA в Microsoft Office / В. Д. Хорев. – Киев : Юниор, 2001. – 296 с., ил.
8. Харрис, Метью. Освой самостоятельно программирование для Microsoft Excel 2000 за 21 день.: Пер. с англ. : Уч. пос. – Москва : Издательский дом “Вильямс”, 2000. – 880 с.: ил.

Навчальне видання

МЕТОДИЧНІ ВКАЗІВКИ
ДО ВИКОНАННЯ РОЗРАХУНКОВО-ГРАФІЧНОЇ РОБОТИ
з дисципліни
**«ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ В БУДІВНИЦТВІ
ТА БАЗИ ДАНИХ»**

*(для студентів 3 курсу денної форми навчання за напрямом
6.060101 – Будівництво спеціальностей «Промислове та цивільне
будівництво»)*

Відповідальний за випуск *Н. О. Манакова*

Укладачі: **МАНАКОВА** Наталія Олегівна,

МАКОГОН Наталія Вікторівна

За авторською редакцією

Комп'ютерне верстання *Н. В. Макогон*

План 2015, поз. 386М

Підп. до друку 02.10.15р.

Формат 60x84/16

Друк на ризографі

Ум. друк. арк. 1,3

Тираж 5 пр.

Зам. №

Видавець і виготовлювач:

Харківський національний університет
міського господарства імені О. М. Бекетова,
вул. Революції, 12, Харків, 61002

Електронна адреса: rektorat@kname.edu.ua

Свідоцтво суб'єкта видавничої справи:

ДК №4705 від 28.03.2014 р.