



УКРАЇНСЬКА ДЕРЖАВНА АКАДЕМІЯ  
ЗАЛІЗНИЧНОГО ТРАНСПОРТУ

# МАТЕМАТИЧНІ МЕТОДИ ТА МОДЕЛІ В РОЗРАХУНКАХ НА ЕОМ

*Навчальний посібник*

*Рекомендовано Міністерством освіти і науки України  
як навчальний посібник для студентів вищих навчальних закладів*

Харків 2008

**ББК 65.9(2)21**

**УДК 681.3.06**

Математичні методи та моделі в розрахунках на ЕОМ: Навч. посібник / І.Г.Бізюк, В.М. Бутенко, О.В. Головка, В.О. Гончаров, В.С. Меркулов; Під заг. ред. М.І. Данька. — Харків: УкрДАЗТ, 2008. — 172 с.

ISBN 5-7763-0080-0

У навчальний посібник включені матеріали лекцій, лабораторно-практичних занять, результати досліджень, виконані співробітниками кафедри "ОТ та СУ" Української державної академії залізничного транспорту при викладанні дисципліни "Математичні моделі на ЕОМ". Розглянуті моделі складних систем та математичні методи їх реалізації, що створюють достатню базу для самостійного творчого пошуку інженерних рішень з конкретного профілю діяльності фахівця транспортної галузі.

Рекомендовано для студентів, слухачів, аспірантів та викладачів вищих навчальних закладів залізничного транспорту.

Іл. 59, табл. 5, бібліогр.: 21 назва.

*Рекомендовано Міністерством освіти і науки України  
як навчальний посібник для студентів вищих навчальних закладів  
(№ 1.4/18-Г2039 від 23 листопада 2007 року)*

**Авторський колектив:**

*Бізюк Ірина Григорівна* — викладач УкрДАЗТ,  
*Бутенко Володимир Михайлович* — канд. техн. наук, доцент,  
*Головка Олександра Володимирівна* — викладач УкрДАЗТ,  
*Гончаров Віктор Олексійович* — доцент УкрДАЗТ,  
*Данько Микола Іванович* — д-р техн. наук, професор,  
*Меркулов Віктор Сергійович* — доцент УкрДАЗТ

**Рецензенти:**

професори А.Л. Єрохін (ХНУВС),  
М.І. Самойленко (ХНАМГ),  
В.М. Самсонкін (ДНДЦ Укрзалізниці)

**©Українська державна академія  
залізничного транспорту, 2008**

# ЗМІСТ

ВСТУП	7
1. ТЕОРЕТИЧНІ ОСНОВИ МАТЕМАТИЧНОГО МОДЕЛЮВАННЯ ТЕХНІЧНИХ СИСТЕМ	12
1.1. Дослідження та проектування технічних об'єктів з використанням ЕОМ	12
1.1.1. Процес проектування як послідовність задач прийняття рішень	12
1.1.2. Визначення і властивості технічної системи	13
1.1.3. Основні поняття і принципи процесу дослідження систем	14
1.1.4. Етапи дослідження систем	15
1.1.5. Аналіз і синтез	17
1.1.6. Системний підхід при аналізі та синтезі систем	18
1.1.7. Зняття невизначеностей у процесі дослідження систем	19
1.2. Основи теорії моделювання	20
1.2.1. Поняття про моделювання	20
1.2.2. Моделі прямої і непрямої аналогії	23
1.3. Чисельні методи розв'язання задач моделювання	26
1.3.1. Обробка результатів експериментів	26
1.3.2. Методи приблизного чисельного розв'язання нелінійних рівнянь	28
1.3.3. Методи приблизного чисельного розв'язання диференціальних рівнянь	30
1.3.4. Методи приблизного чисельного обчислення визначених інтегралів	34
1.3.5. Чисельні методи обчислення визначників і обігу матриць	38
1.3.6. Чисельні методи розв'язання систем лінійних алгебраїчних рівнянь	44
1.4. Основи теорії оптимізації	50
1.4.1. Основні поняття теорії оптимізації	50
1.4.2. Приклади оптимізаційних задач	50
1.4.3. Проектні параметри	51
1.4.4. Цільова функція	52
1.4.5. Знаходження екстремальних значень	53
1.4.6. Простір проектування	54
1.4.7. Обмеження	54
1.4.8. Локальний оптимум	55
1.4.9. Глобальний оптимум	55
1.4.10. Класифікація задач оптимізації	56
1.4.11. Задача вибору оптимальних параметрів контейнера	56
1.5. Методи одновимірного пошуку	58
1.5.1. Суттєвість методів одновимірного пошуку	58

1.5.2. Метод загального пошуку	59
1.5.3. Ділення інтервалу навпіл	60
1.5.4. Метод дихотомії	620
1.5.5. Метод золотого перетину	620
1.6. Методи багатовимірної пошуку	65
1.6.1. Метод покоординатного підйому/спуску. Метод Гаусса-Зейделя	66
1.6.2. Градієнтні методи	69
1.6.3. Східчастий найшвидший підйом	70
1.6.4. Метод випадкового пошуку	70
1.6.5. Симплекс-метод	74
1.6.6. Методи лінійного програмування	75
1.7. Методи багатокритеріальної оптимізації	82
1.7.1. Постановка задачі багатокритеріальної оптимізації	82
1.7.2. Методи згортання показників	83
1.7.3. Мінімаксна згортка	83
1.7.4. Метод штрафних функцій	84
1.7.5. Використання принципу Парето	84
1.8. Динамічне програмування	86
1.8.1. Задачі динамічного програмування	86
1.8.2. Принцип Беллмана	89
1.8.3. Пошук найкоротшого шляху методом динамічного програмування	89
1.9. Метод сіткового планування	90
2. ЛАБОРАТОРНО–ПРАКТИЧНІ РОБОТИ	96
РОБОТА № 1. ДОСЛІДЖЕННЯ НЕЛІНІЙНИХ МОДЕЛЕЙ	99
РОБОТА № 2. ОБРОБКА РЕЗУЛЬТАТІВ ЕКСПЕРИМЕНТІВ	98
РОБОТА № 3. МОДЕЛЮВАННЯ ПРОЦЕСІВ РОЗХОДУ ПАЛИВА	102
РОБОТА № 4. МОДЕЛЮВАННЯ ДИНАМІЧНИХ ДЕТЕРМІНОВАНИХ СИСТЕМ	103
РОБОТА № 5. МЕТОД ВИПАДКОВОГО ПОШУКУ В ЗАДАЧАХ ОДНОВИМІРНОЇ ОПТИМІЗАЦІЇ	105
РОБОТА № 6. МЕТОД ЗОЛОТОГО ПЕРЕТИНУ В ЗАДАЧАХ ОДНОВИМІРНОЇ ОПТИМІЗАЦІЇ	106
РОБОТА № 7. МЕТОД ПОКООРДИНАТНОГО ПІДЙОМУ В ЗАДАЧАХ БАГАТОВИМІРНОЇ ОПТИМІЗАЦІЇ	107
РОБОТА № 8. СИМПЛЕКС–МЕТОД ЛІНІЙНОГО ПРОГРАМУВАННЯ В ЗАДАЧАХ ОПТИМІЗАЦІЇ	109
РОБОТА № 9. МЕТОД ДИНАМІЧНОГО ПРОГРАМУВАННЯ	110
Список літератури	113
ДОДАТОК 1. Зразок оформлення звіту з роботи	113
ДОДАТОК 2. Варіанти додаткових індивідуальних завдань	117

## ВСТУП

В економічному житті України транспорт відіграє особливо важливу роль, забезпечуючи зв'язок між численними регіонами держави. В умовах інтенсивної роботи залізничного транспорту керування перевізним процесом повинне здійснюватися на основі точної інформації про розміщення й стан вагонів, локомотивів і бригад, про хід навантаження й розвантаження для своєчасного здійснення регулювальних заходів. Разом з тим необхідна оптимізація планів перевезень, технічних норм експлуатаційної роботи, плану формування графіка руху поїздів, оперативних планів та інших планових і нормативних документів, щоб максимально поліпшити використання технічних засобів доріг, підвищити продуктивність, поліпшити умови праці. Все це вимагає застосування ЕОМ.

Бурхливий процес математизації та комп'ютеризації науки почався в 50-х роках минулого століття після появи й швидкого вдосконалювання ЕОМ. Він призвів до формування сучасної прикладної математики.

Потреби практики викликали до життя спеціальні наукові методи, що об'єднані за назвою "дослідження операцій". Під цим терміном розуміють застосування математичних, кількісних методів для обґрунтування рішень у всіх областях цілеспрямованої людської діяльності.

Розділи обчислювальної математики, використовувані в процесах прийняття рішень, відіграють усе більш важливу роль у дослідженні операцій. Розвиток лінійного, нелінійного, динамічного програмування, теорії масового обслуговування, теорії ігор і теорії графів характеризують одну зі значних сторін такого роду науки.

Математичні методи й ЕОМ використовуються для розв'язання великих наукових, технічних, економічних задач, для проектування складних об'єктів і керування їхньою роботою, для збору й обробки інформації в природничо-наукових експериментах, для пошуку й реалізації оптимальних режимів виробничо-технологічних процесів і т. д.

Результатом дослідження практично в будь-якій предметній області (виключення — деякі гуманітарні науки: історія, філософія

та ін.) є таблиця експериментального матеріалу. Для того щоб дослідження одержало статус наукового і вирішити поставлені перед інженером задачі, необхідно:

- на підставі цієї таблиці побудувати математичну модель, що дозволяє встановити істотні закономірності досліджуваного об'єкта, і буде достатнім, щоб висновки, отримані в результаті аналізу цієї моделі, підтверджувалися при наступній експериментальній перевірці;
- урахувати, що досить повне розуміння закономірностей досліджуваного об'єкта може бути досягнуто, якщо він описаний не одним, а декількома підходами, що відрізняються взаємодоповнюючими моделями.

У методології як наукового дослідження, так і практичного розв'язання інженерних задач, є два основних напрямки:

- вивчення об'єктів (побудова математичних моделей) будь-якої фізичної природи за експериментальними даними, отриманими у режимі спостереження ("пасивний" експеримент);
- вивчення об'єктів (побудова математичних моделей) будь-якої фізичної природи за експериментальними даними, отриманими при проведенні експериментів за яким-небудь заздалегідь складеним планом ("активний" експеримент).

Переважає більшість реальних технічних, біологічних, медичних і соціальних систем за своїми інформаційними характеристиками (велика кількість вхідних і вихідних параметрів, нелінійна залежність між вхідними й вихідними параметрами, істотний вплив взаємодії різних сполучень вхідних параметрів на вихідні, непогодженість вихідних параметрів — при оптимізації за однією з них погіршуються показники за іншими та ін.) належать до класу "великих" систем.

Людина бере участь на початковому етапі, а саме:

- визначає перелік вхідних і вихідних параметрів об'єкта й вносить їх у комп'ютер у вигляді таблиці даних;
- визначає значення вихідних параметрів (кількісні або якісні), котрі вона хотіла би поліпшити;
- здійснює вибір постановки задачі з набору запропонованих варіантів;

- здійснює вибір звітних матеріалів, що її цікавлять, з набору запропонованих варіантів.

Набору формалізованих алгоритмів, за допомогою яких здійснюється побудова математичних моделей об'єкта як для "активного", так і для "пасивного" експерименту, дана назва — інтелектуальна технологія.

Даний посібник розроблений колективом авторів у процесі викладання дисциплін "Обчислювальна техніка, програмування, моделювання систем", "Математичні моделі на ЕОМ" в Українській державній академії залізничного транспорту та відповідає стандарту Міністерства освіти та науки з цієї дисципліни.

Метою видання є вивчення широкого набору методичних прийомів, що сприяють ефективному вирішенню найважливіших проблем і задач, пов'язаних з організацією перевізного процесу. Це відповідає загальній меті підготовки фахівців, які володіють апаратом математичного (імітаційного) моделювання складних систем і здатні комплексно використовувати в інженерній практиці математичні методи побудови та дослідження моделей за допомогою засобів обчислювальної техніки.

Пропоноване видання заповнює недостачу навчально-методичних матеріалів з основ математичного моделювання. У ньому розглянуті такі питання: дослідження і проектування технічних об'єктів з використанням ЕОМ; основи теорії моделювання; поняття і принципи системного аналізу; чисельні методи розв'язання задач при моделюванні; методи оптимізації.

Докладно розглянуті алгоритми чисельних методів, що найчастіше зустрічаються в інженерних розрахунках.

Посібник адресований, у першу чергу, студентам технічних спеціальностей вузів різного профілю, які вивчають дану або близькі за змістом дисципліни. Також він може бути корисним аспірантам та інженерно-технічним працівникам. Автори сподіваються, що наведені методичні рекомендації допоможуть викладачам організувати роботу студентів у комп'ютерному класі.

Власне кажучи, ця книга є синтезом понять теорії дослідження операцій із практичною їх реалізацією у вигляді обчислювальних алгоритмів.

Книга є складовою частиною публікацій, що повністю охоплюють навчальний матеріал дисциплін "Обчислювальна техніка, програмування, моделювання систем", "Математичні моделі на ЕОМ". Готуються до видання такі навчальні посібники: "Програмування мовами високого рівня", "Прикладні програми загального призначення".

Посібник містить:

- теоретичні основи побудови машинних моделей, чисельні методи їх реалізації та відповідні алгоритми;
- лабораторно-практичні роботи та індивідуальні завдання, де досліджуються особливості використання основних методів;
- розрахунково-графічну роботу, що рекомендується для закріплення й поглиблення отриманих знань і практичних навичок;
- ілюстративний матеріал, що полегшує викладачу структурування та викладення матеріалу;
- методику викладання дисципліни;
- питання, що виносяться на самостійне вивчення під керівництвом і контролем викладача;
- список літератури.

Наведений у книзі математичний апарат нескладний і не виходить за межі звичайного вузівського курсу математики, тому дане видання можна рекомендувати для самоосвіти. У той же час курс вищої математики, що читається у вузах, з упором на вивчення безперервних і детермінованих процесів є недостатнім, щоб розв'язати цілу низку прикладних задач технічної галузі.

Так, наприклад, при моделюванні широко застосовуються методи оптимізації, засновані на використанні лінійного, нелінійного й динамічного програмування, теорії ігор, теорії статистичних рішень, методи теорії розкладів і масового обслуговування. Спільність математичних основ для численних методів дозволяє викласти всі ці методи досить компактно і з єдиних позицій, що значною мірою полегшує їх вивчення та встановлення взаємозв'язку.

"Обчислювальна техніка, програмування, моделювання систем", "Математичні моделі на ЕОМ" є базовими для подальшого



вивчення дисциплін групи проектування, планування, організації, керування вагонним господарством та інших спеціальних дисциплін.

У результаті вивчення дисципліни студент повинен:

**знати:**

- засоби побудування математичних моделей, їх використання та реалізацію на ЕОМ;

- чисельні методи розв'язання алгебраїчних, диференціальних рівнянь та систем рівнянь, обчислення визначених інтегралів тощо;

- методи розроблення програм та застосування пакетів прикладних програм;

**вміти:**

- застосовувати методи інтерполяції та апроксимації при обробці результатів експериментів;

- розробляти алгоритми розв'язання проектних та науково-дослідних задач;

- використовувати методи оптимізації для розв'язання оптимізаційних задач;

- використовувати мови програмування для реалізації математичних методів і моделей на ЕОМ.

# 1. ТЕОРЕТИЧНІ ОСНОВИ МАТЕМАТИЧНОГО МОДЕЛЮВАННЯ ТЕХНІЧНИХ СИСТЕМ

## 1.1. Дослідження та проектування технічних об'єктів з використанням ЕОМ

### 1.1.1. Процес проектування як послідовність задач прийняття рішень

Проектування складних систем можна розглядати як процес прийняття рішень з різноманітних питань.

Це передбачає дослідження системи з метою отримання додаткової інформації та безпосереднього прийняття потрібного проектного рішення на основі свого досвіду та інформації, що отримана у процесі дослідження.

Дослідження системи полягає в застосуванні наукових принципів і засобів до розв'язання задач, що виникають у процесі проектування системи. Велику роль в одержанні інформації для прийняття рішень відіграють дослідження, основані на математичному моделюванні, оскільки без належних кількісних характеристик, що одержані математичним шляхом, неможливо приймати проектні рішення. Однак жодне дослідження не в змозі врахувати безліч різноманітних факторів, що потрібно врахувати при проектуванні системи. Тому для прийняття рішення необхідні і відіграють істотну роль судження проектувальника, основані на його досвіді, інтуїції, знаннях і творчих здібностях (рис. 1.1).

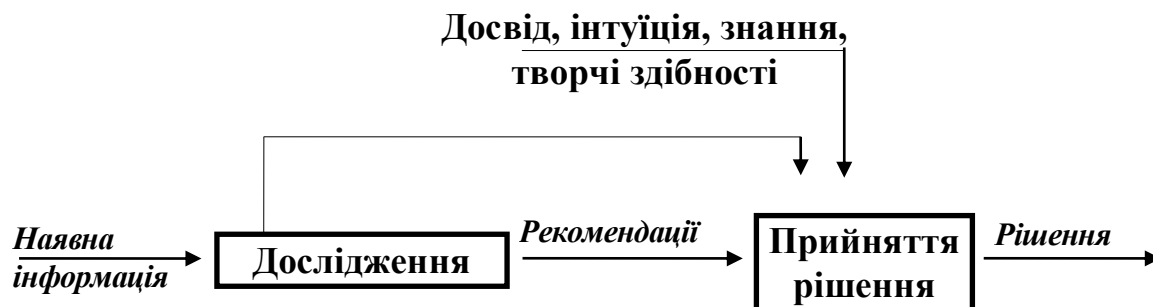


Рис. 1.1. Схема прийняття проектних рішень

У реальному процесі прийняття проектних рішень дослідження і неформальні судження взаємопов'язані і навіть не роздільні.

При більш детальному розгляді схеми прийняття проектних рішень на прикладі розгляду проекту технічної системи вона може бути зображена у вигляді, наведеному на рис. 1.2.

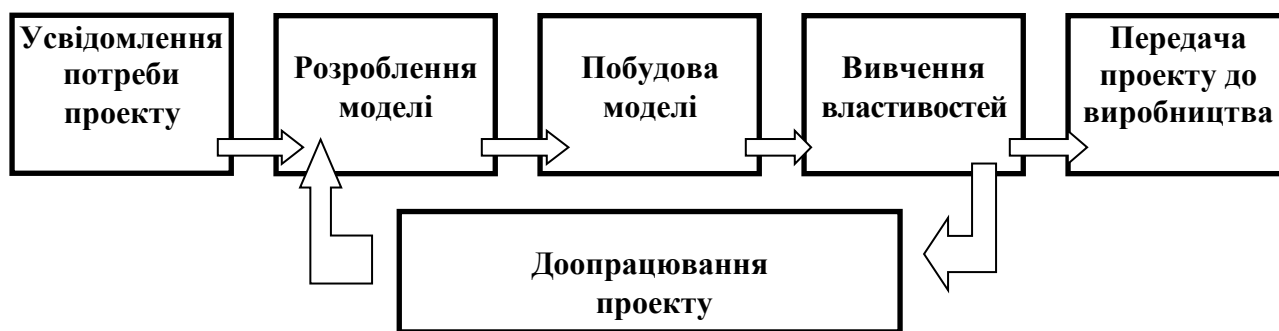


Рис. 1.2. Схема проектування технічної системи

Отже, процес прийняття рішень при проектуванні системи складається з таких основних етапів:

- дослідження системи з метою одержання рекомендацій для вироблення рішень;
- прийняття рішень проектувальником на основі результатів досліджень і неформальних суджень про невраховані при дослідженнях фактори.

### 1.1.2. Визначення і властивості технічної системи

Жодне з філософських визначень поняття "система" не може задовольнити кожную галузь науки.

У конкретних дослідженнях і розробленнях систем правомірно і навіть необхідно оперувати поняттями, що більш точно відображають область реальності, котра досліджується. Тому в літературі можна знайти декілька десятків визначень поняття "система". Наведемо найпоширеніше з визначень.

*Система* — це відокремлений у навколишньому її зовнішньому середовищі і взаємодіючий з ним об'єкт, якому притаманні такі взаємопов'язані властивості:

- має *мету (призначення)*, для досягнення якої він функціонує;
- складається зі взаємопов'язаних *частин-компонентів*, що утворюють багаторівневу ієрархічну структуру і виконують певні функції, направлені на досягнення мети об'єкта;
- має *управління*, завдяки якому всі компоненти функціонують погоджено і цілеспрямовано;
- має у своєму складі або в зовнішньому середовищі *джерела енергії і матеріалів* для функціонування;
- володіє *інтегративними властивостями*, що не зводяться до суми властивостей його компонентів.

*Життєвий цикл технічної системи* — це інтервал часу від початку її створення до кінця її експлуатації. Він складається з таких етапів:

- планування;
- проектування структури;
- проектування елементів;
- виготовлення дослідного зразка;
- серійне виготовлення;
- експлуатація.

### **1.1.3. Основні поняття і принципи процесу дослідження систем**

Дослідження системи в цілому звичайно містить такі істотні елементи:

- мета системи;
- альтернативи;
- показники і критерії;
- моделі;
- алгоритми і програми;
- рекомендації.

Під *альтернативами* розуміють можливі шляхи або засоби досягнення мети системи, що досліджується.

Щоб вибирати альтернативи, необхідно одержувати кількісні оцінки ступеня досягнення мети і найважливіших властивостей

системи. Для цього використовуються відповідні числові характеристики властивостей системи: *показники ефективності, надійності, якості управління та ін.*

Під *показником ефективності* системи будемо розуміти таку її характеристику, що оцінює кількісно ступінь досягнення мети.

Особливу роль відіграють показники, що оцінюють витрати ресурсів. Для створення або експлуатації системи необхідні певні витрати ресурсів (грошей, обладнання, матеріалів, енергії, робочої сили та ін.). Для кожної з альтернатив досягнення мети системи вимагаються свої витрати ресурсів, що найчастіше оцінюються *показником вартості системи та її експлуатації.*

Для вибору більш прийнятної альтернативи використовують *критерії (правила)*, на основі яких за значеннями показників приймаються рекомендації з вибору альтернатив.

Модель у цьому випадку являє собою математичний опис залежності між показниками ефективності та якості системи, альтернативами досягнення мети і характеристиками зовнішнього середовища.

*Алгоритм розв'язання задачі* на ЕОМ — це опис, що визначає послідовність логічних і обчислювальних операцій, котрі слід виконувати над вхідними даними та інформацією, що виникає в ході цих операцій, для того щоб вирішити поставлену задачу. Алгоритм, записаний засобами якої-небудь мови програмування, називається *програмою.*

Результатом дослідження системи є *рекомендації* з вибору більш прийнятної альтернативи. Одержання рекомендацій є *метою дослідження* системи. На основі рекомендацій і неформальних суджень приймається *проектне рішення.*

#### **1.1.4. Етапи дослідження систем**

Процес дослідження системи можна поділити на такі головні етапи:

- постановка задачі;
- вибір критеріїв;
- розроблення моделі;

- дослідження моделі;
- вироблення рекомендацій.

Постановка задачі є важливим етапом дослідження системи і включає в першу чергу:

- формулювання суті задачі дослідження;
- визначення мети системи;
- вибір альтернативних шляхів її досягнення.

Вибір критеріїв прийняття рекомендацій є винятково відповідальним етапом дослідження системи і полягає у виборі таких показників ефективності і якості системи та такого правила прийняття рекомендацій, щоб за числовими значеннями показників можна було судити про успішність розв'язання задачі.

Розроблення математичної моделі — центральний етап дослідження будь-якої системи, бо від якості моделі залежить доля всього дослідження.

Етап розроблення моделі може бути поділений на підетапи:

- побудова моделі;
- вибір відповідного чисельного засобу розв'язання задачі;
- розроблення алгоритму її розв'язання, наприклад, алгоритму оптимізації;
- програмування моделі та обчислювального процесу для ЕОМ.

До цього ж етапу можна віднести підготовку вхідних даних, необхідних як для побудови моделі, так і для проведення наступних обчислень. Дослідження моделі виконується на ЕОМ найчастіше в діалоговому режимі.

Розроблення моделі називають *математичним моделюванням*, або просто моделюванням. Дослідження моделі називають математичним (машинним, обчислювальним) експериментом.

Розроблення рекомендацій — останній етап дослідження, на якому аналізуються результати чисельного дослідження моделі і приймаються рекомендації з вибору варіанта розв'язання поставленої задачі.

Дуже часто означені етапи дослідження важко чітко відокремити один від одного, бо дослідження може йти шляхом паралельного виконання деяких етапів і неодноразових повторень

пройдених етапів, тобто ітеративним шляхом. У таких випадках цикл дослідження повторюється (рис. 1.3).

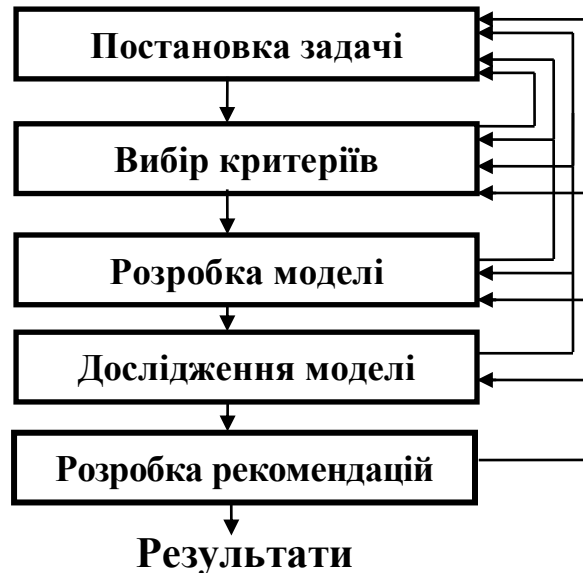


Рис. 1.3. Етапи дослідження системи

### 1.1.5. Аналіз і синтез

Засоби наукового дослідження поділяються на аналіз і синтез.

Під *аналізом* розуміється засіб дослідження шляхом логічного (уявного) розкладу цілого (системи, процесу) на складові та вивчення окремих сторін і властивостей цілого та його складових тощо. *Основним принципом*, що приймається при проведенні аналізу, є *розподіл складної проблеми на окремі завдання*, що легше піддаються розв'язанню, а також розподіленню системи або процесу її функціонування на компоненти.

*Синтез* — засіб дослідження системи або процесу шляхом возз'єднання цілого з частин, узагальнення і зведення в єдине ціле даних, що були добуті аналізом.

Інше значення розглянутих термінів пов'язане з двома протилежними завданнями дослідження системи.

*Завдання аналізу* полягає у визначенні властивостей системи за її структурою і значенням параметрів.

*Завдання синтезу* полягає у визначенні структури і значень параметрів системи за заданими властивостями.

Аналіз системи здійснюється за допомогою розрахунків для серії варіантів структури і параметрів системи — так званих варіантних розрахунків.

Прямий синтез системи — складна проблема. На практиці завдання синтезу звичайно вирішується шляхом вивчення й узагальнення даних, що були добуті аналізом. Оцінивши результати варіантних розрахунків за допомогою критеріїв прийняття рекомендацій, вибирають більш прийнятний варіант структури параметрів системи.

### **1.1.6. Системний підхід при аналізі та синтезі систем**

*Системний підхід* — це методологія науки на рівні загальнонаукових принципів і форм дослідження, що застосовуються в різноманітних галузях науки. Він ґрунтується на діалектико-матеріалістичних принципах системності та пізнання.

В *основі системного підходу* лежить прагнення вивчити об'єкт (систему, проблему, явище, процес) як щось цілісне й організоване, в усій його повноті та в усій множині зв'язків в об'єкті. Цей загальний принцип орієнтує дослідника на розгляд об'єктів як систем.

На рівні конкретної наукової методології принципи системного підходу, що застосовуються при дослідженні і розробленні систем, формулюються більш конкретно. Так, наприклад, відповідно до системного підходу кожний компонент системи повинен розроблятися так, щоб для системи в цілому забезпечити досягнення поставленої перед нею мети і при цьому з ефективністю, що вимагається. Щоб краще зрозуміти суттєвість цього принципу, вкажемо його альтернативу — *несистемний підхід* — це прагнення розробляти елементи системи як самостійні і не пов'язані між собою.

Принцип системного підходу можна розвинути далі, якщо взяти до уваги, окрім системності об'єкта дослідження, ще і системність мислення.

Повнота, завершеність і ефективність дослідження будуть забезпечені, якщо дослідник буде організовувати своє дослідження як систему. Системний підхід припускає, що основні елементи дослідження: мета компонентів системи; показники ефективності та



вартості; моделі та результати дослідження — формуються як ієрархічні системи, відповідні ієрархічній структурі системи.

Системний підхід називають інколи *цільовим* підходом.

Причина багатьох помилкових рішень — орієнтація на проміжні цілі, тобто, коли приймають рішення, часто забувають про кінцеву мету.

### **1.1.7. Зняття невизначеностей у процесі дослідження систем**

Основна складність при підготовці даних для дослідження технічної системи полягає в невизначеності багатьох характеристик технічної системи і зовнішнього середовища. Про неточні і неповні знання прийнято говорити як про невизначеність знань або, коротко кажучи, про *невизначеності*.

Слід розрізняти невизначеності в теперішньому і в майбутньому часі.

Коли досліджується система в майбутньому, що має місце при проектуванні технічної системи, кількість невизначених факторів істотно зростає.

Подолання невизначеностей є основною проблемою при підготовці вхідних даних для дослідження технічної системи.

Можна зазначити такі засоби зняття невизначеностей вхідних даних:

1) найприродніший, очевидний і тривіальний засіб полягає в *пошуку і підборі додаткової інформації*;

2) другий засіб полягає в проведенні *спеціальних досліджень*, теоретичних або експериментальних, для вивчення невизначених факторів і явищ. Однак можуть бути такі технічні системи, для яких неможливе проведення експериментів. Тому потрібні також теоретичні засоби зняття невизначеностей і, особливо, для невизначеностей у майбутньому;

3) третій засіб полягає в *синтезі правдоподібних значень невідомих характеристик зовнішнього середовища* за деяким малим обсягом інформації, наявної в розпорядженні дослідника;

4) для розкриття невизначеностей у майбутньому існують і розробляються *засоби прогнозування*;

5) для підготовки вхідних даних у ситуації невизначеностей застосовується також *метод експертних оцінок*. Експертні оцінки використовуються не тільки для одержання оцінки невизначених вхідних даних, але і для одержання даних, яких не вистачає.

## **1.2. Основи теорії моделювання**

### **1.2.1. Поняття про моделювання**

При дослідженні різноманітних процесів і явищ у науці й техніці широко використовується моделювання. *Моделюванням* називають побудову копії (моделі) якого-небудь процесу або об'єкта. *Завдання моделювання* полягає у відтворенні явища, подібного до оригіналу, і його дослідженні.

Існує ряд визначень поняття "модель". Наведемо одне з них.

*Модель* — це такий матеріальний або уявний об'єкт, що у процесі пізнання (вивчення) заміщує об'єкт — оригінал, зберігаючи деякі найбільш важливі для даного дослідження типові його риси.

*Моделі призначені* для вивчення складних процесів, явищ, конструювання нових споруд тощо. Не можна, наприклад, проводити експерименти з економікою країни, з минулим, з сонячною системою, планетами і т. п.

Моделі використовують для виявлення найбільш істотних факторів, що формують ті або інші властивості об'єктів, а також дозволяють навчитися правильно управляти об'єктами.

Розрізняють декілька типів моделей: *геометричні, фізичні та математичні*. У вжитку найбільш розповсюджені геометричні моделі, що дають зовнішнє уявлення про об'єкт, котрий нас цікавить, і більшою частиною використовуються для демонстраційних цілей. Вони показують принцип чинності, взаємне розташування, компоновання і т. п. Прикладами геометричних моделей є макети машин, цехів, осель, архітектурних виставок тощо. Це, звичайно, експонати виставок, музеїв і просто наочні експонати.

Фізичні та математичні моделі призначені для визначення числових значень величин, що характеризують поведінку об'єкта шляхом модифікації відповідних величин у моделі. При фізичному моделюванні дослідження проводять на моделі, подібній до об'єкта, що відрізняється від оригіналу тільки розмірами. Процеси, що протікають у моделі та оригіналі, мають один і той же фізичний характер. Модель повинна бути подібною до основного об'єкта (споруди) не абсолютно, а тільки в тій мірі, в якій це необхідно для перевірки основних властивостей об'єкта (споруди). Наприклад, якщо модель створена для перевірки міцності мосту, немає необхідності виготовляти на цій моделі перила або освітлювальні ліхтарі. Фізичне моделювання зберігає особливості поведінки об'єкта дослідження, але істотно полегшує одержання результатів, що вимагаються, бо для моделі вибираються найбільш зручні геометричні розміри (збільшені або зменшені) та діапазони модифікації фізичних величин.

*Завданням фізичного моделювання є визначення характеристик основної споруди за характеристиками моделі, знайденими при її дослідженні.* Особливістю фізичного моделювання є те, що для визначення характеристик основної споруди не потрібно математичного опису процесів, що відбуваються в моделі або основному спорудженні. Достатньо мати лише уявлення про механізм (природу) явища, щоб правильно розрахувати параметри основного спорудження за результатами випробувань моделі цієї споруди. Засіб фізичного моделювання має важливе значення, коли в комплекс явищ, що характеризують процес, котрий досліджується, входять такі явища, що не піддаються математичному опису. Фізичне моделювання дозволяє не тільки досліджувати процеси, що мають складний характер, з найменшими матеріальними витратами, але й уточнювати їхній математичний опис в оригіналі, знаходити значення параметрів машин і приладів. Фізичне моделювання застосовують у суднобудуванні, авіабудуванні та гідротехнічному будівництві для розв'язання гідротехнічних і аеродинамічних задач. Його застосовують також для розв'язання статичних і динамічних задач механіки, для яких існують добре розроблені прийоми моделювання. Прикладами фізичного моделювання можуть бути

випробування моделей морських кораблів у "дослідних басейнах", моделей літаків і ракет в аеродинамічних трубах і т. п. Істотним *недоліком* фізичного моделювання є те, що для дослідження кожного нового об'єкта необхідно будувати надто дорогі моделі.

Використання моделювання як інструмента пізнання ґрунтується на *теорії подібності*, основні положення якої зводяться до наступного:

- у світі існує безліч зовні різноманітних процесів і явищ, між якими є певна аналогія, тобто якісна відповідність величин, схожість властивостей і відношень;

- наявність аналогії між двома процесами або явищами в одних властивостях дає основу для прийняття гіпотези про те, що якщо перший процес має властивість "А", то і другий процес має ту ж властивість;

- між аналогічними процесами або явищами можливо встановити подібність. *Подібними* називаються такі процеси і явища, між відповідними параметрами яких є пропорційність.

За наявності подібних процесів або явищ з поведінки одного з них можна судити про поведінку іншого.

Об'єктивною ознакою аналогічності зовні різноманітних процесів служить *ідентичність математичних виразів*, що описують їхню суттєвість.

Дійсно, багато фізичних процесів, що належать до різних областей явищ, описуються аналогічними, найчастіше диференціальними рівняннями.

Математичне моделювання виконується на моделі, фізична природа якої може відрізнитися від фізичної природи оригіналу.

Головна *перевага математичного моделювання* над фізичним полягає в можливості дослідження явищ природи, що важко піддаються вивченню, на добре вивчених явищах. Основні переваги засобу математичного моделювання дозволили йому посісти переважне місце при проведенні наукових досліджень. Таким чином, модель необхідна для того щоб:

- зрозуміти, що собою являє конкретний об'єкт: яка його структура, основні властивості, закони розвитку і взаємодія з навколишнім світом;

- навчитися управляти об'єктом (або процесом) і визначити найкращі засоби управління при заданих меті і критеріях;
- прогнозувати прямі та побічні наслідки реалізації заданих засобів і форм впливу на об'єкт.

### 1.2.2. Моделі прямої і непрямой аналогії

Як вже зазначалося, математичне моделювання може виконуватися на моделі, фізична природа якої може відрізнятися від фізичної природи оригіналу.

При математичному моделюванні розрізняють моделі прямої і непрямой аналогії. Процеси, що відбуваються у фізичній системі, котра досліджується, і моделі описуються ідентичними рівняннями.

*У моделі прямої аналогії всі параметри фізичної системи, що досліджується, мають свої аналоги.*

Якщо в якості моделі використовують електричні ланцюги, то таке моделювання називають електричним. При цьому можна виділити два напрямки реалізації електричних моделей:

- розроблення еквівалентних схем (схем заміщення або аналогій);
- використання аналогових обчислювальних машин (АОМ).

У першому випадку дослідження процесів, що відбуваються в якій-небудь фізичній системі, виконується на схемі, побудованій на основі електричних кіл.

Припустимо, нас цікавлять коливання, що виникають в якійсь конкретній механічній системі (рис. 1.4).

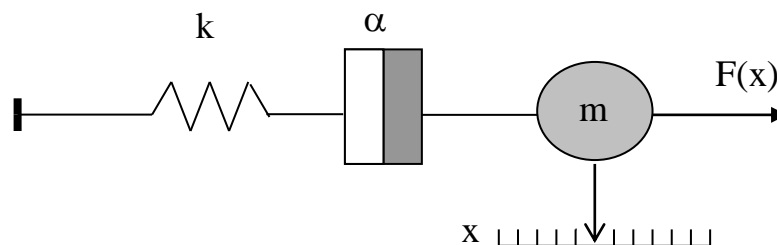


Рис. 1.4. Механічна система

Нехай ці коливання описуються диференційними рівняннями другого порядку

$$m \frac{d^2 x}{dt^2} + \alpha \frac{dx}{dt} + \kappa x = F(x), \quad (1.1)$$

де  $m$  — маса вантажу;  $\alpha$  — коефіцієнт демпфірування;  $\kappa$  — коефіцієнт жорсткості пружини;  $x$  — величина переміщення вантажу.

На іншому прикладі розглянемо, як змінюється струм в електричному ланцюзі, що складається з активного опору  $R$ , індуктивності  $L$  і ємності  $C$ , коли він підключається до джерела живлення  $E(t)$  (рис. 1.5). Для цієї схеми, коли замкнуто ключ  $K$ , можна записати

$$L \frac{d^2 i}{dt^2} + R \frac{di}{dt} + \frac{1}{C} i = E(t). \quad (1.2)$$

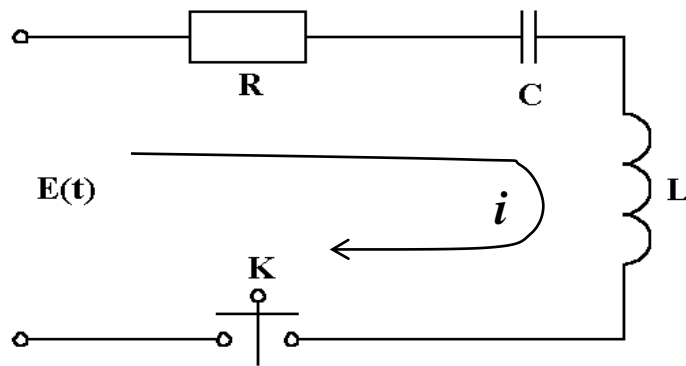


Рис. 1.5. Електричне коло

Порівнюючи співвідношення (1.1) і (1.2), можна помітити, що обидва вони є диференційними рівняннями другого порядку і масі  $m$  першого рівняння відповідає  $L$  в другому рівнянні, коефіцієнту демпфірування  $\alpha$  відповідає  $R$  в другому рівнянні, коефіцієнту жорсткості  $\kappa$  в першому рівнянні відповідає величина  $1/C$  в другому рівнянні.

Таким чином, хоча були розглянуті явища різноманітної фізичної природи (механічної та електричної), рівняння, що описують ці процеси, виявилися ідентичними. Отже, необов'язково виробляти набір пружин різноманітної жорсткості і демпфери з різноманітними параметрами для вивчення поведінки механічної

системи. Це питання можна вирішити значно швидше при значно менших витратах і зі значно більшою точністю, якщо скористуватися електричною схемою, де потрібно лише змінювати величину  $R$ ,  $L$ ,  $C$  і вимірювати струм, що протікає в ланцюзі.

Питання зв'язку між величиною натури (механічної системи) і величиною моделі (електричного ланцюга) вирішується шляхом введення коефіцієнтів пропорційності, що являють собою відношення величини натури до відповідних величин моделі.

Розглянутий приклад математичного моделювання характеризує моделювання на основі прямих аналогій, бо тут використовуються системи аналогій між явищами різноманітної фізичної природи. Перехід з однієї області фізичних явищ в іншу переслідує тут мету спростити і здешевити виготовлення моделей, полегшити методику і підвищити точність вимірювання цих величин.

При побудові моделей непрямої аналогії вибираються найбільш прості блоки для виконання математичних операцій, що потрібні для розв'язання рівнянь, котрі об'єднуються в аналогову обчислювальну машину (АОМ).

Від електричних моделей прямої аналогії АОМ відрізняються тим, що у них відсутня аналогія між параметрами фізичної системи, що вивчається, і параметрами машини.

При дослідженні систем на моделях непрямої аналогії завжди потрібні рівняння, що описують конкретну фізичну систему, приведену до вигляду, зручного для набору задачі. Для цього виконують додаткові, часто трудомісткі, обчислення коефіцієнтів рівнянь, що вводять до АОМ. Вхідні дані, що вводять у машини, тобто коефіцієнти і функції, що одержані в результаті перетворення рівнянь, найчастіше втрачають прямий зв'язок з параметрами фізичних елементів досліджуваної системи.

Моделі прямої аналогії достатньо просто реалізуються за допомогою найпростіших елементів. Наприклад, гідравлічний розрахунок трубопроводу для систем водопостачання та інших гідравлічних мереж може бути виконаний за допомогою електричної моделі, складеної з ламп розжарення (аналог водоводу) і автотрансформаторів, що настроюються на відповідний "гідравлічний опір".

Основним *недоліком* моделей прямої аналогії є їх невелика точність, а також складність створення моделей для математичних залежностей високих порядків. Це обмежує коло задач, що розв'язуються за допомогою моделей — аналогів. Звичайно похибка розв'язання, що досягає 3-5%, задовольняє більшість задач, що зустрічаються в інженерній практиці.

Моделі непрямой аналогії складаються з елементів, котрі моделюють окремі математичні операції, і тому вони є більш універсальними. Тому моделі прямої аналогії будуються для спеціалізованих застосувань, а моделі непрямой аналогії найчастіше у вигляді універсальних приладів — математичних машин АОМ.

### 1.3. Чисельні методи розв'язання задач моделювання

#### 1.3.1. Обробка результатів експериментів

В інженерних розрахунках часто виникає потреба обробки результатів експериментальної залежності, що задана у табличному вигляді координат точок  $(x_i, y_i)$ .

Задача проведення кривої  $f(x)$  через ці точки або побудови полінома  $(n-1)$ -ї степені, значення якого у точці  $x_i$  збігаються зі значеннями  $y_i$ , та подальше визначення  $f(x^*)$ , де  $x^*$  — проміжна точка серед інтервалу заданих координат, називається *задачею інтерполяції*.

Для розв'язання такого типу задач застосовують *метод Лагранжа*, головна розрахункова формула якого має такий вигляд:

$$f(x) = \frac{(x-x_2)(x-x_3)\dots(x-x_n)}{(x_1-x_2)(x_1-x_3)\dots(x_1-x_n)} y_1 + \frac{(x-x_1)(x-x_3)\dots(x-x_n)}{(x_2-x_1)(x_2-x_3)\dots(x_2-x_n)} y_2 + \dots + \frac{(x-x_1)(x-x_2)\dots(x-x_{n-1})}{(x_n-x_1)(x_n-x_2)\dots(x_n-x_{n-1})} y_n = \sum_{i=1}^n y_i \prod_{\substack{j=1 \\ j \neq i}}^n \frac{x-x_j}{x_i-x_j}.$$

При складанні алгоритму (рис. 1.6) позначимо коефіцієнти при  $y_i$  через  $B$  та використаємо формулу

$$f(x^*) = \sum_{i=1}^n B y_i.$$



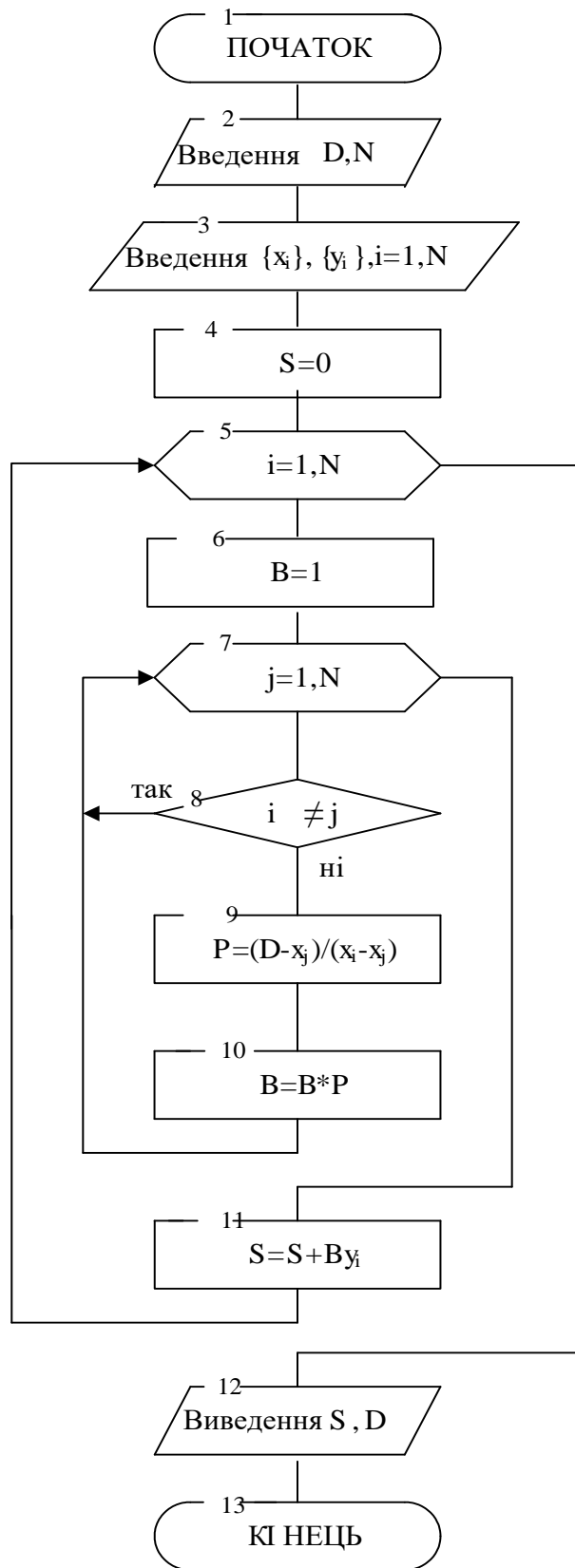


Рис. 1.6. Алгоритм методу Лагранжа

Далі позначимо  $\mathbf{x}^*$  літерою  $\mathbf{D}$ , тоді

$$B = \frac{(D - x_1)(D - x_2)\dots(D - x_{i-1})(D - x_{i+1})\dots(D - x_n)}{(x_i - x_1)(x_i - x_2)\dots(x_i - x_{i-1})(x_i - x_{i+1})\dots(x_i - x_n)}$$

Початкове значення  $\mathbf{B}=\mathbf{1}$ , потім  $\mathbf{B} = \mathbf{B} \frac{D - x_j}{x_i - x_j}$ . Якщо  $i = j$ , то обчислення за формулою не робиться. Для визначення суми спочатку беремо  $\mathbf{f}(\mathbf{x}^*) = \mathbf{0}$ , потім  $\mathbf{f}(\mathbf{x}^*) = \mathbf{f}(\mathbf{x}^*) + \mathbf{B}y_i$ .

В алгоритмі, наведеному на рис. 1.6, літерою  $\mathbf{S}$  позначено  $\mathbf{f}(\mathbf{x}^*)$ .

### 1.3.2. Методи приблизного чисельного розв'язання нелінійних рівнянь

Нехай для дослідження моделі необхідно визначити корінь рівняння  $\mathbf{f}(\mathbf{x})=\mathbf{0}$ , тобто знайти таке значення  $x = x^*$ , підстановка якого у рівняння робить його тотожністю.

Процедура визначення кореня складається з двох етапів:

- *відділення кореня (ізоляція)* — визначення якнайменших інтервалів, на яких міститься один корінь;
- *уточнення кореня* — доведення значення кореня до заданої точності.

**Відділення коренів.** Якщо безперервна та монотонна функція на кінцях інтервалу  $[\mathbf{a}, \mathbf{b}]$  має значення різних знаків, то на цьому інтервалі міститься хоча б один корінь. Розіб'ємо інтервал існування функції  $[\mathbf{x}_n, \mathbf{x}_k]$  на рівні відрізки довжиною  $\Delta \mathbf{x}$  та будемо обчислювати значення функції на кінцях цих відрізків ( $\mathbf{f}(\mathbf{a})$  та  $\mathbf{f}(\mathbf{b})$  відповідно). Умовою наявності кореня буде  $\mathbf{f}(\mathbf{a}) * \mathbf{f}(\mathbf{b}) \leq 0$ .

Алгоритм відділення коренів наведений на рис. 1.7.

**Уточнення кореня.** Нехай відомо, що функція на кінцях інтервалу  $[\mathbf{a}, \mathbf{b}]$  має значення різних знаків. Для уточнення кореня застосуємо, наприклад, метод половинного ділення, що складається з наступних дій:

- 1) розділимо  $[\mathbf{a}, \mathbf{b}]$  на дві рівні частини та визначимо значення функції у точці  $t = \frac{\mathbf{a} + \mathbf{b}}{2}$ ;

- 2) якщо  $f(t) = 0$ , то  $t$  — корінь рівняння; виведемо його значення — інакше перейти до дії 3;
- 3) аналізуємо знак добутку  $f(a) * f(t)$ ;
- 4) якщо цей добуток має від'ємне значення, то вважаємо  $b = t$ , корінь буде на ділянці  $[a, t]$ ;
- 5) якщо цей добуток має додатне значення, то вважаємо  $a = t$ , корінь буде на ділянці  $[t, b]$ ;
- б) аналізуємо виконання умови  $|b-a| > E$ , де  $E$  — точність уточнення кореня, і якщо вона виконується, то повертаємося до дії 1, у протилежному випадку закінчуємо алгоритм.

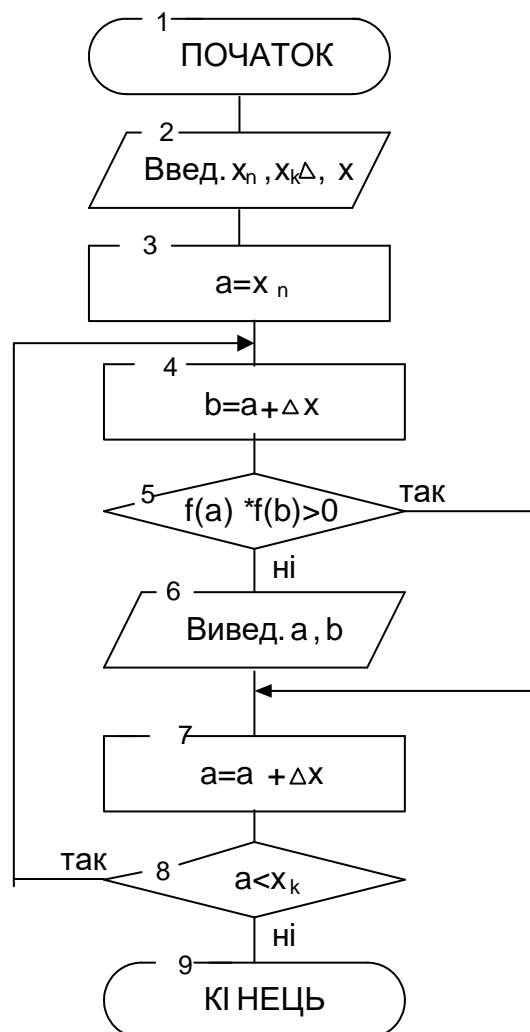


Рис. 1.7. Алгоритм виділення коренів

Алгоритм методу половинного ділення наведений на рис. 1.8.

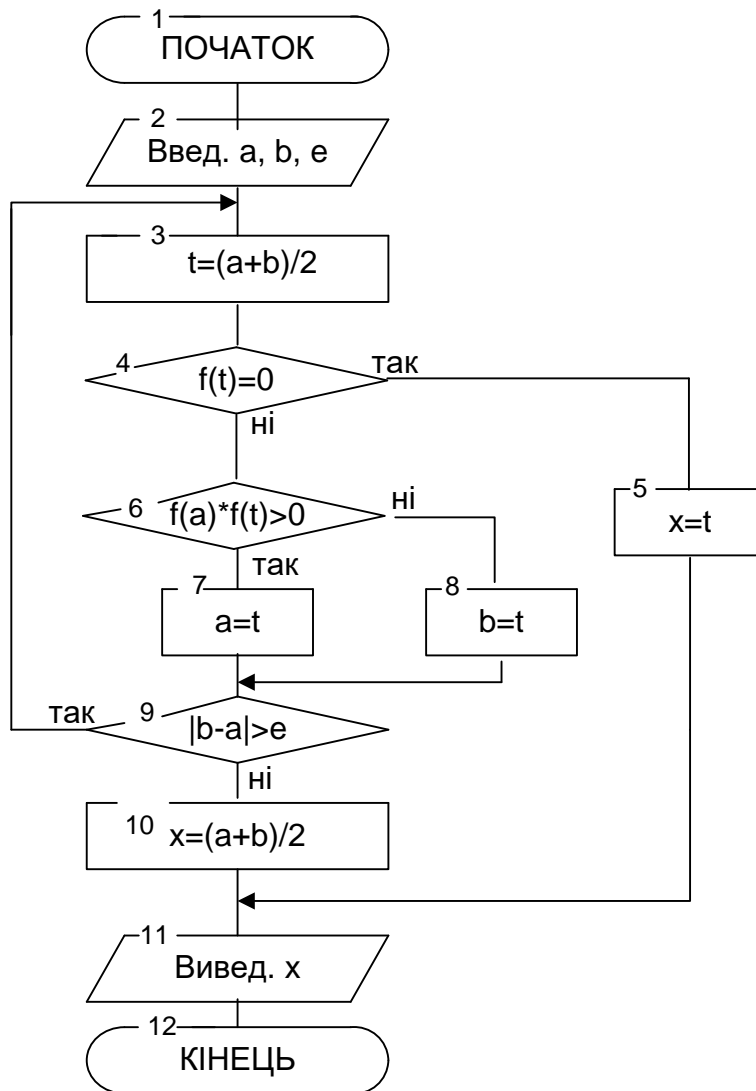


Рис. 1.8. Алгоритм методу половинного ділення

### 1.3.3. Методи приблизного чисельного розв'язання диференційних рівнянь

Нехай поведінка моделі описується диференційним рівнянням  

$$y' = f(x, y).$$

Розв'язати таке рівняння означає знайти таку функцію  $y = y(x)$ , що задовольняє початкову умову  $y(x_0) = y_0$  та визначається виразом

$$Y = \int f(x, y(x))dx + c,$$

де  $c$  — довільна константа.

Характерним для усіх чисельних методів є те, що при розв'язанні диференційних рівнянь потрібно знайти не первісну, а лише значення самої функції  $y_0, y_1, \dots, y_k$  на певному інтервалі  $[x_0, x_k]$ . Кожне наступне значення функції  $y$  обчислюється через попереднє за допомогою рекурентних співвідношень.

**Метод Ейлера.** У його основі лежить розкладення функції в ряд Тейлора біля точки  $x_0$

$$y(x_0+h) = y(x_0) + h y'(x_0) + \frac{1}{2!} h^2 y''(x_0) + \dots$$

Оскільки значення  $h$  мале, залишимо у розрахунковій формулі перші два елементи ряду

$$y(x_0+h) = y(x_0) + h y'(x_0).$$

Цю формулу, враховуючи вихідне рівняння, можна записати:

$$y(x_{n+1}) = y(x_n) + h f(x_n, y_n), \quad n=0,1,2, \dots$$

Алгоритм реалізації методу Ейлера наведений на рис. 1.9.

**Метод Рунге-Кутта.** Точнішим та більш розповсюдженим методом приблизного розв'язання диференційних рівнянь є метод Рунге-Кутта.

У його модифікації 4-го порядку на кожному кроці права частина рівняння обчислюється 4 рази та перехід від точки  $(x_n, y_n)$  до точки  $(x_{n+1}, y_{n+1})$  здійснюється за допомогою таких співвідношень:

$$T_1 = f(x_n, y_n);$$

$$T_2 = f(x_n + \frac{h}{2}, y_n + \frac{T_1}{2});$$

$$T_3 = f\left(x_n + \frac{h}{2}, y_n + \frac{T_2}{2}\right);$$

$$T_4 = f(x_n + h, y_n + T_3);$$

$$y_{n+1} = y_n + \frac{h}{6}(T_1 + 2T_2 + 2T_3 + T_4);$$

$$x_{n+1} = x_n + h.$$

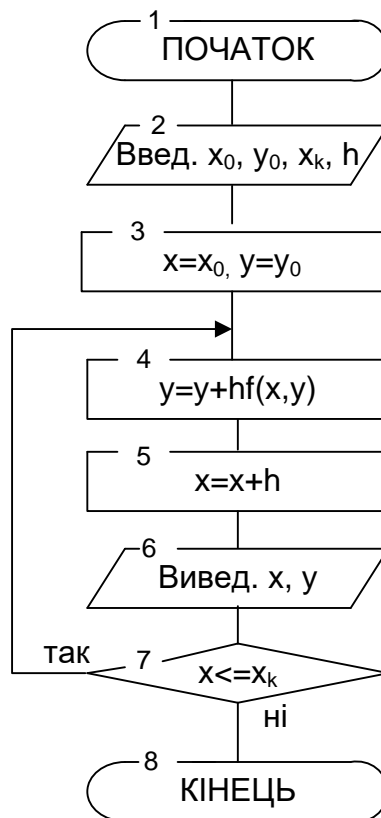


Рис. 1.9. Алгоритм реалізації методу Ейлера

Алгоритм реалізації методу наведений на рис. 1.10.

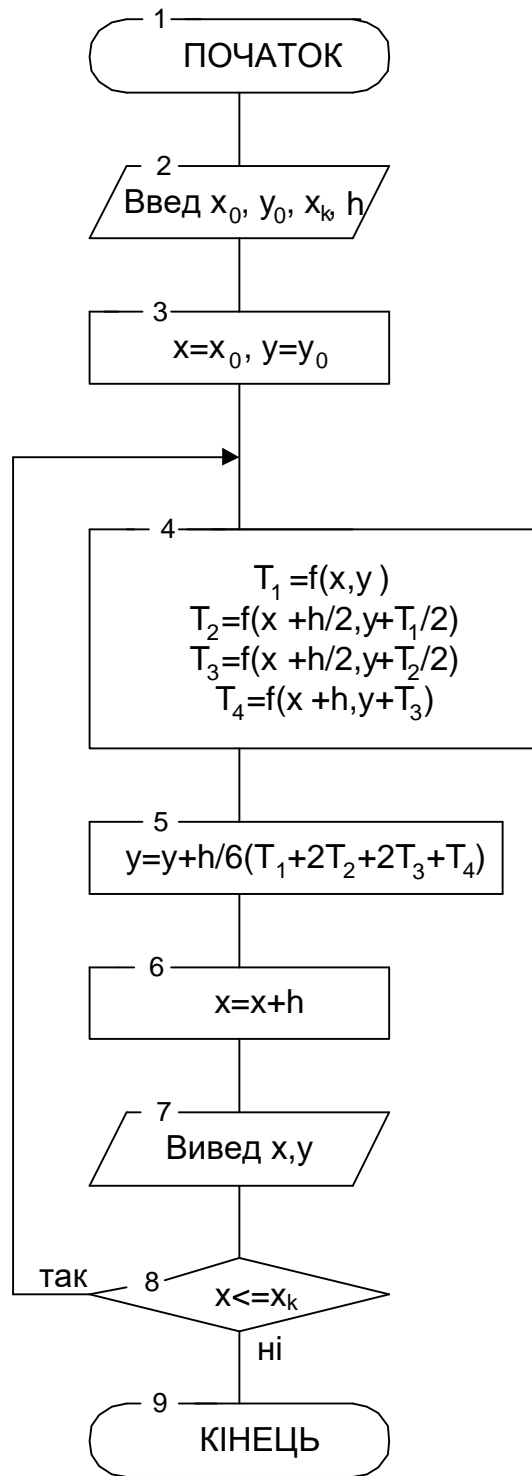


Рис. 1.10. Алгоритм методу Рунге-Кутта

### 1.3.4. Методи приблизного чисельного обчислення визначених інтегралів

В інженерних розрахунках та при моделюванні часто виникає необхідність обчислення значень визначеного інтегралу, що має

$$\text{вигляд } D = \int_a^b f(x)dx.$$

*Визначений інтеграл* — це площа, котра обмежена з одного боку підінтегральною кривою та віссю ОХ, а з іншого — напівпрямими, що проходять через точки  $a$  і  $b$ , паралельно осі ОУ (рис. 1.11).

Формули для приблизного обчислення отримують заміною підінтегральної функції  $f(x)$  інтерполяційним поліномом. У цьому випадку визначений інтеграл може бути записаний у вигляді

$$D = \int_a^b f(x)dx = \sum_{k=1}^m A_k f(x_k) + R,$$

де  $x_k$  — вузли інтерполяції;  $A_k$  — коефіцієнти, що залежать від суті формули та вибору вузлів;  $R$  — похибка обчислювальної процедури.

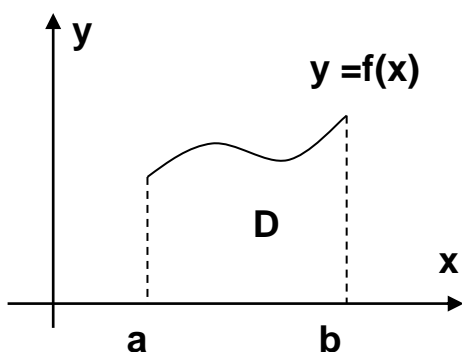


Рис. 1.11. Геометрична інтерпретація визначеного інтегралу

У залежності від того, як інтерполюється  $f(x)$ , отримують різні формули для обчислення визначених інтегралів. Якщо використовують поліноми нульового порядку, маємо формулу прямокутників, першого порядку — формулу трапецій, другого порядку — формулу Симпсона.

Після заміни функції поліномом значення визначеного інтегралу обчислюється як сума площин елементарних фігур, що обмежені віссю  $x$  та кривою  $y = f(x)$ .



При використанні *методу прямокутників* приблизне значення інтегралу визначається за формулою

$$D = \int_a^b f(x)dx \approx \frac{b-a}{n}(y_1 + y_2 + \dots + y_n) = \frac{b-a}{n} \sum_{i=1}^n y_i,$$

де  $y_i$  — значення  $f(x)$  на початку кожного  $i$ -го інтервалу;  $n$  — кількість відрізків, на які розділений діапазон інтегрування;  $a$  — нижня межа інтегрування;  $b$  — верхня межа інтегрування.

У цьому методі крива підінтегральної функції замінюється ламаною лінією, відрізки якої паралельні осі абсцис з наступним визначенням суми площин елементарних прямокутників.

Алгоритм методу прямокутників наведений на рис. 1.12.

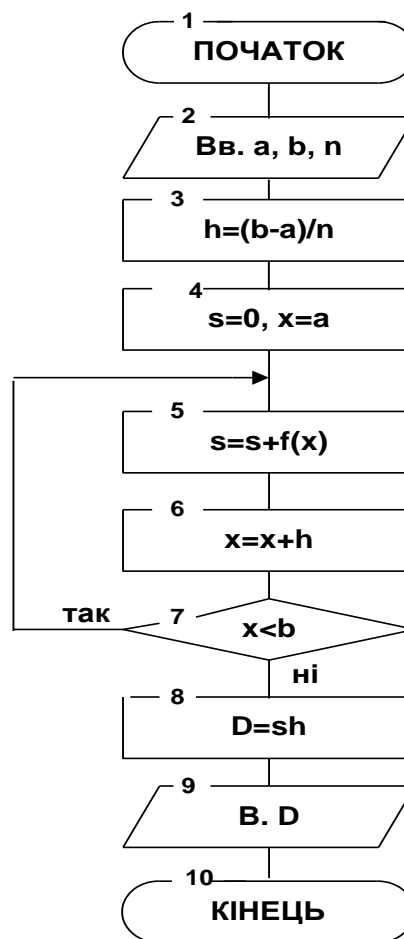


Рис. 1.12. Алгоритм методу прямокутників

У методі трапецій інтервал інтегрування  $[a, b]$  розділяється на  $n$  рівних відрізків довжиною  $h = \frac{(b-a)}{n}$  з наступним визначенням суми площин елементарних трапецій, що апроксимують підінтегральну функцію. Відрізки ламаної з'єднують точки з координатами  $(x_i, y_i)$  та  $(x_{i+1}, y_{i+1})$ . Приблизне значення інтегралу визначається за формулою

$$D = \int_a^b f(x)dx \approx \frac{b-a}{n} \left( \frac{y(a)}{2} + y_2 + y_3 + \dots + \frac{y(b)}{2} \right),$$

що побудована на основі формули для визначення площі елементарної трапеції:

$$S_i = \frac{y_i + y_{i+1}}{2} h.$$

В алгоритмі в циклі обчислюється сума значень функції в точках  $x_2, x_3, \dots, x_{n-1}$ .

Алгоритм методу трапецій наведений на рис. 1.13.

У методі Симпсона інтервал інтегрування  $[a, b]$  розділяється на парне число рівних відрізків довжиною  $h = \frac{(b-a)}{n}$ . Далі підінтегральна крива на кожному відрізку апроксимується параболою, що проходить через три точки. Означений інтеграл обчислюється як сума елементарних криволінійних трапецій за формулою

$$D = \int_a^b f(x)dx \approx \frac{h}{3} (y(a) + 4y_2 + 2y_3 + \dots + 4y_{n-2} + 2y_{n-1} + y(b)).$$

Алгоритм зручно розділити на такі етапи:

- знаходження значень функції з парними індексами  
 $S_1 = y_2 + y_4 + \dots + y_{n-2};$
- знаходження значень функції з непарними індексами  
 $S_2 = y_3 + y_5 + \dots + y_{n-1};$

- обчислення інтегралу  $D = \int_a^b f(x)dx \approx \frac{h}{3}(y(a) + 4S1 + 2S2 + y(b))$ .

Через  $x$  в алгоритмі позначений аргумент, що відповідає парним номерам індексів, а через  $z$  — непарним номерам.

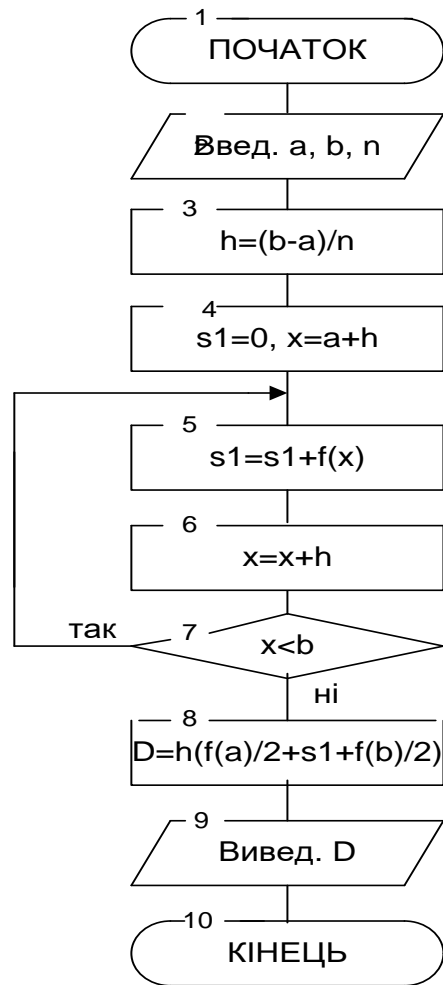


Рис. 1.13. Алгоритм методу трапецій

Алгоритм методу Симпсона наведений на рис. 1.14.

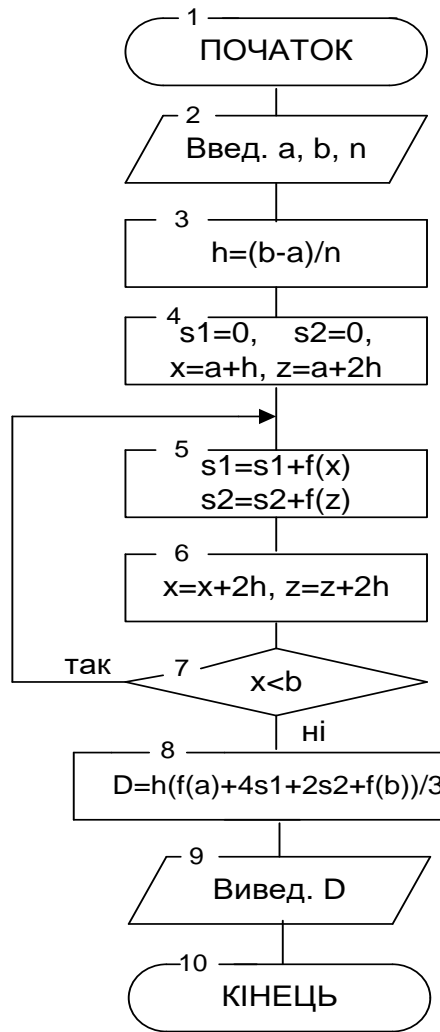


Рис. 1.14. Алгоритм методу Симпсона

### 1.3.5. Чисельні методи обчислення визначників і обігу матриць

Визначник квадратної матриці  $A$  — число, що дорівнює сумі  $n!$  членів  $(-1)^r a_{1j_1} a_{2j_2} \dots a_{nj_n}$ , кожний з яких відповідає одному з  $n!$  різних упорядкованих множин  $j_1, j_2, \dots, j_n$ , отриманих  $r$ -парними перестановками елементів із множини  $1, 2, \dots, n$ .

Визначники матриць малих розмірностей доцільно обчислювати за формулами:

$$D = a_{11}a_{22} - a_{21}a_{12} \quad (2 \times 2);$$

$$D = a_{11}a_{22}a_{33} + a_{21}a_{32}a_{13} + a_{12}a_{23}a_{31} - a_{31}a_{22}a_{13} - a_{32}a_{23}a_{11} - a_{21}a_{12}a_{33} \quad (3 \times 3).$$

Визначник матриць із будь-якою розмірністю  $n \times n$  обчислюємо за методом Гаусса (рис. 1.15). Він зводиться до перетворення матриці до трикутного вигляду за допомогою такої формули:

$$a_{ij}^{(k)} = a_{ij}^{(k-1)} - a_{ij}^{(k-1)} \frac{a_{kj}^{(k-1)}}{a_{kk}^{(k-1)}}, \text{ де } k=1, 2, \dots, n-1 \text{ та } a_{kk}^{(k-1)} \neq 0.$$

Якщо вихідна матриця наведена до трикутного вигляду, то процес обчислення значно спрощується.

Розглянемо послідовність необхідних перетворень.

Необхідна умова для реалізації простого методу Гаусса полягає у виконанні умови  $a_{kk}^{(k-1)} \neq 0$  на всіх етапах перетворення. У випадку  $a_{kk}^{(k-1)}=0$  виникне некоректна операція ЕОМ. Великі похибки виникають, якщо  $a_{kk}^{(k-1)} \neq 0$  є малим числом. Для підвищення точності обчислення визначника і виключення появи ситуації ділення на нуль у ході обчислень застосовуються варіанти методу з вибором головного елемента по стовпцю, рядку або по всій матриці.

В останньому випадку перед перетвореннями в матриці вибирають найбільший за модулем (головний) елемент, припустимо  $a_{ij}$ . Далі переставляють 1-й рядок матриці з  $i$ -м рядком (це  $i$  є вибір головного елемента по рядку), а 1-й стовець переставляють із  $j$ -м (вибір головного елемента по стовпцю). При кожній перестановці знак визначника змінюється, тому результат потрібно помножити на  $(-1)^P$ , де  $P$  — число перестановок (рис.1.15).

Матриця  $A^{-1}$  називається зворотною до квадратної матриці  $A$ , якщо їхній добуток дорівнює одиничній матриці  $A A^{-1} = A^{-1} A = I$

У лінійній алгебрі доводиться, що будь-яка невинроджена матриця  $A$  (тобто з відмінним від нуля визначником  $D$ ) має зворотну. При цьому  $\det A^{-1} = 1/D$ .



Обіг матриці  $A$  полягає в знаходженні зворотної матриці  $A^{-1}$ . Для обігу матриці  $A$  використовують таке основне співвідношення:

$$AA^{-1} = \begin{pmatrix} a_{11} & a_{12} & a_{1n} \\ a_{21} & a_{22} & a_{2n} \\ \dots & & \\ a_{n1} & a_{n2} & a_{nn} \end{pmatrix} \times \begin{pmatrix} x_{11} & x_{12} & x_{1n} \\ x_{21} & x_{22} & x_{2n} \\ \dots & & \\ x_{n1} & x_{n2} & x_{nn} \end{pmatrix} = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \dots & & & \\ 0 & 0 & \dots & 1 \end{pmatrix}.$$

*Мінором елемента  $\alpha_{ij}$*  називається визначник  $n-1$  порядку, утворений з визначника матриці  $A$  закреслюванням  $i$ -го рядка й  $j$ -го стовпця.

Алгебраїчним доповненням  $A_{ij}$  елемента  $\alpha_{ij}$  називається його мінор, узятий зі знаком "+", якщо сума  $i+j$  парна, й зі знаком "-", якщо сума непарна, тобто

$$A_{ij} = (-1)^{i+j} \begin{vmatrix} a_{11} & \dots & a_{1,j-1} & a_{1,j+1} & \dots & a_{1n} \\ \dots & & & & & \\ a_{i-1,1} & \dots & a_{i-1,j-1} & a_{i-1,j+1} & \dots & a_{i-1,n} \\ a_{i+1,1} & \dots & a_{i+1,j-1} & a_{i+1,j+1} & \dots & a_{i+1,n} \\ \dots & & & & & \\ a_{n1} & \dots & a_{n,j-1} & a_{n,j+1} & \dots & a_{nn} \end{vmatrix}.$$

Кожний елемент  $x_{ij}$  ( $i, j=1, \dots, n$ ) зворотної матриці дорівнює відношенню алгебраїчного доповнення  $A_{ji}$  елемента  $\alpha_{ji}$  (не  $\alpha_{ij}$ ) вихідної матриці  $A$  до значення її визначника  $D$ :

$$A^{-1} = \begin{pmatrix} \frac{A_{11}}{D} & \frac{A_{21}}{D} & \dots & \frac{A_{n1}}{D} \\ \frac{A_{12}}{D} & \frac{A_{22}}{D} & \dots & \frac{A_{n2}}{D} \\ \dots & \dots & \dots & \dots \\ \frac{A_{1n}}{D} & \frac{A_{2n}}{D} & \dots & \frac{A_{nn}}{D} \end{pmatrix}.$$

Можна також підрахувати число операцій, необхідних для обчислення зворотної матриці без використання спеціальних

прийомів. Це число дорівнює сумі числа операцій, за допомогою яких обчислюється  $n^2$  алгебраїчних доповнень, кожне з яких є визначником  $n-1$  порядку, і  $n^2$  ділень алгебраїчних доповнень на визначник  $D$ . Таким чином, загальне число операцій:

$$N = [(n-1)(n-1)!-1]n^2 + n^2 + nn!-1 = n^2n!-1.$$

У той же час відзначимо, що, перемноживши  $A$  і  $A^{-1}$ , одержимо  $n$  систем рівнянь із  $n^2$  невідомими.

$$\left\{ \begin{array}{l} a_{11}x_{11} + a_{12}x_{21} + \dots + a_{1n}x_{n1} = 1; \\ a_{21}x_{11} + a_{22}x_{21} + \dots + a_{2n}x_{n2} = 0; \\ \dots \\ a_{n1}x_{11} + a_{n2}x_{21} + \dots + a_{nn}x_{n1} = 0; \\ \\ a_{11}x_{12} + a_{12}x_{22} + \dots + a_{1n}x_{n2} = 0; \\ a_{21}x_{12} + a_{22}x_{22} + \dots + a_{2n}x_{n2} = 1; \\ \dots \\ a_{n1}x_{12} + a_{n2}x_{22} + \dots + a_{nn}x_{n2} = 0; \\ \\ a_{11}x_{1n} + a_{12}x_{2n} + \dots + a_{1n}x_{nn} = 0; \\ a_{21}x_{1n} + a_{22}x_{2n} + \dots + a_{2n}x_{nn} = 0; \\ \dots \\ a_{n1}x_{1n} + a_{n2}x_{2n} + \dots + a_{nn}x_{nn} = 1. \end{array} \right.$$

Розв'язання цих рівнянь визначає значення всіх елементів матриці  $A^{-1}$  -  $x_{ij}$  (рис. 1.16).

Оскільки матриця системи не змінюється, то виключення невідомих при використанні методу Гаусса проводиться тільки один раз. Для кожної системи робиться тільки зворотний хід після деяких перетворень. Оцінки показують, що це досить економічний спосіб обігу матриці, він вимагає приблизно лише в 3 рази більше дій, ніж при розв'язанні однієї системи рівнянь (рис. 1.16).





### 1.3.6. Чисельні методи розв'язання систем лінійних алгебраїчних рівнянь

Розглянемо методи розв'язання систем лінійних алгебраїчних рівнянь (СЛАР). Метод називається точним (прямим), якщо за кінцеве число арифметичних операцій з точними числами можна одержати точне рішення СЛАР. У протилежному випадку метод наближений (ітераційний). Відзначимо, що точні методи — це не настільки ідеалізовані алгоритми, що на ЕОМ їх не можна реалізувати для СЛАР із цілочисельними коефіцієнтами. Використовуючи на ЕОМ арифметику цілих чисел, за допомогою точних методів можна знаходити точні рішення.

Задача розв'язання СЛАР має важливе прикладне значення: вона використовується в обчислювальній математиці, математичній фізиці, при обробці результатів експериментальних досліджень.

Необхідно розв'язати СЛАР

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = a_{1n+1}; \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = a_{2n+1}; \\ \dots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = a_{nn+1}, \end{cases} \quad (1.3)$$

де  $x_k$  — невідомі величини;  $a_{ij}$  — задані елементи розширеної матриці СЛАР.

#### *Метод Гаусса*

Метод Гаусса належить до прямих методів. Алгоритм його складається з двох етапів. Перший етап називається прямим ходом методу й полягає в послідовному виключенні невідомих з рівнянь, починаючи з  $x_1$ .

З першого рівняння системи (1.3) виразимо  $x_1$ :

$$x_1 = (a_{1n+1} - a_{12}x_2 - \dots - a_{1n}x_n) / a_{11}. \quad (1.4)$$

Це можливо при  $a_{11} \neq 0$ ; у протилежному випадку треба здійснити перестановку рівнянь системи. Відповідно до формули

(1.4) необхідно кожний елемент першого рядка розширеної матриці СЛАР розділити на діагональний елемент

$$a_{1j}^{(1)} = \frac{a_{1j}}{a_{11}}. \quad (1.5)$$

Потім підставляємо вираз (4) в усі інші рівняння системи, тим самим виключаємо  $x_1$  із всіх рівнянь, крім першого.

Елементи розширеної матриці перетворимо за формулою

$$a_{ij}^{(1)} = a_{ij} - a_{ij}a_{1j}^{(1)}, \quad i = 2, 3, \dots, n+1. \quad (1.6)$$

У результаті виключення першого невідомого  $x_1$  із всіх рівнянь всі елементи першого стовпця перетвореної матриці будуть дорівнювати нулю, крім  $a_{11}^{(1)} = 1$ .

Невідоме  $x_2$  виразимо з другого рівняння системи й виключимо з інших рівнянь і т. д. У результаті одержимо СЛАР з верхньою трикутною матрицею, у якій всі елементи нижче головної діагоналі дорівнюють нулю

$$\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n+1} \\ 0 & a_{22}^{(1)} & \dots & a_{2n+1}^{(1)} \\ \dots & \dots & \dots & \dots \\ 0 & \dots & 0 & a_{nn+1}^{(n-1)} \end{pmatrix}.$$

Запишемо вираз для невідомих  $x_k$  і перетворення елементів розширеної матриці системи, що узагальнюють формули (1.4)-(1.6),

$$x_k = \left( a_{k,n+1} - \sum_{j=k+1}^n a_{kj} x_j \right) / a_{kk}; \quad (1.7)$$

$$a_{kj}^{(m+1)} = a_{ij}^{(m)} - a_{ik}^{(m)} a_{kj}^{(m)}.$$

На цьому етапі ми ще не обчислили жодної компоненти вектора рішень  $\{x_k\}$ , але еквівалентними перетвореннями привели систему до такої форми, для якої легко обчислити всі компоненти рішень  $\{x_k\}$ .

Другий етап розв'язання називається зворотним ходом методу Гаусса й полягає в послідовному визначенні невідомих  $x_k$  за першою формулою з (1.7), починаючи з невідомих  $x_n$  і закінчуючи  $x_1$ .

Точність результатів буде визначатися точністю виконання арифметичних операцій при перетворенні елементів матриці. Для зменшення похибки при діленні на діагональний елемент (друга формула в (1.7)) рекомендується здійснити таку перестановку рівнянь, щоб поставити на діагональ найбільший за модулем із всіх елементів розглянутого стовпця. Така процедура називається вибором головного елемента стовпця. Кількість арифметичних операцій у методі Гаусса пов'язана з розмірністю системи й приблизно дорівнює  $2/3 n^2$ . Контроль отриманих рішень можна провести шляхом їхньої підстановки у вихідну СЛАР й обчислення нев'язок  $r_k$  різниць між правими й лівими частинами рівнянь:

$$r_k = a_{k,n+1} - \sum_{j=1}^n a_{kj} x_j . \quad (1.8)$$

При малій похибці рішень величини  $r_k$  будуть близькими до нуля (рис. 1.17).

### Метод Зейделя

Для розв'язання СЛАР (1.3) ітераційним методом Зейделя перетворимо її до виду:

$$\begin{cases} x_1 = (a_{1,n+1} - a_{11}x_1 - a_{12}x_2 - \dots - a_{1n}x_n) / a_{11} + x_1 ; \\ x_2 = (a_{2,n+1} - a_{21}x_1 - a_{22}x_2 - \dots - a_{2n}x_n) / a_{22} + x_2 ; \\ \dots \\ x_n = (a_{n,n+1} - a_{n1}x_1 - a_{n2}x_2 - \dots - a_{nn}x_n) / a_{nn} + x_n . \end{cases} \quad (1.9)$$

Задавши стовпець початкових наближень  $x_1^0, x_2^0, x_n^0$ , підставимо їх у праву частину системи (1.9) й обчислимо нові наближення  $x'_1, x'_2, x'_n$ , котрі знову підставимо в (1.9) і т. д.

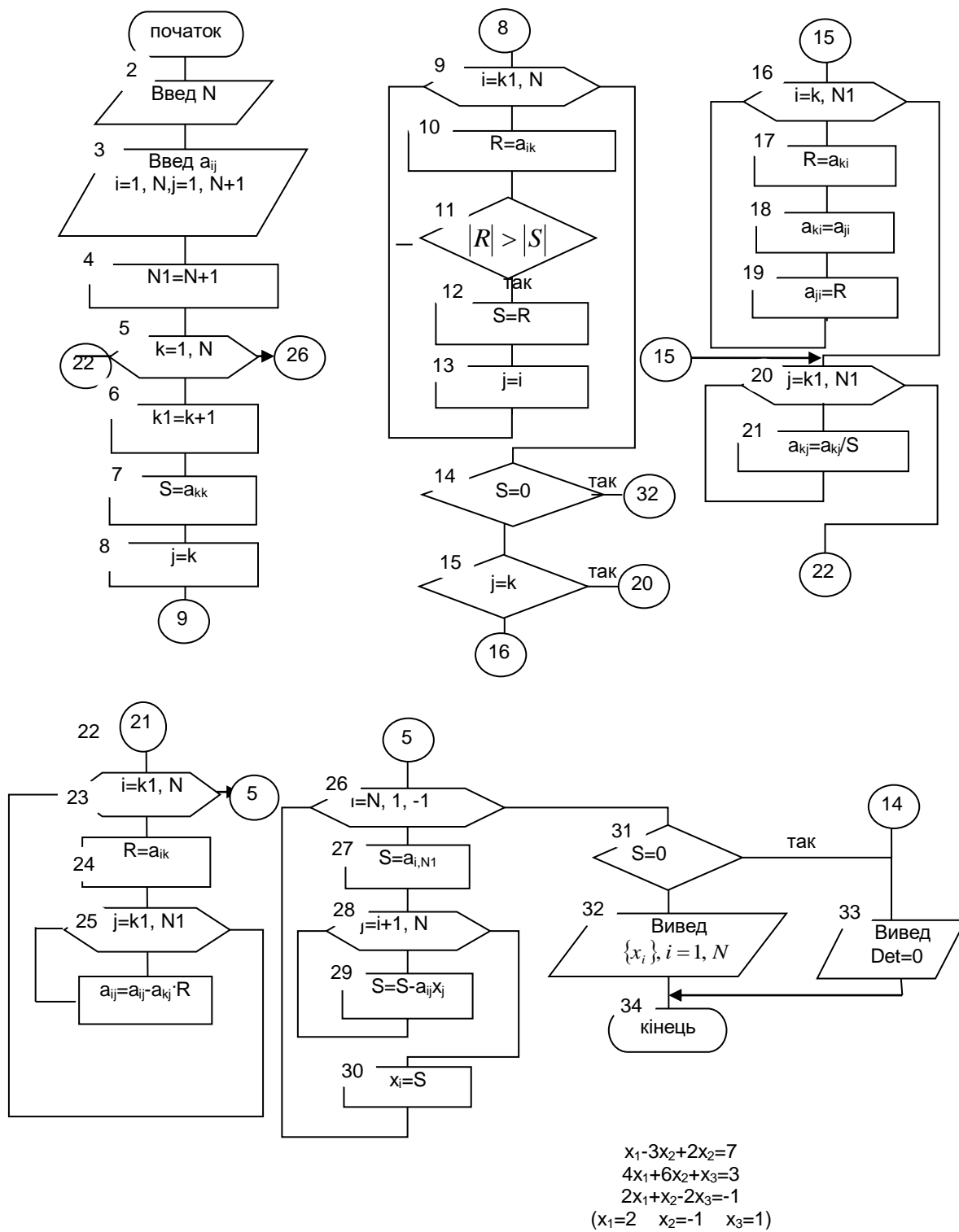


Рис. 1.17. Схема алгоритму розв'язання системи лінійних алгебраїчних рівнянь методом Гаусса

Якщо використати наближення до рішень, знайдені при виконанні поточної ітерації (що призводить до прискорення збіжності), одержимо метод Зейделя.

$$\begin{cases} x_1^{(m+1)} = \varphi_1(x_1^{(m)}, x_2^{(m)}, \dots, x_n^{(m)}); \\ x_2^{(m+1)} = \varphi_2(x_1^{(m+1)}, x_2^{(m)}, \dots, x_n^{(m)}); \\ \dots \\ x_n^{(m+1)} = \varphi_n(x_1^{(m+1)}, x_2^{(m+1)}, \dots, x_n^{(m)}) \end{cases} \quad (1.10)$$

Для збіжності ітераційних методів необхідно, щоб значення діагональних елементів у матриці СЛАР були переважними в порівнянні з іншими елементами. Умову збіжності можна забезпечити перетворенням вихідної матриці шляхом перестановки рівнянь і невідомих. Методи простих ітерацій Зейделя мають різні області збіжності. Ці методи можна застосовувати й до розв'язання систем нелінійних рівнянь.

Закінчуємо ітераційний процес, коли виконується умова

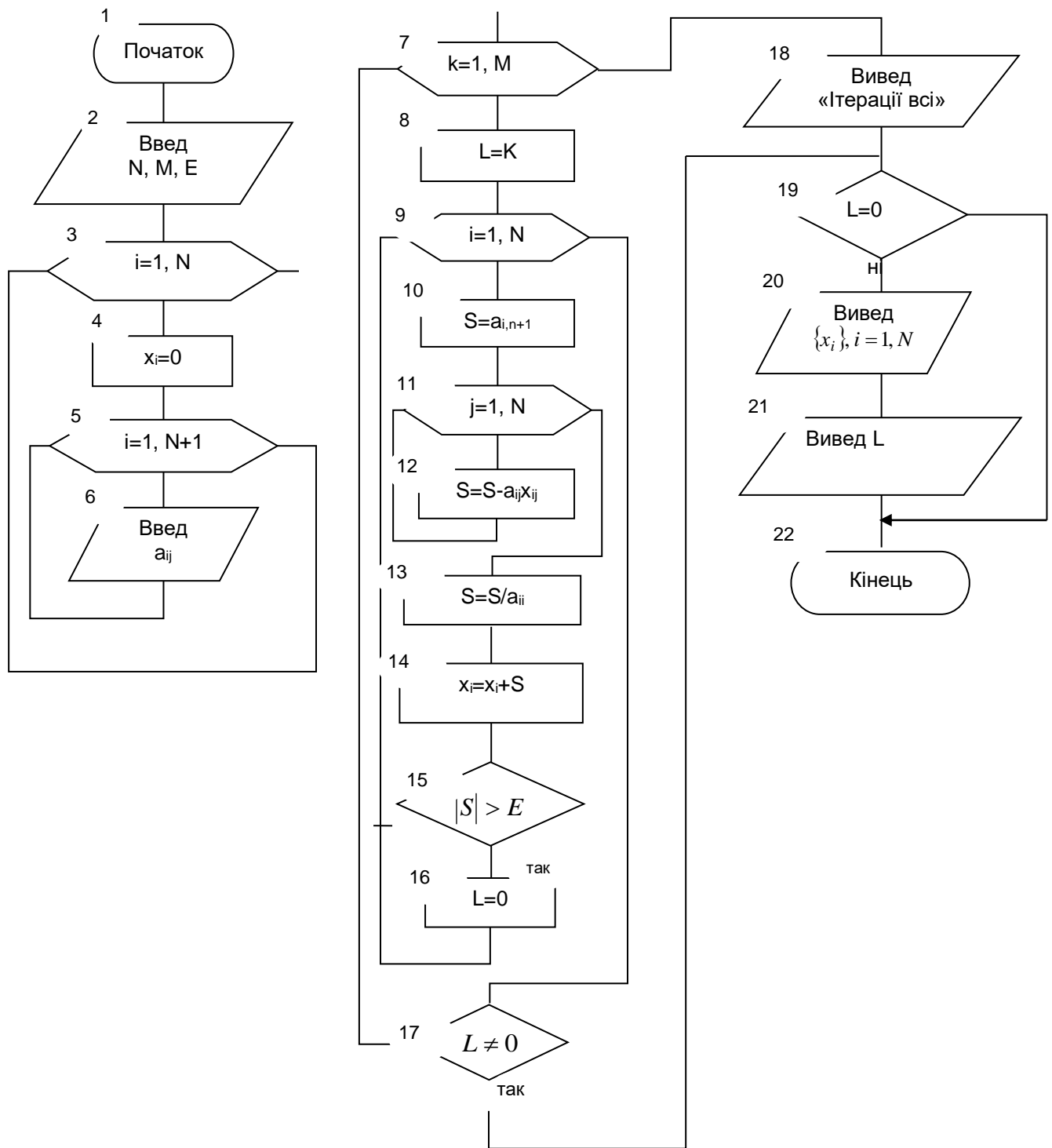
$$\left| x_k^{(m+1)} - x_k^{(m)} \right| < \varepsilon, \quad (1.11)$$

де  $\varepsilon$  — задана похибка  $k=1, 2, \dots, n$ .

Згідно з алгоритмом (рис. 1.18) ітерації можуть виконуватися максимально  $M$  разів (цикл по  $k$ ).

Якщо за  $M$  ітерацій умова (1.11) не виконується, то виводиться повідомлення "ітерації всі". Якщо рішення знайдене, то  $L$  приймає значення лічильника ітерацій, що виводиться після рішень, у протилежному випадку  $L=0$ .

Початкові наближення беремо рівними нулю.



$$\begin{cases} x_1 + 4x_2 + 2x_3 = 3 \\ 2x_1 + 7x_2 + 3x_3 = 4 \\ 8x_3 + x_3 = -5 \end{cases} \quad \varepsilon = 10^{-5} \quad L = 4 \quad (x_1; x_2 = -1; x_3 = 3)$$

Рис. 1.18. Схема алгоритму розв'язання систем лінійних алгебраїчних рівнянь методом Зейделя

## 1.4. Основи теорії оптимізації

### 1.4.1. Основні поняття теорії оптимізації

Методи оптимізації дозволяють вибрати найкращий варіант конструкції з усіх можливих варіантів. Звичайно це процес чи послідовність операцій, що дозволяють отримати уточнене рішення.

Розглядаючи деяку довільну систему, що описується  $m$  рівняннями з  $n$  невідомими, можна виділити три основних типи задач:

- якщо  $m = n$  — алгебраїчна задача. Така задача, звичайно, має одне рішення;
- якщо  $m > n$  — задача перевизначена і, як правило, не має рішення;
- якщо  $m < n$  — задача невизначена і має нескінченно багато рішень.

У практиці проектування часто доводиться мати справу з задачами третього типу. При цьому інженеру допомагає інтуїція, котра дозволяє сформулювати умови для вибору оптимального плану (варіанта).

### 1.4.2. Приклади оптимізаційних задач

#### План постачань підприємств

Є ряд підприємств, що споживають відомі види сировини, і є ряд сировинних баз, що можуть поставляти цю сировину підприємствам. Бази пов'язані з підприємствами шляхами сполучення (залізниця, водний, автомобільний транспорт), як правило, зі своїми тарифами. Вимагається розробити такий план постачань підприємств сировиною (з якої бази, в якій кількості, яку сировину доставити), щоб потреби в сировині були повністю забезпечені при мінімальних витратах на перевезення.

#### Будівництво ділянки магістралі

Споруджується ділянка залізничної магістралі. У нашому розпорядженні певна кількість ресурсів (засобів): людей, будівельних машин і механізмів, ремонтних майстерень, вантажних автомобілів тощо. Потрібно так спланувати будівництво, тобто призначити



черговість робіт, розподілити машини і людей по ділянках шляху, забезпечити ремонтні роботи, щоб воно було завершено в мінімально можливий термін при відповідній якості робіт.

### **Снігозахист залізниць**

У деяких районах замети снігом залізниць складають серйозну заваду руху поїздів. Будь-яке припинення руху призводить до економічних втрат. Існує ряд можливих засобів снігозахисту (профіль дороги, захисні щити, застосування снігоочищувачів і т. ін.), кожний з яких вимагає відомих витрат на спорудження та експлуатацію.

Відомі напрямки вітрів, є дані про частоту та інтенсивність снігопаду. Вимагається розробити найбільш ефективні економічні засоби снігозахисту з урахуванням втрат, пов'язаних із заметами.

### **Бібліотечне обслуговування**

Велика бібліотека обслуговує запити, що надходять від абонентів. У фондах бібліотеки є книги, що користуються підвищеним попитом, книги, на які запити надходять рідко, і, нарешті, книги, на які запити майже ніколи не надходять. Є ряд можливостей розподілу книг на стелажах і у сховищах, а також з диспетчеризації запитань із зверненнями до інших бібліотек. Потрібно розробити таку систему бібліотечного обслуговування, при якій запити абонентів задовольнялися б максимально.

## **1.4.3. Проектні параметри**

Це незалежні змінні параметрів, що повністю й однозначно визначають задачу проектування, котра розв'язується. Проектні параметри — невідомі величини, значення яких підраховуються в процесі оптимізації. Проектними параметрами можуть бути основні чи похідні величини, що служать для кількісного опису системи, наприклад, невідомі значення довжини, маси, часу, температури тощо.

Число проектних параметрів характеризує ступінь складності даної задачі проектування. Звичайно число проектних параметрів позначають через  $n$ , а самі проектні параметри через  $X$  з відповідними індексами. Таким чином,  $n$  проектних параметрів задачі будемо позначати через  $X_1, X_2, X_3, \dots, X_n$ .

#### 1.4.4. Цільова функція

Це вираз, для якого інженер-дослідник має визначити *max* або *min*<sup>1</sup>. Цільова функція дозволяє кількісно порівняти альтернативні рішення задачі оптимізації.

З математичної точки зору, цільова функція описує деяку  $(n+1)$ -вимірну поверхню. Її значення визначається проектними параметрами  $F = F(X_1, X_2, X_3, \dots, X_n)$ .

Прикладами цільових функцій, що часто зустрічаються в інженерній практиці, є *вартість, вага, міцність, габарити, ККД* тощо.

Якщо маємо тільки один проектний параметр, то цільову функцію можливо представити кривою на площині (рис. 1.19). Якщо проектних параметрів два, то цільова функція буде зображуватися функцією поверхні в просторі трьох вимірів (рис. 1.20). При трьох та більше проектних параметрах поверхні, що задаються цільовою функцією, називають гіперповерхнями. Вони не піддаються зображенню звичайними засобами.

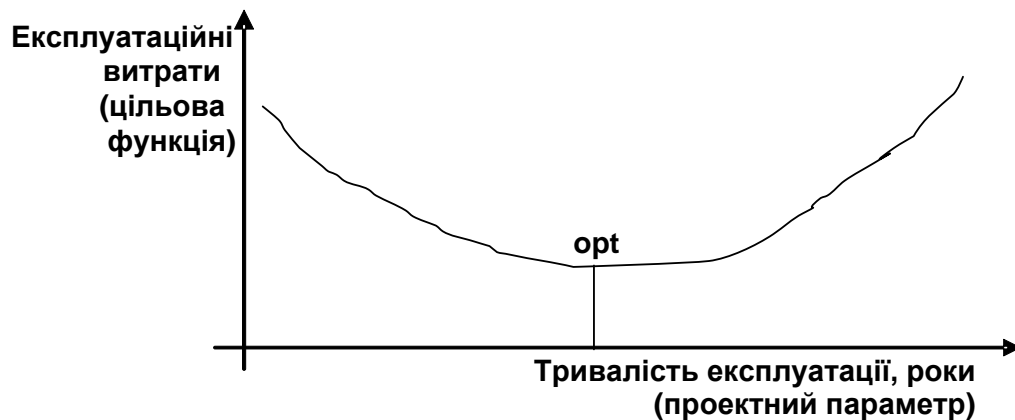


Рис. 1.19. Приклад одновимірної цільової функції

<sup>1</sup> *max* – найбільше значення цільової функції, *min* – найменше значення цільової функції.

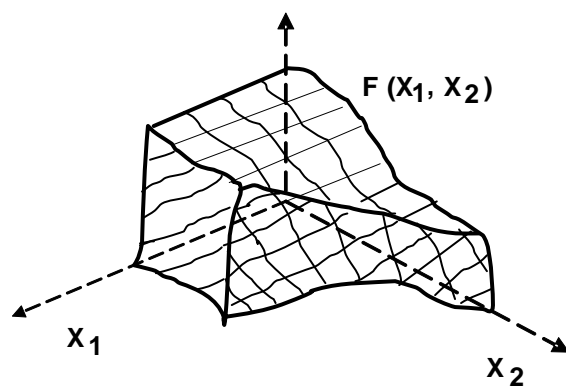


Рис. 1.20. Приклад двовимірної цільової функції

Цільова функція в деяких випадках може приймати найнесподіваніші форми. Наприклад, може приймати тільки цілочисельні значення (кількість зубів шестерні), іноді два значення "так" - "ні".

У ряді задач оптимізації виникає потреба введення більш ніж однієї цільової функції. Наприклад, *max* міцність, *min* вага, *min* вартість при проектуванні літака. У таких випадках потрібно вводити систему пріоритетів і поставити у відповідність до кожної цільової функції деякий безрозмірний множник. У результаті з'являється "функція компромісу", що дозволяє в процесі оптимізації користуватися єдиною складовою цільовою функцією.

#### 1.4.5. Знаходження екстремальних значень

Одні алгоритми оптимізації придатні для знаходження *max*, інші — для знаходження *min*. Однак незалежно від типу розв'язуваної задачі на екстремум можна користуватися одним і тим же алгоритмом, тобто задачу мінімізації можна легко перетворити на задачу зі знаходження *max*, змінивши знак цільової функції на зворотній. Це можна проілюструвати наступним чином (рис. 1.21).

Змінивши знак цільової функції на протилежний, ми перетворимо задачу пошуку *max* на задачу пошуку *min*.

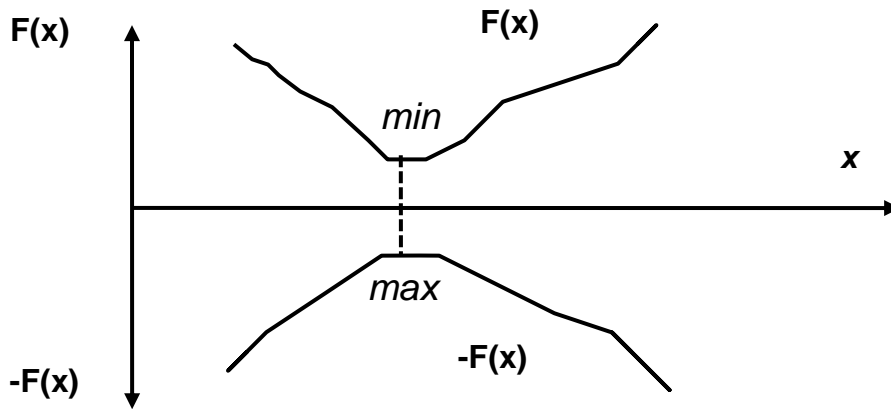


Рис. 1.21. Пошук *min* (*max*) цільової функції

### 1.4.6. Простір проектування

Так називають область, що визначається всіма  $n$ -проектними параметрами. Простір проектування не такий великий, як може здатися, оскільки він звичайно обмежений рядом умов, що пов'язані з фізичною суттю задачі. Обмеження можуть бути настільки сильними, що задача не буде мати жодного задовільного рішення. Обмеження поділяються на дві групи :

- обмеження — рівності;
- обмеження — нерівності.

### 1.4.7. Обмеження

Це залежність між проектними параметрами, котрі повинні враховуватися при знаходженні рішення. Вони можуть мати вигляд рівностей:

$$\left\{ \begin{array}{l} C_1 ( X_1, X_2, X_3, \dots, X_n ) = 0; \\ C_2 ( X_1, X_2, X_3, \dots, X_n ) = 0; \\ \dots \\ C_k ( X_1, X_2, X_3, \dots, X_n ) = 0. \end{array} \right.$$

Якщо яке-небудь з цих відношень можна визначити відносно одного з проектних параметрів, то це дозволяє виключити даний

параметр з процесу оптимізації. Тим самим зменшується число вимірів простору проектування і спрощується розв'язання задачі.

**Обмеження нерівності** — це особливий вид обмежень. Їх може бути скільки завгодно. Вони мають вигляд

$$\left\{ \begin{array}{l} R_1 \leq (X_1, X_2, X_3, \dots, X_n) \leq Z_1; \\ R_2 \leq (X_1, X_2, X_3, \dots, X_n) \leq Z_2; \\ \dots \\ R_k \leq (X_1, X_2, X_3, \dots, X_n) \leq Z_k. \end{array} \right.$$

### 1.4.8. Локальний оптимум

Так називається точка простору проектування, в якій цільова функція має найбільше (найменше) значення в порівнянні з її значеннями в усіх інших точках її найближчого оточення.

На рис. 1.22 показана одновимірна цільова функція, що має два локальних оптимуми. Найчастіше простір проектування містить багато локальних оптимумів, і слід бути обережним, щоб не прийняти перший із них за оптимальне рішення задачі.

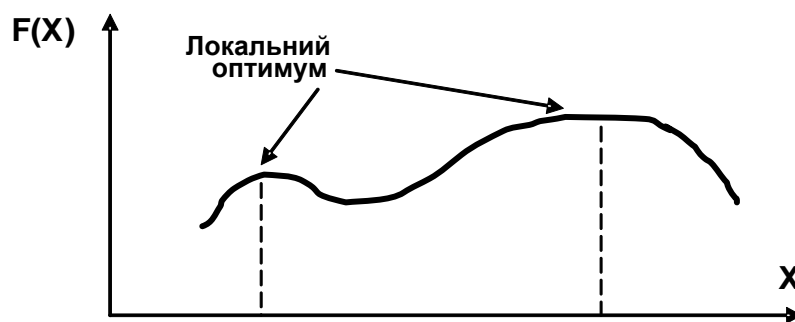


Рис. 1.22. Локальні оптимуми

### 1.4.9. Глобальний оптимум

Глобальний оптимум — це оптимальне рішення для всього простору проектування. Воно краще за інші рішення, котрі відповідають локальним оптимумам, і тому саме його шукає конструктор.

Трапляються випадки декількох рівних глобальних оптимумів, розташованих у різних частинах простору проектування.

#### 1.4.10. Класифікація задач оптимізації

Оптимізаційні задачі можна поділити на групи в залежності:

а) від кількості цільових функцій, що оптимізуються, — *однокритеріальні* та *багатокритеріальні* задачі оптимізації;

б) того, чи лежить значення оптимального параметра на межі допустимої області або ні, — *безумовні* та *умовні* задачі оптимізації;

в) кількості локальних екстремумів — *одноекстремальні* (один мінімум або максимум) та *багатоекстремальні* задачі оптимізації;

г) кількості ( $n$ ) оптимальних параметрів — *однопараметричні* ( $n=1$ ) — одновимірна оптимізація або *багатопараметричні* ( $n \geq 2$ ) — багатовимірна оптимізація;

д) виду цільової функції та системи обмежень (задача *лінійного програмування* — лінійна цільова функція та система лінійних обмежень, задача *нелінійного програмування* — нелінійна цільова функція та система лінійних обмежень тощо);

е) характеру задачі оптимізації — статичні та динамічні (*задачі динамічного програмування*).

#### 1.4.11. Задача вибору оптимальних параметрів контейнера

Нехай треба спроектувати прямокутний контейнер об'ємом  $1 \text{ м}^3$ . Контейнер призначений для перевезення сипучих вантажів. Бажано, щоб на виготовлення таких контейнерів йшло якомога менше матеріалу (за умови постійної товщини це означає, що площа поверхні повинна бути мінімальною), тому що при цьому контейнер буде дешевим. Для того щоб контейнер легше було брати автозавантажувачем, його ширина повинна бути не менше ніж  $1,5 \text{ м}$  (рис. 1.23).

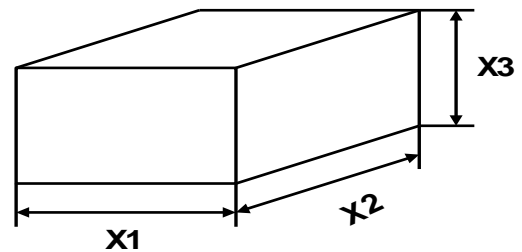


Рис. 1.23. Контейнер для вантажів

Сформулюємо цю задачу у вигляді, зручному для застосування алгоритму оптимізації.

**Проектні параметри**  $X_1, X_2, X_3$ .

**Цільовою функцією**, котру треба мінімізувати, є площа бічної поверхні контейнера:  $S = 2(X_1 X_2 + X_2 X_3 + X_1 X_3)$ , м<sup>2</sup>.

**Обмеження рівняння**: Об'єм  $= X_1 X_2 X_3 = 1$  м<sup>3</sup>.

**Обмеження нерівності**:  $X_1 \geq 1.5$  м.

Спостережливий конструктор помітить, що обмеження рівняння завдяки своїй простоті дозволяє зменшити вимірність задачі.

Дійсно, якщо  $X_3 = 1 / (X_1 X_2)$ , то параметр  $X_3$  можна виключити зі списку проектних параметрів. Знову формулюючи задачу, маємо: **проектні параметри** —  $X_1, X_2$ .

**Цільова функція**:  $S = 2(x_1 x_2 + \frac{1}{x_1} + \frac{1}{x_2})$ , м<sup>2</sup>.

**Обмеження рівняння** немає.

**Обмеження нерівності**  $X_1 \geq 1.5$  м.

Після того як задача сформульована стандартним способом, її можна розв'язати будь-яким методом. Якщо скористатися звичайним визначенням мінімуму і прийняти  $\frac{\partial S}{\partial x_1} = \frac{\partial S}{\partial x_2} = 0$ , то отримаємо відповідь  $X_1 = X_2 = X_3 = 1$  м.

Але при такому розв'язанні не виконано обмеження нерівності  $X_1 \geq 1.5$  м, тобто таке рішення відкидається.

Цей приклад демонструє важливу обставину того, що у зв'язку з введенням додаткових умов оптимальне рішення може відповідати точці, в якій локальний градієнт не дорівнює нулю.

Тому, щоб повністю розв'язати задачу, необхідно використати спеціальні методи розв'язання задач оптимізації.

## 1.5. Методи одновимірного пошуку

### 1.5.1. Суттєвість методів одновимірного пошуку

Задачу одновимірної оптимізації можна поставити наступним чином. Значення проектного параметра  $x$  повинні знаходитися в інтервалі  $a \leq x \leq b$ . На початку процесу оптимізації цей інтервал має довжину  $b-a$  (рис. 1.24). Нехай необхідно відшукати *min* цільової функції  $F(x)$ .

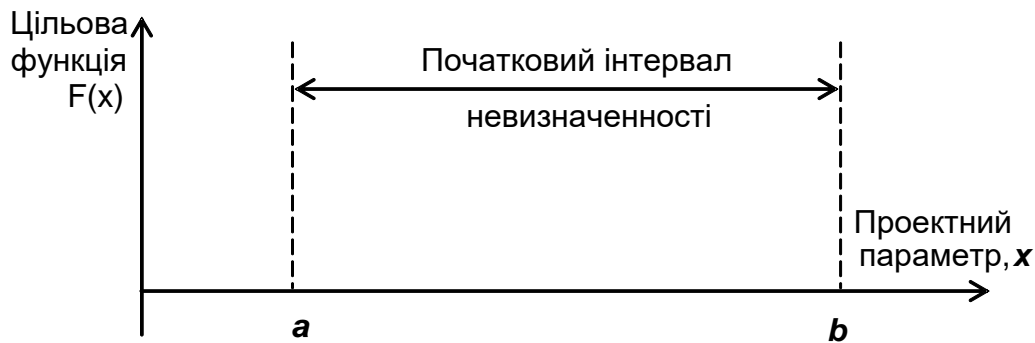


Рис. 1.24

Обчисливши значення цільової функції  $F_1$  і  $F_2$  при значеннях  $X_1$  і  $X_2$  в означеному інтервалі, можна звужити інтервал невизначеності. Наприклад, для випадку обчислення двох значень цільової функції на інтервалі  $[a;b]$  можливі такі варіанти (рис. 1.25-1.27):

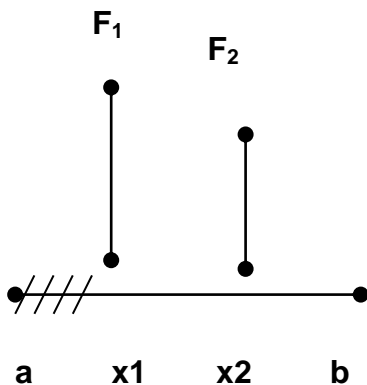


Рис. 1.25

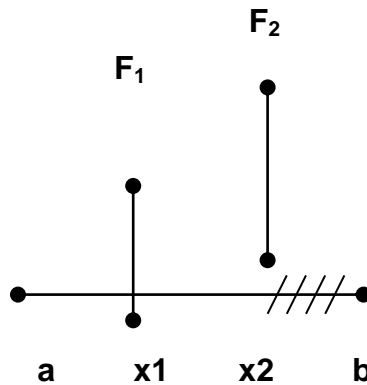


Рис. 1.26

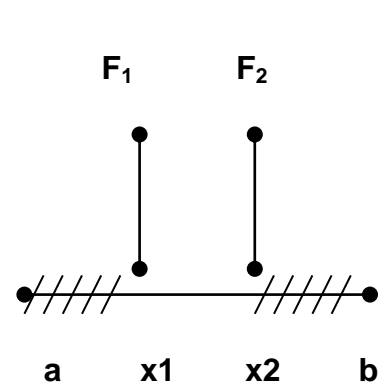


Рис. 1.27



Порівнявши обчислені значення цільової функції в точках  $X_1$  і  $X_2$ , можна відкинути той інтервал, в якому завідомо буде відсутній *min*. При цьому:

- якщо  $F(x_1) > F(x_2)$ , відкидаємо  $[a, x_1]$ ;
- якщо  $F(x_1) < F(x_2)$ , відкидаємо  $[x_2, b]$ ;
- якщо  $F(x_1) = F(x_2)$ , відкидаємо  $[a, x_1]$ ,  $[x_2, b]$ , бо на цих відрізках не може бути *min*, інакше в цьому випадку функція не була б унімодальною.

У цьому випадку вимога безперервності до функції не обов'язкова, а достатньо, щоб  $F(x)$  була унімодальною, тобто була тільки одна точка *min*.

У залежності від стратегії пошуку точок  $X_1$  і  $X_2$  на інтервалі розроблені різноманітні засоби систематичного звуження інтервалу, що відрізняються швидкістю стягування інтервалу. Найбільше розповсюдження знайшли такі методи (засоби):

- загального пошуку;
- ділення інтервалу навпіл;
- дихотомії;
- золотого перетину.

### 1.5.2. Метод загального пошуку

Природним засобом звуження інтервалу невизначеності для одновимірної функції є ділення його на декілька рівних частин з наступним обчисленням цільової функції у вузлах отриманої сітки (рис. 1.28).

У результаті інтервал невизначеності звужується до двох кроків сітки. Звичайно говорять про розбиття інтервалу невизначеності. Ефективність методу загального пошуку падає при зменшенні інтервалу невизначеності.

Розбиття інтервалу невизначеності характеризується коефіцієнтом розбиття —  $f$ . Поділивши інтервал невизначеності на  $N$  частин, отримаємо  $N+1$  вузол і тоді  $f = \frac{2}{N+1}$ .

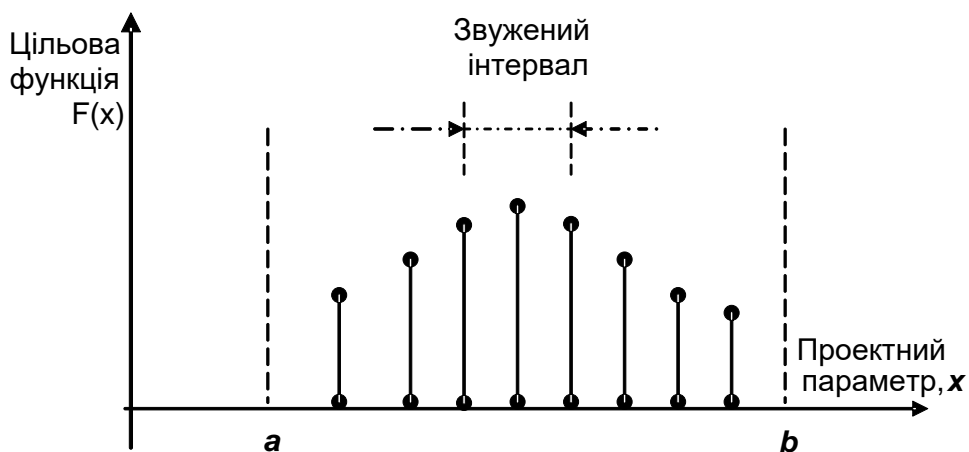


Рис. 1.28

Розрахунки показують, що для  $f=0.01$  треба обчислити цільову функцію в 199 точках. При  $f=0.001$  кількість розрахунків буде становити  $N=1999$ .

Існує більш ефективний шлях — необхідно спочатку взяти  $f=0.1$ , обчислити 19 значень цільової функції, а після цього на скороченому інтервалі взяти  $f=0.1$ , тобто виконати ще 19 обчислень цільової функції і в підсумку за 38 обчислень буде отримане значення  $f=0.01$  замість 199 обчислень.

Схема алгоритму методу загального пошуку з визначенням максимуму функції і проектного параметра  $X$ , при якому він досягається, наведено на рис. 1.29.

У цілому задачу пошуку екстремуму можна порівняти з пошуком в озері найглибшого місця. При кожному замірі одержується нова інформація. Причому, якщо в наступному замірі глибина виявилася більше, ніж в попередньому, то інформація корисна, а якщо навпаки — то ні. Постійно розробляються засоби пошуку, що дозволяють швидше знайти оптимальне рішення.

### 1.5.3. Ділення інтервалу навпіл

Користуючись тією ж методикою, що і для загального пошуку, але враховуючи значення функції в підінтервалах неоднакову кількість разів, можна додатково підвищити ефективність пошуку. Враховуючи  $N$  значень функції на  $i$  інтервалах, що послідовно

звужуються, отримаємо для коефіцієнта розбиття невизначеності

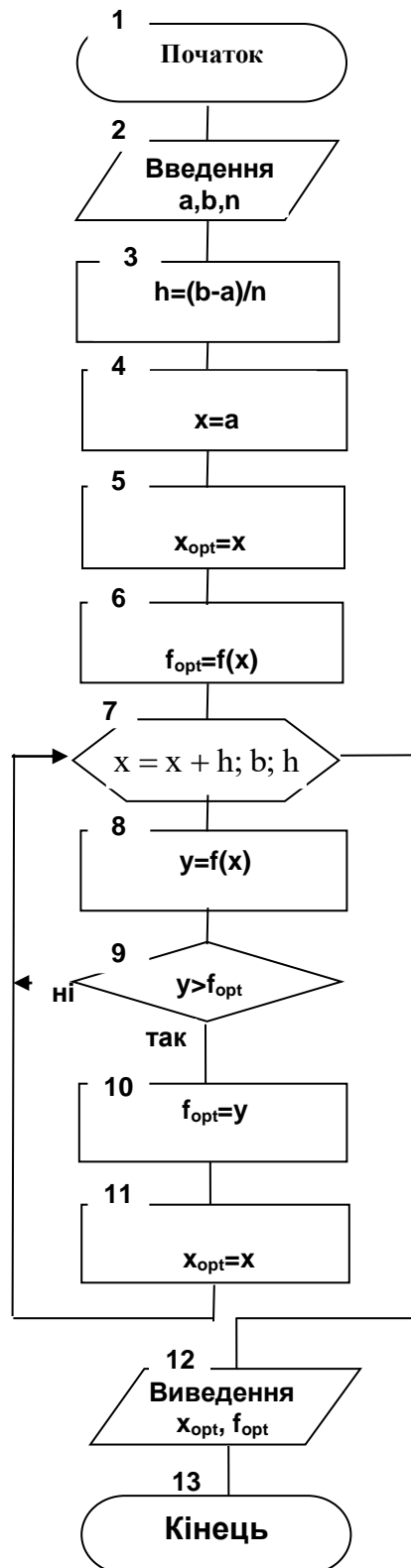
$$f = \left(\frac{2}{N+1}\right)^i.$$


Рис. 1.29. Алгоритм методу загального пошуку

При такому засобі пошуку цільову функцію вираховують  $J$  раз, причому  $J = N \cdot i$ .

За допомогою співвідношення  $i = J/N$  і виразу для інтервалу невизначеності можна знайти  $J$ :

$$J = \frac{N \ln(1/f)}{\ln\left(\frac{N+1}{2}\right)}$$

Інтервал невизначеності кожний раз ділиться навпіл, звідси і засіб, названий методом ділення інтервалу навпіл.

#### 1.5.4. Метод дихотомії

Обчислення цільової функції у двох точках інтервалу невизначеності дозволяє звужити його. Можна таким чином вибрати ці точки, що інтервал невизначеності буде мінімальним.

Реалізуючи метод дихотомії, зробимо таким чином: прийемо  $Z_2 = \xi$ , причому досить мале. Визначимо значення цільової функції в точках  $X_1$  і  $X_2$ . Якщо  $F(X_1) > F(X_2)$ , то будемо розглядати інтервал  $[a, X_2]$  (рис. 1.30), якщо  $F(X_1) < F(X_2)$ , то будемо розглядати інтервал  $[X_1, b]$  (рис. 1.31). Далі ділимо отриманий інтервал навпіл, знаходимо  $X_{cp}$ , а також значення функції в точках  $X_{cp} + \xi/2$  та  $X_{cp} - \xi/2$ . Потім процедура повторюється, поки відстань між лівою і правою межами інтервалу невизначеності не стане менше  $\xi$ .

#### 1.5.5. Метод золотого перетину

З кожних трьох значень цільової функції, обчислених в інтервалі невизначеності, у подальшому використовується тільки два, а третє не дає додаткової інформації і в подальшому не використовується. У методі золотого перетину цільова функція вираховується в точках інтервалу невизначеності, розташованих таким чином, щоб обчислене значення цільової функції давало нову корисну інформацію. Золотий перетин (відкрив Евклід) полягає в поділу інтервалу  $a, b$  на дві нерівні частини у співвідношенні

$\frac{Z_1}{Z} = \frac{Z_2}{Z_1}$ , тобто більший відрізок так відноситься до всього інтервалу, як менший до більшого (рис. 1.32).

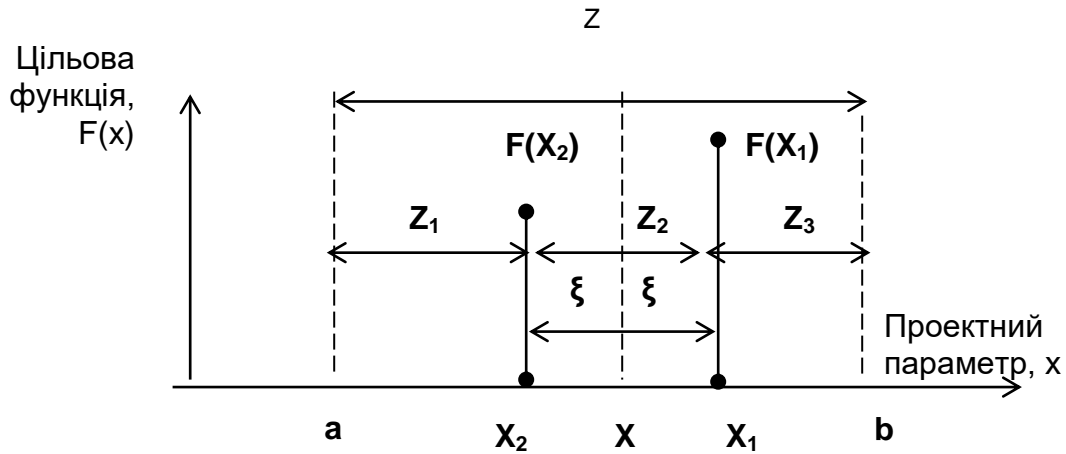


Рис. 1.30

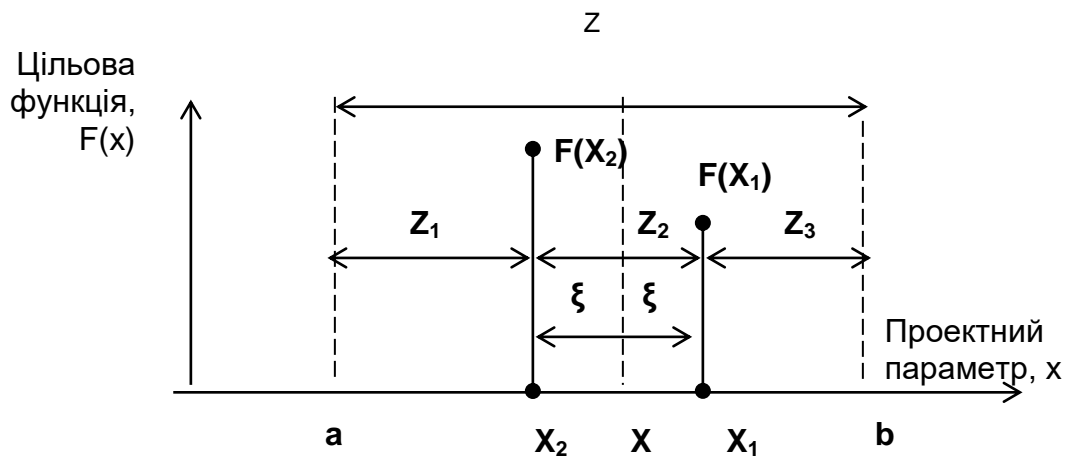


Рис. 1.31

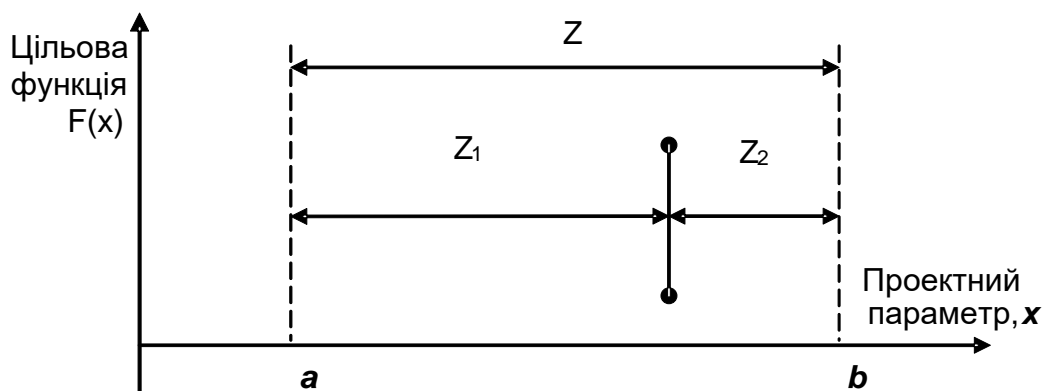


Рис. 1.32

Із співвідношення золотого перетину маємо

$$Z_1^2 = Z Z_2, \quad (1.12)$$

а також з урахуванням, що

$$Z = Z_1 + Z_2, \quad (1.13)$$

підставимо  $Z$  з виразу (1.13) до виразу (1.12) і поділивши на  $Z_1^2$ , отримаємо

$$\left(\frac{Z_2}{Z_1}\right)^2 + \frac{Z_2}{Z_1} - 1 = 0.$$

Для додатного кореня цього рівняння

$$\frac{Z_2}{Z_1} = 0.618.$$

Одне обчислення цільової функції дозволяє зменшити інтервал невизначеності в  $1/0.618$  раз. При обчисленні  $N$  значень цільової функції коефіцієнт розбиття інтервалу невизначеності складає  $f = 0.618^{N-1}$ . Алгоритм реалізації методу дихотомії наведено на рис. 1.33.

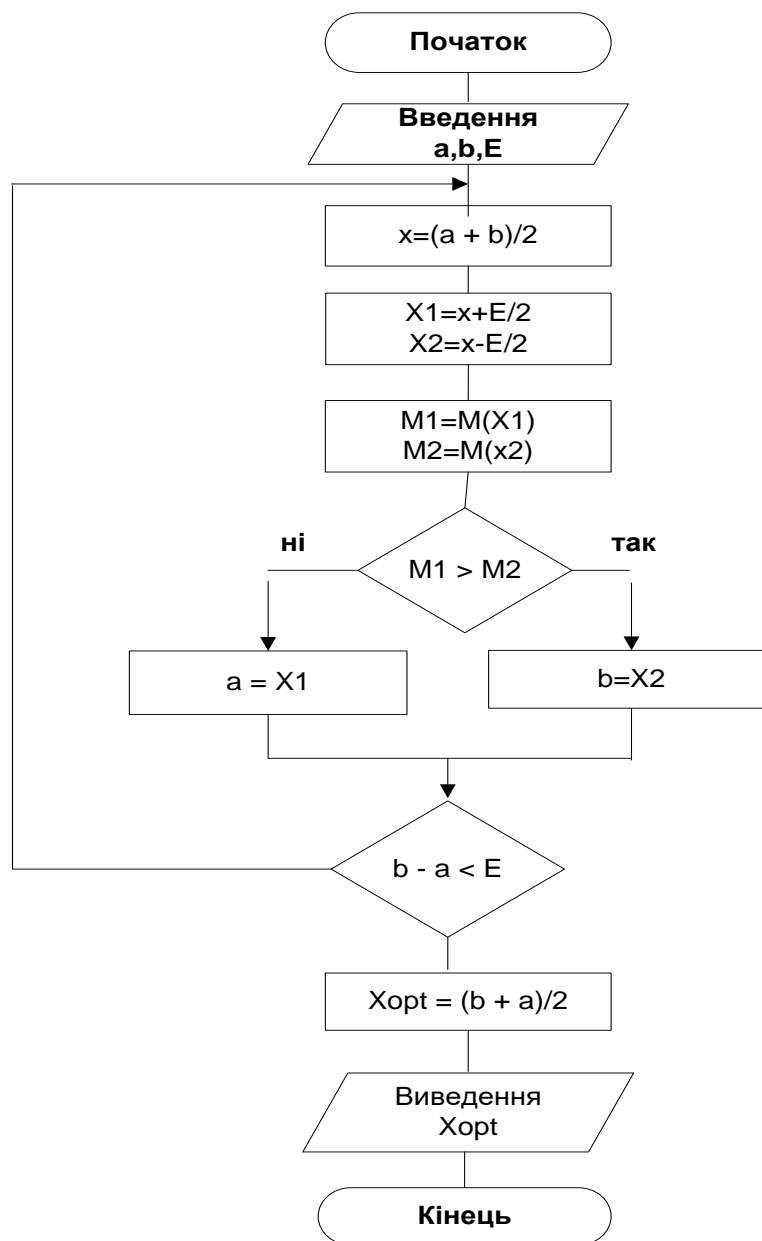


Рис. 1.33. Алгоритм реалізації методу дихотомії

## 1.6. Методи багатовимірного пошуку

Це методи для цільових функцій декількох змінних (проектних параметрів). Обсяг обчислень, необхідний для звуження інтервалу невизначеності в багатовимірному просторі, є статечною функцією, показник якої дорівнює вимірності простору. Так, якщо при коефіцієнті розбиття  $f=0.1$  для одновимірного простору необхідно

виконати 19 обчислень, то у випадку двовимірного простору це число складатиме  $19^2=361$ ;  $19^3=6\ 859$ ;  $19^4=130\ 321$ ;  $19^5=2\ 476\ 099$ .

Оскільки при виборі оптимальної конструкції нерідко доводиться мати справу з п'ятьма і більше змінними, серйозність труднощів, обумовлених багатомірністю, стає очевидною.

Методи оптимізації в багатовимірному просторі поділяються на дві групи — прямі і непрямі.

*Прямі методи* засновані на порівнянні значень цільової функції, що обчислюються, у різних точках, а *непрямі* — на використанні необхідних і достатніх умов математичного визначення максимуму і мінімуму функції.

*Стратегія прямих методів* — поступове наближення до оптимуму, а при використанні непрямих методів прагнуть знайти рішення, не досліджуючи неоптимальних точок.

### **1.6.1. Метод покоординатного підйому (спуску). Метод Гаусса-Зейделя**

Логічним розвитком методики одновимірного пошуку для багатовимірного пошуку була б послідовна зміна кожного проектного параметра доти, поки не буде досягнутий максимум (мінімум) цільової функції. По завершенні цієї процедури для всіх змінних можна повернутися до першого і подивитися, чи не можна ще більш удосконалити розв'язання.

Цей метод, названий методом покоординатного підйому (спуску), не завжди дозволяє знайти оптимальне рішення. Метод не застосовують у випадку, якщо лінії рівня мають точки зламу (рис. 1.34). Подібний тип рельєфу цільової функції часто називають "яружним". Незважаючи на це метод покоординатного підйому часто використовують на першій стадії розв'язання задачі, застосовуючи далі більш складні методи. До переваг методу слід віднести можливість використання простих алгоритмів одновимірного пошуку, таких як метод золотого перетину.

Розглянемо функцію  $Q(x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)})$  при фіксованих значеннях  $x_2^{(0)}, x_3^{(0)}, \dots, x_n^{(0)}$  як функцію однієї змінної  $x_1$ . Знаходимо  $\min Q(x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)})$ . Значення  $x_1$ , що досягається  $\min$ , позначимо  $x_1^{(1)}$ :



$$Q(x_1^{(1)}, x_2^{(0)}, \dots, x_n^{(0)}) \leq Q(x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)}).$$

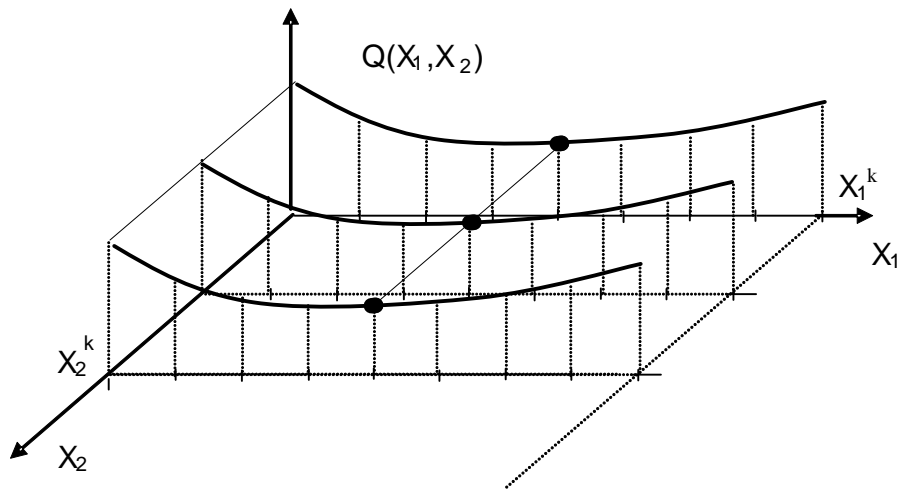


Рис. 1.34. Метод покоординатного спуску

Далі при фіксованих значеннях  $Q(x_1^{(1)}, x_3^{(0)}, \dots, x_n^{(0)})$  розглянемо  $Q(x_1^{(1)}, x_2, x_3^{(0)}, \dots, x_n^{(0)})$  як функцію однієї змінної  $x_2$ .

Знаходимо  $\min Q(x_1^{(0)}, x_2, x_3^{(0)}, \dots, x_n^{(0)})$ , значення  $x_2$ , що доставляє  $\min$ , позначимо  $x_2^{(1)}$ :

$$Q(x_1^{(1)}, x_2^{(1)}, \dots, x_n^{(1)}) \leq Q(x_1^{(1)}, x_2^{(0)}, \dots, x_n^{(0)}).$$

У результаті однієї ітерації за координатою спуску відбувається перехід з початкової точки  $x^{(0)} = (x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)})$  в точку  $x^{(1)} = (x_1^{(1)}, x_2^{(1)}, \dots, x_n^{(1)})$ .

Якщо при цьому виявляється, що  $Q(x_1^{(1)}) < Q(x_1^{(0)})$ , то виконується наступна ітерація покоординатного спуску, в якому значення початкової точки приймається  $x^{(1)}$ , одержуємо  $x^{(2)}$  і т. д. Цей процес продовжується доти, поки не виконається яка-небудь умова закінчення алгоритму, наприклад, виконається задана кількість ітерацій або  $|Q(x_1^{(k+1)}) - Q(x_1^{(k)})| < \xi$ , де  $\xi$  — задана точність. Зазначимо, що центральною ланкою розглянутого алгоритму є пошук  $\min$  функції однієї змінної.

У випадку двох параметрів задача зводиться до вигляду: знайти  $x^* = \arg \min Q(x)$ , якщо  $x \in x^0$ , де  $x^0 = \{x_1, x_2\}$ ;  $x_1^{(0)} \leq x_1 \leq x_1^{(k)}$ ;  $x_2^{(0)} \leq x_2 \leq x_2^{(k)}$ .

Схему алгоритму методу покоординатного спуску наведено на рис. 1.35.

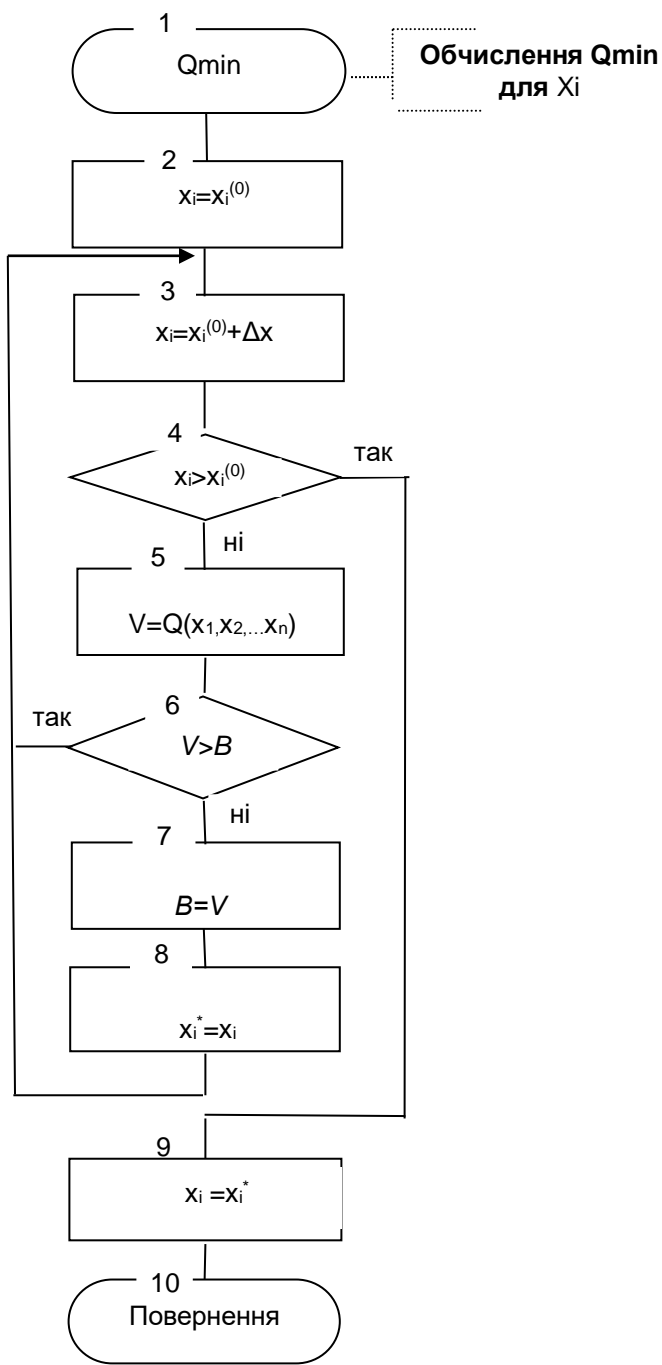
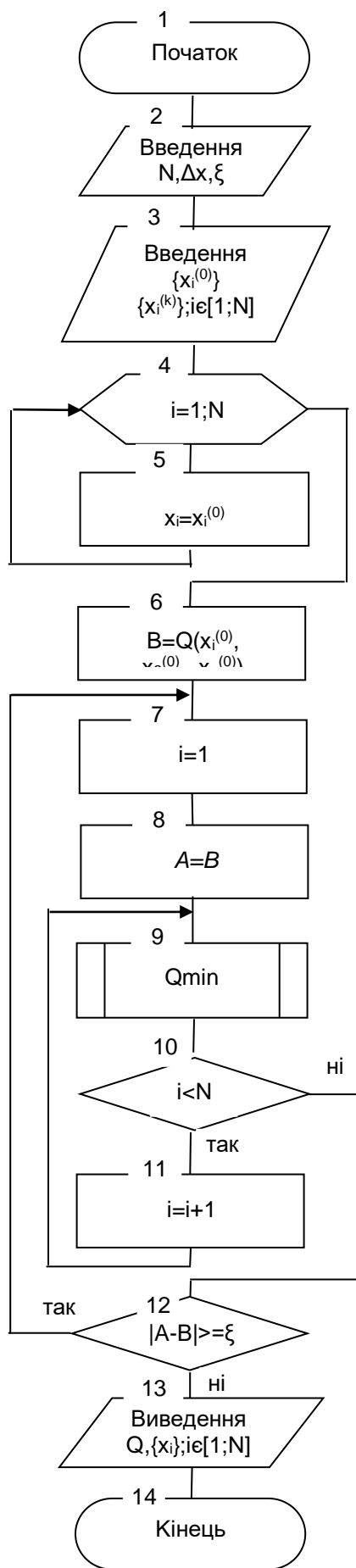


Рис. 1.35. Алгоритм та підпрограма реалізації методу покоординатного спуску

## 1.6.2. Градієнтні методи

У багатьох алгоритмах багатовимірної оптимізації використовується інформація про градієнти. Проілюструємо це положення наступним простим прикладом. Уявимо, що альпіністу зав'язали очі і сказали, що він повинний дістатися вершини "унімодалної" гори. Навіть нічого не бачачи, він може це зробити, якщо весь час буде рухатися нагору. Хоча будь-яка стежка, що веде нагору, врешті-решт приведе його до вершини, найкоротшою з них буде найкрутіша, якщо правда, альпініст не наштовхнеться на вертикальний обрив, котрий прийдеться обходити (математичним еквівалентом обриву на поверхні, утвореної цільовою функцією, є ті її місця, де поставлені умовні обмеження).

Метод оптимізації, в основу якого покладена ідея руху найкрутішою стежкою, називається методом найшвидшого підйому (найшвидшого спуску).

*Вектор градієнта* перпендикулярний лінії рівня і вказує напрямком до нової точки в просторі проектування. Отриману інформацію про напрямок градієнта можна використовувати різним чином для побудови алгоритму пошуку (рис. 1.36).

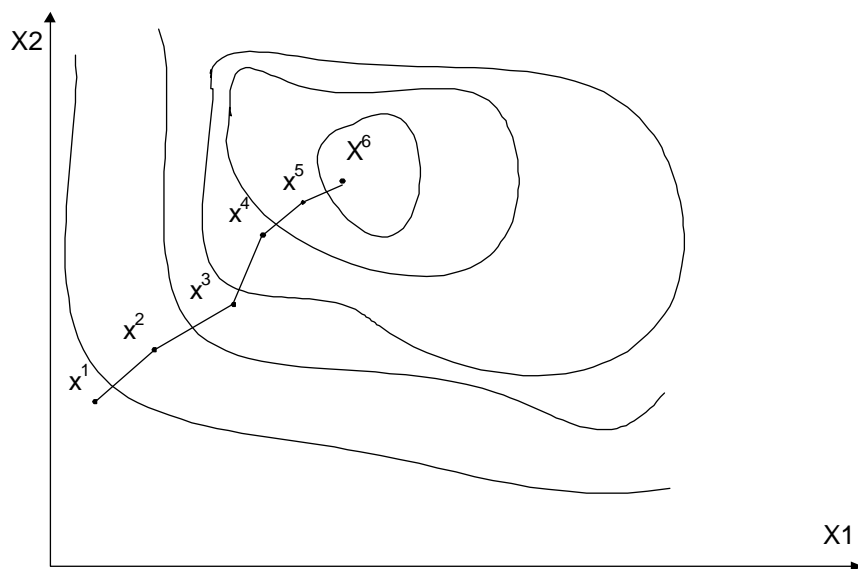


Рис. 1.36. Градієнтний метод

### 1.6.3. Східчастий найшвидший підйом (спуск)

Ряд методів пошуку заснований на зсуві на постійний крок у напрямку градієнта з наступним обчисленням цільової функції. Якщо її величина виявляється більше за попередню, обчислюється градієнт у новій точці, і вся процедура повторюється, причому часто при цьому крок збільшують. Якщо ж величина цільової функції не змінюється або убуває, то крок зсуву від попередньої точки зменшують і повторюють усю процедуру обчислень. Так роблять доти, поки подальше зменшення кроку уже не призводить до поліпшення результату.

Одержавши одновимірний оптимум у напрямку даного градієнта, знаходять новий градієнт і повторюють процес доти, поки наступні обчислення дозволяють поліпшувати отриманий результат. Головна перевага цього методу полягає в тому, що параметр  $S$  можна використовувати в якості незалежної змінної для пошуку за методом золотого перетину, і це забезпечує високу ефективність методу (рис. 1.37).

### 1.6.4. Метод випадкового пошуку

Цей метод запропонований математиком Бруксом і заснований на положеннях теорії ймовірності. Нехай простір проектування являє собою куб чи гіперкуб зі стороною, що дорівнює одиниці, і розділений на кубічні чарунки шляхом розподілу на 10 рівних частин кожної сторони куба, що відповідає одному з проектних параметрів.

При  $N=2$  число чарунок дорівнює 100, при  $N=3$  воно дорівнює 1000.

Одну з 10% найбільш перспективних чарунок складе 0,9. Якщо випадково вибрати дві чарунки, то ймовірність пропуску буде  $0,9^2$ , тобто 0,81.

Взагалі ймовірність перебування принаймні однієї чарунки з найбільш перспективних, частка яких дорівнює  $f$ , після  $N$  спроб складатиме  $P = 1 - (1-f)^N$ .



У табл. 1.1 зазначено, скільки чарунок треба вибрати випадково, щоб забезпечити задану ймовірність при заданій частці найбільш перспективних чарунок. З неї видно, що при випадковому виборі 44 чарунок імовірність досягнення  $f=0,1$  складає 99%. Це дуже непогано, якщо згадати, що для 100% забезпечення цільову функцію у випадку п'яти змінних довелося б обчислити 2476099 разів.

Таблиця 1.1

F	Імовірність			
	0.80	0.90	0.95	0.99
0.1	16	22	29	44
0.05	33	45	59	90
0.01	161	230	299	459

Метод випадкового пошуку має *дві переваги*.

По-перше, він придатний для будь-якої цільової функції незалежно від того, є вона унімодальною чи мультимодальною.

По-друге, імовірність успіху при спробах не залежить від вимірності розглянутого простору. Хоча цей метод не дозволяє безпосередньо знайти оптимальне рішення, він створює придатні передумови для застосування надалі інших методів пошуку. Тому його застосовують у сполученні з одним чи декількома методами інших типів.

Для розв'язання задач одновимірної оптимізації часто використовують метод випадкового пошуку Монте-Карло. Його суть зводиться до того, що в області пошуку оптимального значення відповідно до заданої щільності розподілення генеруються випадкові точки  $X_1, X_2, \dots$ . Для кожного значення  $X$  обчислюється значення функції, що підлягає оптимізації з запам'ятовуванням мінімуму для  $X$ . Схема алгоритму для пошуку  $\min$  цільової функції наведена на рис. 1.38.

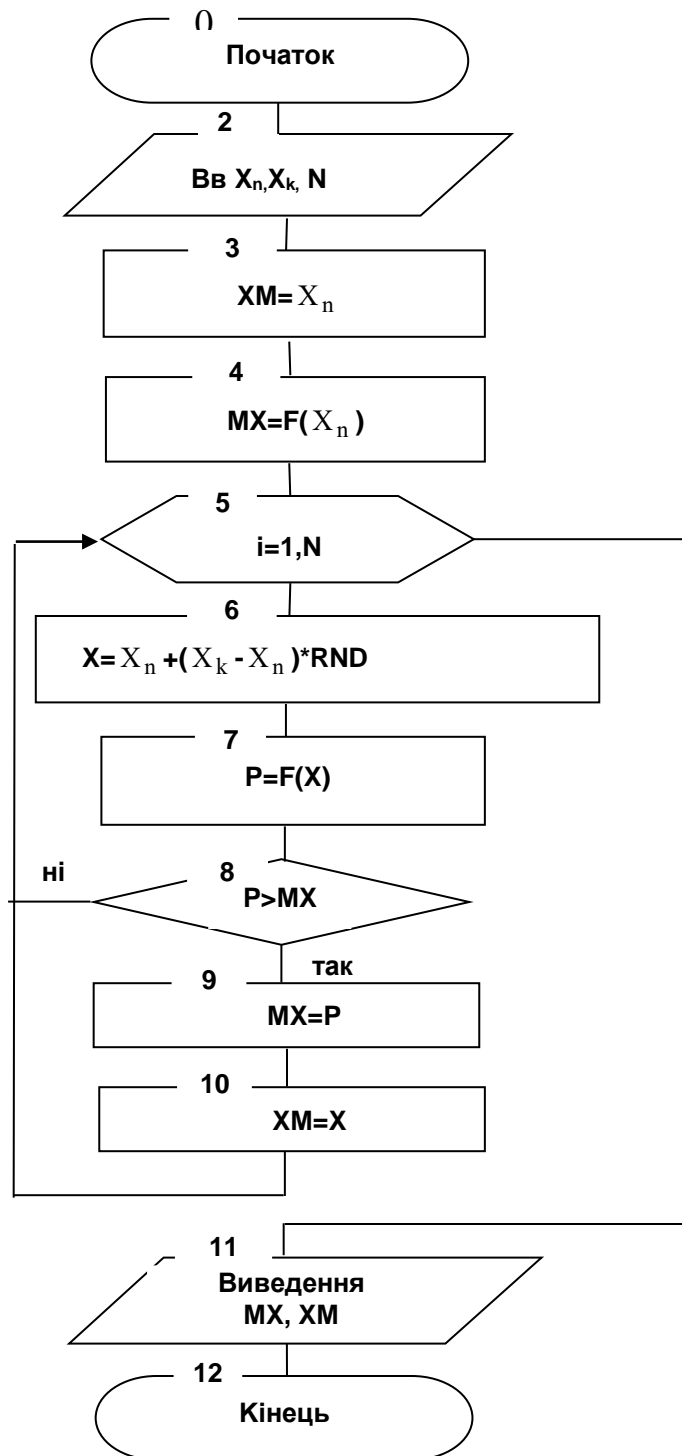


Рис. 1.38. Схема алгоритму пошуку максимального значення функції методом Монте-Карло

Отримання псевдовипадкового значення  $X$  в діапазоні від  $a$  до  $b$  відбувається за допомогою генератора випадкових чисел (функція RND), який є в кожній алгоритмічній мові. При цьому використовується формула

$$X = a + (b - a) \text{RND},$$

де  $a$  — початкове (ліве) значення інтервалу;  $b$  — кінцеве (праве) значення інтервалу.

Функція генерує випадкові числа за рівномірним законом розподілу в інтервалі від 0 до 1.

Алгоритм одержання випадкових чисел :

1. Взяти два довільних числа  $a_1, a_2$ .
2. Перемножити  $a_1 * a_2$ .
3. Взяти  $n$  розрядів добутку (це  $a_3$ ).
4. Перемножити  $a_2 * a_3$  і т. д.

### 1.6.5. Симплекс-метод

Симплексом називається  $N$ -мірна геометрична фігура, ребра якої являють собою прямі лінії, що перетинаються в  $N+1$  вершині. У двовимірному випадку це трикутник, у тривимірному — тетраедр.

Схеми пошуку з використанням симплексів засновані на спостереженні за зміною цільової функції в їхніх вершинах. Головним у цих схемах є процес відображення — перебування вершини нового симплекса, розташованої симетрично щодо площини, що проходить через одну зі сторін вихідного симплекса. Вибір напрямку пошуку вершини нового симплекса визначається положенням тієї вершини вихідного симплекса, у якій цільова функція має найгірше значення. Нова точка називається "доповненням" найгіршої точки. Якщо в тільки що отриманій вершині нового симплекса значення цільової функції виявляється гіршим, то алгоритм передбачає повернення у вихідну точку — вершину колишнього симплекса. Потім здійснюється перехід до тієї вершини колишнього симплекса, у якій цільова функція має







Якщо загальна кількість виробленого продукту дорівнює загальній кількості споживаного  $\sum_{i=1}^m a_i = \sum_{j=1}^n b_j$ , то систему обмежень складуть рівності:

$$\sum_{j=1}^n x_{ij} = a_i, i = \overline{1, m},$$

$$\sum_{i=1}^m x_{ij} = b_j, j = \overline{1, n}.$$

Відповідно до умов задачі зворотні перевезення не допускаються:  
 $x_{ij} \geq 0$ .

Якщо кількість виробленого продукту менше, ніж споживаного

$$\sum_{i=1}^m a_i < \sum_{j=1}^n b_j,$$

то система обмежень буде

$$\sum_{j=1}^n x_{ij} = a_i, i = \overline{1, m},$$

$$\sum_{i=1}^m x_{ij} \leq b_j, j = \overline{1, n}.$$

### ***Задача про використання ресурсів***

Розглянемо приклад реалізації задачі лінійного програмування.

Для здійснення грабарств виділена така техніка:

самоскиди — 12,

екскаватори — 8,

бульдозери — 16,

автогрейдери — 12, а також виділені команди для їх

обслуговування.

Наведені роботи можуть бути виконані землерийними комплексами двох типів:

**1-й тип**

самоскиди — 2,  
 екскаватори — 1,  
 бульдозери — 4,

продуктивність: 20 м<sup>3</sup>/год,

**2-й тип**

самоскиди — 2,  
 екскаватори — 2,  
 автогрейдери — 4,

продуктивність: 30 м<sup>3</sup>/год.

Яку кількість комплексів 1-го і 2-го типів треба організувати з виділених машин, щоб виконати роботу в найкоротший термін?

**Постановка задачі**

Нехай  $x_1$  — необхідна кількість комплексів 1-го типу, а  $x_2$  — 2-го типу, тоді *цільова функція* буде мати вигляд  $f = 20x_1 + 30x_2$   
 → **max**.

*Обмеження* для наявної техніки:

самоскиди —  $2x_1 + 2x_2 \leq 12$ ;  
 екскаватори —  $x_1 + 2x_2 \leq 8$ ;  
 бульдозери —  $4x_1 \leq 16$ ;  
 автогрейдери —  $4x_2 \leq 12$ .

При чому  $x_1 \geq 0$ ,  $x_2 \geq 0$ .

Зведемо задачу до канонічної форми, додавши в кожне рівняння по додатковій невідомій:

$f(x) = 20x_1 + 30x_2 + 0x_3 + 0x_4 + 0x_5 + 0x_6 \rightarrow \max$ ;  
 $2x_1 + 2x_2 + 0x_3 + 1x_4 + 0x_5 + 0x_6 = 12$ ;  
 $4x_1 + 0x_2 + 0x_3 + 0x_4 + 1x_5 + 0x_6 = 16$ ;  
 $0x_1 + 4x_2 + 0x_3 + 0x_4 + 0x_5 + 1x_6 = 12$ ;  
 $x_1 \geq 0, x_2 \geq 0, \dots, x_6 \geq 0$ .

Число рівнянь  $m$  менше за число невідомих  $n$ . Ми маємо незліченну безліч рішень ( $n - m$ ) *вільних невідомих*, що можуть приймати довільні значення, а інші  $m$  *базисних невідомих* виражаються через них.

Набір базисних невідомих називається *базисом*. Припустимий план, у якому всі вільні невідомі дорівнюють нулю, називається *базисним припустимим планом*, скорочено БПП.

Оптимальний план шукається шляхом поліпшення БПП.

Для нашої задачі БПП має вигляд:

$x_1 = 0, x_2 = 0, x_3 = 12, x_4 = 8, x_5 = 16, x_6 = 12$ .

З нього і починають поліпшення плану. Значення цільової функції при цьому дорівнює 0.

Схема алгоритму симплекс-методу (СМ) наведена на рис. 1.40.

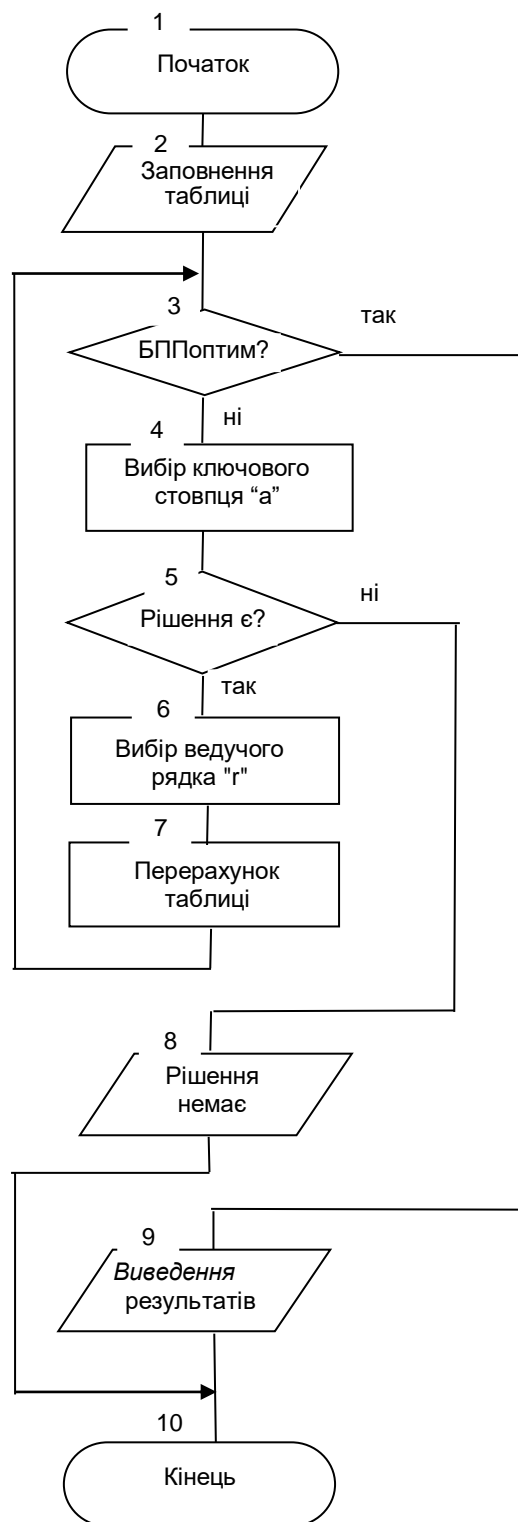


Рис. 1.40. Алгоритм симплекс-методу

Алгоритм починається з заповнення таблиці. Верхній рядок індексний з коефіцієнтами цільової функції (остання клітинка зі збереженням поточного значення цільової функції зі зворотним знаком) і заповнюється нулем. У нульовому стовпці зберігаються індекси базисних змінних.

Поліпшення БПП полягає в послідовному введенні в базис невідомих, у яких коефіцієнти в цільовій функції (індексний рядок) додатні.

0		20	30	0	0	0	0	0	
1	3	2	2	1	0	0	0	12	
2	4	1	2	0	1	0	0	8	
3	5	4	0	0	0	1	0	16	
4	6	0	4	0	0	0	1	12	

↑ "a" – ключовий стовпець  
↑ індекси базисних невідомих

← індексний рядок  
← цільова функція  
↖ "r" – ведучий рядок

### ***Перевірка на оптимальність***

Якщо в індексному рядку немає жодного додатного елемента, тоді БПП поліпшити не можна — він оптимальний.

### ***Пошук ключового стовпця***

Якщо в індексному рядку кілька додатних чисел, у базис бажано вводити ту невідому, у якої найбільше число в індексному рядку.

### ***Перевірка на можливість розв'язання задачі***

Якщо в ключовому стовпці немає жодного додатного числа, то задача не має рішення, тому що може бути знайдений план з як завгодно більшої числової функції.

### ***Пошук ведучого рядка***

Вибравши ключовий стовпець, ми вибрали невідому, котру треба вивести, а тепер необхідно вибрати ту невідому, що треба ввести, для чого елементи стовпця вільних елементів поділяються на додатні елементи в тих же рядках і серед отриманих елементів вибирається найменший, а рядок, у якому він знаходиться, називається ведучим.

Введемо умовні позначення:

$a$  — номер ключового стовпця,

$r$  — номер ведучого рядка.

Генеральний елемент (дозволяючий) знаходиться на перетині ключового стовпця і ключового рядка (із двох однакових обирається нижній).

Перерахування таблиці полягає в заміні змінної в базисі й у пов'язаних із заміною змінними в таблиці і перерахуванням усіх, крім "0", елементів за формулами:

$$S_{ij}^* = S_{ij} - \frac{S_{rj} * S_{ia}}{S_{ra}}, \text{ якщо } i \text{ не дорівнює } r,$$

$$S_{rj}^* = \frac{S_{rj}}{S_{ra}},$$

де  $S_{ij}$  — колишнє значення елемента;  $S_{ij}^*$  — перераховане значення.

У ведучому рядку індекс невідомої виводиться з базису, змінюється на індекс невідомої, що вводиться в базис, тобто  $S_{ra} = a$ .

Останній елемент 1-ого рядка

$$12 - \frac{12 * 2}{4} = 6.$$

	0	1	2	3	4	5	6	7	
0		20	0	0	0	0	-7.5	-90	
1	3	2	0	1	0	0	-0.5	6	
2	4	1	0	0	1	0	-0.5	2	"r"
3	5	4	0	0	0	1	0	16	
4	2	0	1	0	0	0	0.25	3	

"a"

---

БПП не оптимальний, оскільки в індексному рядку є додатне число, тому заново перерахуємо таблицю.

	0	1	2	3	4	5	6	7	
0		0	0	0	-20	0	2.5	-130	
1	3	0	0	1	-2	0	0.5	2	
2	1	1	0	0	1	0	-0.5	2	"r"
3	5	0	0	0	-4	1	2	8	
4	2	1	1	0	0	0	0.25	3	

"a"

---

Продуктивність зросла до 130 м<sup>3</sup>, але в індексному рядку є ще один додатний елемент, тому знову перерахуємо таблицю.

	0	1	2	3	4	5	6	7
0		0	0	0	-15	-1.25	0	-140
1	3	0	0	1	-1	-0.25	0	0
2	1	1	0	0	0	0.25	0	4
3	6	0	0	0	-2	0.5	1	4
4	2	0	1	0	0.5	-0.125	4	2

Тепер БПШ поліпшити не можна — він оптимальний.

Отже, рішення задачі таке:  $x_1=4$ ,  $x_2 = 2$ ,  $x_6 = 4$  — чотири автогрейдери не задіяні (так званий фіктивний комплекс).

## 1.7. Методи багатокритеріальної оптимізації

### 1.7.1. Постановка задачі багатокритеріальної оптимізації

У загальному випадку задача багатокритеріальної оптимізації формулюється як задача одночасної мінімізації деякої сукупності показників

$$f_1(x), f_2(x), \dots, f_m(x).$$

Варто помітити, що строго математично задача в такій постановці не має сенсу, тому що мінімуми окремих показників у загальному випадку досягаються при різних значеннях вектора  $X$ . Разом з тим математичні методи прийняття рішень разом з методами оптимізації можуть допомогти досліднику прийняти правильні розумні рекомендації в цьому випадку.

Існуючі методи багатокритеріальної оптимізації можуть бути зведені до двох груп:

**1-а група** припускає введення додаткових гіпотез, що дозволяють звести задачу багатокритеріальної оптимізації до задачі однокритеріальної. Цей прийом називається скаляризацією чи згортанням показників (критеріїв).

**2-а група** способів припускає скорочення безлічі вихідних варіантів рішень шляхом неформального аналізу цих варіантів.



## 1.7.2. Методи згортання показників

Найпростіший спосіб зведення багатокритеріальної задачі оптимізації до задачі однокритеріальної полягає у виділенні одного основного показника, наприклад  $f_1(x)$ , і переведенні інших (допоміжних) показників у розряд обмежень. Даний спосіб найбільш розповсюджений в інженерній практиці. Для його використання досить лише розумно призначити припустимі границі допоміжних показників.

*Спосіб лінійної згортки.* Суть цього способу полягає в переході від  $m$  показників  $f_i(x)$ ,  $i = 1, m$  до одного показника  $f(x)$  вигляду

$$f(x) = \sum_{i=1}^m \alpha_i f_i(x), \text{ де } \alpha_i \text{ — вагові важелі, що характеризують значущість}$$

відповідного показника і установлюють визначений компроміс між ними за рахунок ранжирування цілей за їх важливістю. Як правило,  $\alpha_i$  — додатні, нормовані тим чи іншим способом коефіцієнти, наприклад,

$$\sum_{i=1}^m \alpha_i = 1.$$

Призначення коефіцієнтів  $\alpha_i$  є тією додатковою гіпотезою, що зводить вихідну задачу з багатьма показниками до задачі з одним показником. Сам процес призначення  $\alpha_i$  є неформальним актом. Він вимагає ретельного аналізу всієї задачі. Остаточне призначення коефіцієнтів  $\alpha_i$  часто здійснюється шляхом послідовних наближень на основі попередніх розв'язань задачі при різних значеннях  $\alpha_i$ .

## 1.7.3. Мінімаксна згортка

Дуже часто в задачах з багатьма показниками вдається сформулювати деяку систему показників  $f_i^*$ ,  $i=1, m$ , що є по суті оцінками зверху для розглянутих показників

$$f_i(x) \leq f_i^*, i=1, m.$$

Якщо тепер як міру близькості показників до своїх контрольних значень  $f_i$  використовувати таку функцію максимуму:

$$f(x) = \max \frac{f_i(x)}{f_i^*}, i = 1, m,$$

то задача скалярної оптимізації буде мати вигляд

$$x^* = \arg \min \max \frac{f_i(x)}{f_i^*}, i = 1, m.$$

У такий спосіб багатокритеріальна задача оптимізації зводиться до пошуку гарантованого рішення в змісті близькості до контрольних значень вихідних показників.

#### 1.7.4. Метод штрафних функцій

При використанні цього методу оптимізації вводять штрафну функцію, з якою утвориться нова складена цільова функція

$$F(X) = M(X) + G(X),$$

де  $G(X)$  — штрафна функція.

Якщо всі умови задовольняються, то функції  $M(X)$  і  $F(X)$  мають той самий мінімум. Якщо хоча б одна умова не задовольняється, то цільова функція здобуває нескінченно велике значення, дуже далеко від мінімального значення  $M(X)$ . Тим самим на кожну конструкцію, що не відповідає поставленим умовам, "накладається штраф".  $G(X) = 0$  у всіх точках дотримання обмежень і прагне до нескінченності в точках, де умови не виконуються. Можна розв'язувати задачу, послідовно вводячи кілька штрафних функцій.

#### 1.7.5. Використання принципу Парето

Поряд зі згортанням багатьох показників до одного можливі й інші шляхи розв'язання задач багатокритеріальної оптимізації, наприклад, з використанням принципу Парето.

Сутність даного методу полягає в тому, що виключаються з неформального аналізу такі варіанти рішення, що свідомо є недопустимими. Наприклад, припустимо, що  $X'$  і  $X''$  два можливі варіанти розв'язання задачі, такі, що має місце нерівність  $f(X') \leq f(X'')$ ,  $i = 1, m$ .

У цьому випадку очевидно, що рішення  $X'$  переважніше за  $X''$ . Значить усі вектори  $X''$ , що задовольняють цю умову, можуть бути виключені з розгляду. Тому неформальному аналізу повинні бути піддані лише вектори, для яких не існує кращих векторів. Такі вектори називають векторами, що не поліпшуються. Безліч векторів, що не поліпшуються, називається безліччю Парето.

Звернемося до найпростішого прикладу з двома цільовими функціями  $f_1(x)$  і  $f_2(x)$ . У цьому випадку кожному значенню  $X$  буде відповідати точка на площині  $f_1, f_2$ . Рівності  $f_1=f_1(x)$  і  $f_2=f_2(x)$  визначають параметрично деяку криву  $a, b, c, d, e, g, h$  на цій площині. Однак безлічі Парето відповідає не вся крива, а лише частина її (рис. 1.41).

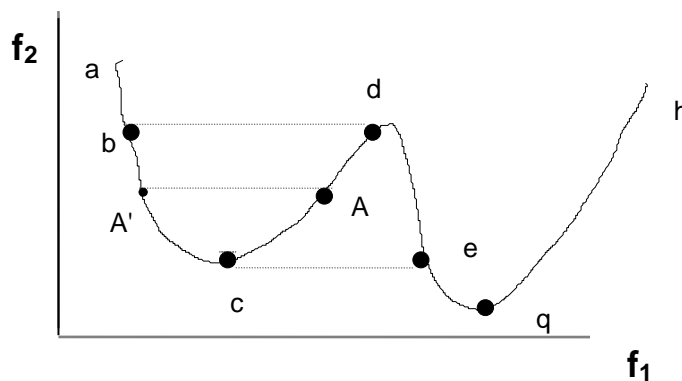


Рис. 1.41. Оптимізація за принципом Парето

Ділянка  $cd$  не може належати безлічі Парето, тому що для кожної точки цієї ділянки  $A$  знайдеться точка  $A'$  ділянки  $bc$ , у якій значення цільової функції буде менше, ніж у точці  $A$ . Зовсім аналогічно з розгляду повинні бути виключені ділянки  $de$  і  $gh$ . Тобто до безлічі Парето в даному випадку належать лише ділянки  $ac$  і  $eh$ , причому точка  $e$  також повинна бути виключена.

Таким чином, принцип Парето полягає в тому, що за оптимальне рішення  $x^*$  повинне бути обране таке, що належить безлічі Парето,  $x^* \in \Pi$ , тобто принцип Парето не виділяє єдиного рішення, воно лише звужує безліч можливих альтернатив, але часто таке звуження виявляється істотним, а остаточний вибір залишається за дослідником, який приймає відповідні рекомендації.

## 1.8. Динамічне програмування

### 1.8.1. Задачі динамічного програмування

Динамічне програмування (інакше, "динамічне планування") являє собою особливий математичний метод оптимізації рішень, спеціально пристосований до багатокрокових (поетапних) операцій.

Уявимо собі, що досліджувана операція  $Q$  являє собою процес, котрий розвивається в часі і розпадається на ряд "кроків" чи "етапів". Деякі операції розчленовуються на кроки природно: наприклад, при плануванні господарської діяльності групи підприємств природним кроком є господарський рік. В інших операціях поділ на кроки доводиться вводити штучно: наприклад, процес будови моста або іншого штучного спорудження на залізниці можна умовно розбити на етапи, кожний з яких займає якийсь часовий відрізок  $\Delta t$ .

Процес, про який йдеться, є керованим, тобто на кожному кроці приймається якийсь рішення, від якого залежить успіх даного кроку й операції в цілому. Управління операцією складається з ряду елементарних крокових керувань.

Розглянемо приклад природно-багатокрокової операції  $Q$ . Нехай планується діяльність групи (системи) промислових підприємств  $\Pi_1, \Pi_2, \dots, \Pi_k$  на деякий період часу  $T$ , що складається з  $m$  господарських років (рис. 1.42).

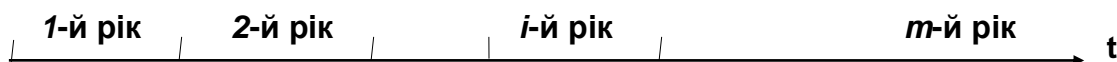


Рис. 1.42

На початку періоду на розвиток системи підприємств виділяються якісь основні засоби, що повинні бути якось розподілені між підприємствами. У процесі функціонування системи виділені засоби частково витрачаються (амортизуються). Крім того, кожне підприємство за рік приносить деякий прибуток, що залежить від вкладених засобів. На початку кожного господарського року наявні засоби можуть перерозподілятися між підприємствами: кожному з них виділяється якась частка засобів.

Ставиться питання: як потрібно на початку кожного року розподіляти наявні засоби між підприємствами, щоб сумарний прибуток від усієї системи підприємств за весь період  $T = m$  був максимальним?

Перед нами типова задача динамічного програмування.

Розглядається керований процес-функціонування системи підприємств. Управління процесом складається з розподілу (і перерозподілу) засобів. Кроком управління є виділення якихось засобів кожному з підприємств на початку господарського року.

Нехай на початку  $i$ -го року підприємствам  $\Pi_1, \Pi_2, \dots, \Pi_k$  виділяються відповідно засоби  $X_i^{(1)}, X_i^{(2)}, \dots, X_i^{(k)}$ .

Сутність цих значень являє собою не що інше, як управління на  $i$ -му кроці:  $U_i = (X_i^{(1)}, X_i^{(2)}, \dots, X_i^{(k)})$ .

Управління  $U$  операцією в цілому являє собою сукупність усіх крокових керувань:  $U = (U_1, U_2, \dots, U_m)$ .

Управління може бути добрим чи поганим, ефективним чи неефективним. Ефективність управління  $U$  оцінюється тим же показником  $W$ , що й ефективність операції в цілому. У нашому прикладі показник ефективності (цільова функція) являє собою сумарний прибуток від усієї системи підприємств за  $m$  років. Він залежить від управління операцією  $U$ , тобто від усієї сукупності крокових керувань:  $W = W(U) = W(U_1, U_2, \dots, U_m)$ .

Виникає питання: як вибрати крокові управління  $U_1, U_2, \dots, U_m$  для того, щоб величина  $W$  обернулася на максимум?

Поставлена задача називається задачею оптимізації управління, а управління, при якому показник  $W$  досягає максимуму, оптимальним управлінням. Будемо позначати оптимальне управління (на відміну від управління взагалі  $U$ ) буквою  $u$ . Оптимальне управління  $u$  багатокроковим процесом складається з сукупності оптимальних крокових керувань:  $u = (u_1, u_2, \dots, u_m)$ .

Таким чином, перед нами стоїть задача: визначити оптимальне управління на кожному кроці  $u_i = (1, 2, \dots, m)$  і, значить, оптимальне управління всією операцією  $u$ .

Зазначимо, що в нашому прикладі (управління фінансуванням системи підприємств) показник ефективності  $W$  являє собою суму прибутків за всі окремі роки (кроки):

$$W = \sum_{i=1}^m w_i ,$$

де  $w_i$  — прибуток від усієї системи підприємств за  $i$ -й рік.

Показник, що має таку властивість, називається адитивним. Поставимо задачу динамічного програмування в загальному вигляді.

Нехай маємо операцію  $Q$  з адитивним показником ефективності, що розпадається (навіть штучно) на  $m$  кроків. На кожному кроці застосовується якийсь управління  $U_i$ . Потрібно знайти оптимальне управління  $u = (u_1, u_2, \dots, u_m)$ , при якому показник ефективності  $W = \sum_{i=1}^m w_i$  обертається на максимум.

Поставлену задачу можна розв'язувати по-різному: чи шукати відразу оптимальне управління  $u$ , чи ж будувати його поступово крок за кроком, на кожному етапі розрахунку оптимізуючи тільки один крок. Звичайно другий спосіб виявляється простіше, ніж перший, особливо при великому числі кроків.

Така ідея поступової, покрокової оптимізації процесу і складає суть методу динамічного програмування.

З першого погляду ця ідея може здатися досить тривіальною. Справді, що може, здавалося, бути простіше: якщо важко оптимізувати операцію в цілому, то розбити її на ряд кроків; кожен такий крок буде окремою, маленькою операцією, оптимізувати яку вже неважко. Треба вибрати на кожному кроці таке управління, при якому ефективність цього кроку максимальна. Чи так це?

Виявляється зовсім не так! Принцип динамічного програмування аж ніяк не припускає, що кожен крок оптимізується окремо, незалежно від інших. Обирати крокове управління треба з урахуванням усіх його наслідків у майбутньому. Планування повинне бути далекоглядним, з урахуванням перспективи. Немає сенсу, якщо ми виберемо на даному кроці таке управління, при якому ефективність цього кроку максимальна, якщо надалі це перешкодить одержати гарні результати

інших кроків. Ні, вибираючи управління на кожному кроці, треба робити це неодмінно "з оглядкою на майбутнє", інакше можливі серйозні помилки.

Розглянемо приклад: нехай планується робота групи промислових підприємств, одні з яких зайняті випуском предметів споживання, інші ж роблять для цього машини. Задачею є одержання за  $m$  років максимального обсягу випуску предметів споживання.

Нехай плануються капіталовкладення на перший рік.

Виходячи з вузьких інтересів даного кроку (року), ми повинні були б усі засоби вкласти у виробництво предметів споживання, пустити наявні машини на повну міць і домогтися під кінець року максимального обсягу продукції. Але чи правильним буде таке рішення з огляду операції в цілому? Очевидно ні. Маючи на увазі майбутнє, необхідно виділити якусь частку засобів і на виробництво машин. При цьому обсяг продукції за перший рік, природно, знизиться, зате будуть створені умови, що дозволять збільшувати її виробництво в наступні роки.

Таким чином, плануючи багатокрокову операцію, необхідно вибирати управління на кожному кроці з обліком його майбутніх наслідків на ще майбутніх кроках.

Однак з цього правила є виключення. Серед усіх кроків існує один, котрий може плануватися попросту, без "оглядки на майбутнє". Який це крок? Очевидно, останній — після нього інших кроків немає. Цей крок, єдиний з усіх, можна планувати так, щоб він приніс найбільшу вигоду. Спланувавши оптимально цей останній крок, можна до нього пристроювати передостанній, до нього передпередостанній і т. д.

Тому процес динамічного програмування розгортається від кінця до початку: раніше за всі планується останній  $m$ -й крок. А як його спланувати, якщо ми не знаємо, чим скінчився передостанній? Очевидно потрібно зробити різні припущення про те, чим скінчився передостанній ( $m-1$ )-й крок, і для кожного з них знайти таке управління, при якому вигравш (прибуток) на останньому кроці був би максимальний. Розв'язавши цю задачу, ми знайдемо умовне

оптимальне управління на  $m$ -му кроці, тобто те управління, яке треба застосувати, якщо  $(m-1)$ -й крок закінчився певним чином.

Припустимо, що ця процедура виконана і для кожного результату  $(m-1)$ -го кроку ми знаємо умовне оптимальне управління на  $m$ -му кроці і відповідний йому умовний оптимальний виграш. Тепер ми можемо оптимізувати управління на передостанньому,  $(m-1)$ -му кроці. Зробимо всі можливі припущення про те, чим скінчився передпередостанній,  $(m-2)$ -й крок, і кожному з цих припущень знайдемо таке управління на  $(m-1)$ -му кроці, щоб виграш за останні два кроки (з яких останній вже оптимізований) був максимальним. Далі оптимізується управління на  $(m-2)$ -му кроці і т. д.

Одним словом, на кожному кроці шукається таке управління, що забезпечує оптимальне управління процесу щодо досягнутого в даний момент стану. Цей принцип вибору управління називається принципом оптимальності. Саме управління, що забезпечує оптимальне продовження процесу щодо заданого стану, називається умовним оптимальним управлінням на даному кроці.

Тепер припустимо, що умовне оптимальне управління на кожному кроці відомо: ми повинні знати, що робити далі, у якому б стані не був процес до початку кожного кроку. Тоді ми можемо знайти вже не "умовне", а просто оптимальне управління на кожному кроці.

Дійсно, нехай нам відомий початковий стан процесу, позначимо його  $S$ . Тепер ми уже знаємо, що робити на першому кроці: треба застосувати умовне оптимальне управління, вироблене нами для першого кроку, що належить до стану  $S$ . У результаті цього управління після першого кроку система перейде в інший стан  $S_1$ ; але для цього стану ми знову знаємо умовне оптимальне управління на другому кроці  $u$  і т. д. У такий спосіб ми знайдемо оптимальне управління процесом  $u = (u_1, u_2, \dots, u_m)$ , що призводить до максимально можливого виграшу  $W_{max}$ . Таким чином, у процесі оптимізації управління методом динамічного управління багатокроковий процес "проходиться" двічі:

- перший раз — від кінця до початку, у результаті чого знаходяться умовні оптимальні управління на кожному кроці й оптимальний виграш (теж умовний) на всіх кроках, починаючи з даного і до кінця процесу;



• другий раз — від початку до кінця, у результаті чого знаходяться (уже не умовні) оптимальні крокові управління на всіх кроках операції.

### **1.8.2. Принцип Беллмана**

Варто помітити, що основну ідею методу динамічного програмування складає сформульований Робертом Беллманом принцип оптимальності, котрий полягає в тім, що яким би не був стан процесу, що оптимізується, в результаті якогось числа кроків, управління на найближчому кроці повинне бути обране так, щоб воно забезпечувало максимальний вигаш на всіх кроках, що залишилися, включаючи даний.

### **1.8.3. Пошук найкоротшого шляху методом динамічного програмування**

Маємо мережу шляхів. З кожного вузла мережі можливе переміщення як по діагоналі, так і по горизонталі та вертикалі. Числа в графах мережі показують витрати на шлях з даного вузла до наступного (це може бути відстань, час, витрати коштів, матеріалів і т. п. у залежності від мети дослідження системи).

Потрібно знайти такий шлях з початкової точки  $X_0$  до кінцевої точки  $X_k$ , при якому сума витрат буде мінімальною (рис. 1.43).

Використовуємо метод динамічного програмування. Для цього розіб'ємо мережу на 4 кроки (4 вертикальні ділянки). Знаходимо, починаючи з кінця, "умовний" оптимальний варіант при переході з вузлів однієї вертикалі в іншу. Причому щоразу записуємо в кружок вузла мінімальні витрати від даного вузла до точки кінця шляху  $X_k$ . Спочатку проставляємо суму витрат, що накопичується, у вузлах останньої вертикалі 4–4. Суми складуть відповідно 5 та 10 одиниць. Потім знаходимо мінімальні суми у вузлах 3-ої вертикалі, тобто шукаємо оптимальне управління на останньому кроці, виходячи з трьох варіантів (вузлів) кінцевих умов передостаннього (3-го) кроку. Відповідно оптимальні значення для цих вузлів будуть 3, 5, 10.

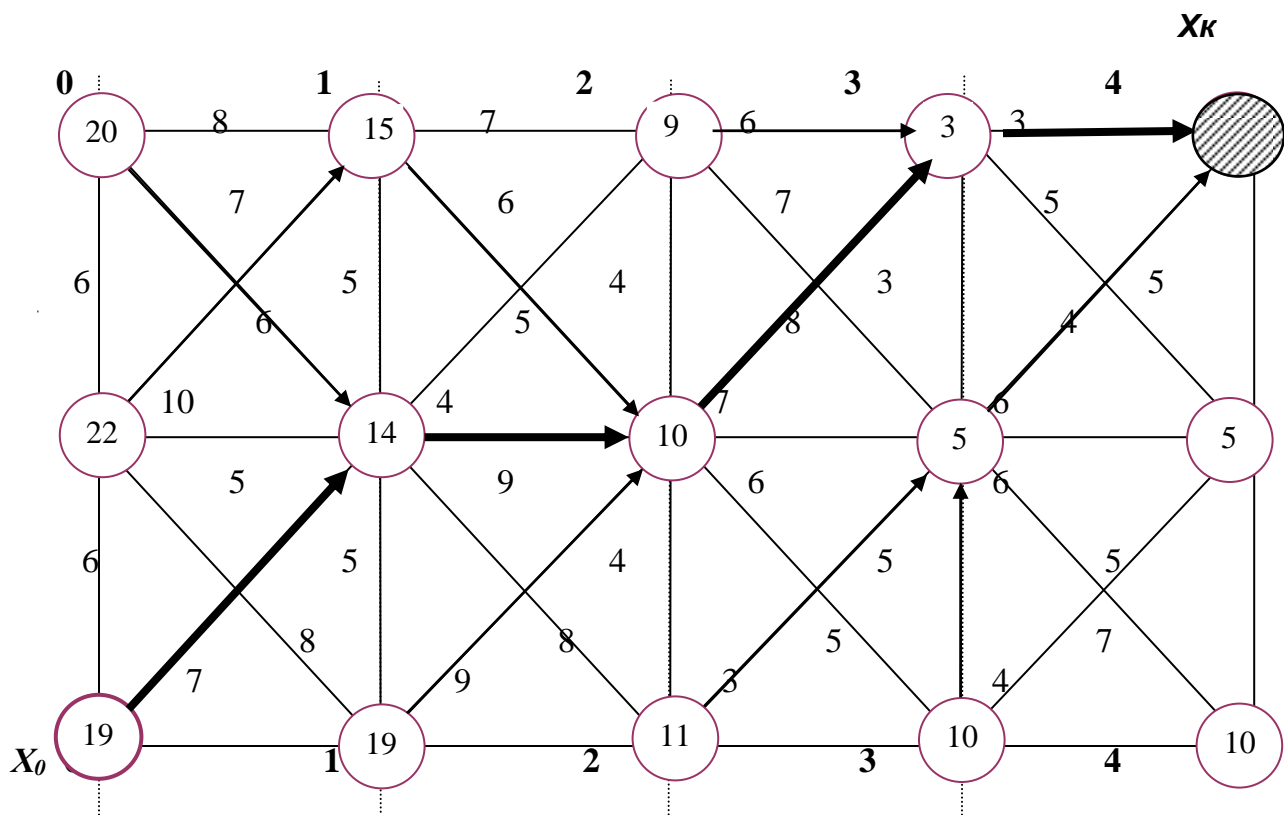


Рис. 1.43. Пошук найкоротшого шляху

Позначимо стрілками шляхи, за якими отримані ці умовні оптимальні рішення. Потім аналогічно відшукуються мінімальні витрати для 3, 2 і 1-го кроків. На цьому закінчується розрахунок "умовного" оптимального управління для вузла  $X_0$ . При цьому відразу одержують відповідь про оптимальне управління при проходженні з  $X_0$  до  $X_k$ . Мінімальні витрати складають 19 одиниць. Отже, шукана оптимальна траса прокладається по стрілках "умовного оптимального управління", починаючи з  $X_0$ .

### 1.9. Метод сіткового планування

Це логіко-математичний метод, в основу якого покладене графічне (топологічне) зображення виробничого плану у вигляді мережі з вузлами.

Метод дозволяє розв'язувати ряд задач планування. Одна з основних задач методу — це визначення критичного шляху, тобто

такої послідовності технологічно взаємозалежних робіт від початку до кінця плану, що має максимальну тривалість.

Найчастіше метод сіткового планування (СП) використовують для розв'язання задач раціонального планування складного комплексу взаємозалежних робіт (будівництво великих об'єктів і т. п.).

Сітковий план дозволяє визначити оптимальний розподіл матеріальних і трудових ресурсів, оптимальні моменти початку та закінчення робіт, виявити перешкоди до своєчасного завершення всього плану, виявити перелік робіт, що визначають терміни виконання плану.

Вхідними даними для сіткового плану служить сітковий план-список робіт плану, де зазначений перелік робіт, на які кожна робота спирається, тобто після завершення яких робіт може початися дана робота.

Наприклад. Скласти сітковий план відновлення, ремонту підвісного трубопроводу. Трубопровід оновлюється, замінюються клапани. Готових труб і клапанів немає. Монтажні креслення для виконання робіт підготовлені. Потрібна установка підмостків, тому що трубопровід підвісний.

При розв'язанні задачі фахівці склали перелік робіт, необхідних при ремонті (табл. 9.1).

Спочатку складається сіткова схема. Починають її складати, звичайно, з кінця, вроздріб, потім з'єднують (зшивають мережі).

При цьому об'єкти позначають символами у вигляді прямокутників, кружків або трикутників, а взаємозв'язки між ними — дугами зі стрілками, ці сполучні елементи являють собою граfi (рис. 1.44).

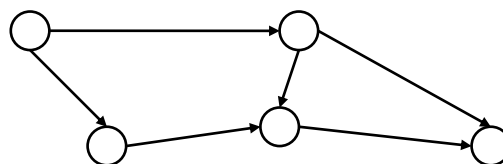


Рис. 1.44

Таблиця 9.1

## Перелік робіт, необхідних при ремонті трубопроводу

№ з/п	Умовне позначення робіт	Опис робіт	Тривалість робіт у днях	Індекси робіт
1	А	Складання спецнарядів на матеріали	10	0-1
2	Б	Вимикання старої лінії	1	0-2
3	В	Встановлення підмостків	6	1-2
4	Г	Розбирання підмостків	4	9-10
5	Д	Доставка труб	3	1-3
6	Е	Підготовка секцій труб	7	3-5
7	Ж	Установка труб	5	5-6
8	З	Зварювання труб	8	6-7
9	И	Пригін труб і клапанів	9	7-8
10	К	Доставка клапанів	2	1-4
11	Л	Встановлення клапанів	4	4-6
12	М	Зняття старих труб і клапанів	12	2-4
13	Н	Ізоляція труб	14	7-9
14	О	Випробування системи під тиском	2	8-10
15	П	Збирання і пуск системи	6	10-11

Точки графів (вершини) нумерують порядковими числами. Момент закінчення однієї роботи означає початок іншої, і завжди є початок і кінець графа.

Дві роботи називають рівнобіжними, якщо вони виходять з однієї вершини (рис. 1.45).

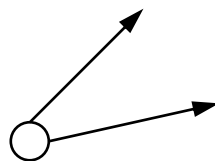


Рис. 1.45

Для відображення на графіку залежності, що безпосередньо не спирається одна на одну, вводиться так звана фіктивна робота (нульові витрати часу). Зображується фіктивна робота пунктирною лінією (рис. 1.46).

У сітковій схемі відображуються тільки взаємозв'язки без зазначення тривалості, витрат і засобів. Складає сіткову схему фахівець, який добре знає технологію виробництва. Початок графа ліворуч, кінець — праворуч. Стрілки йдуть зліва-направо. Не повинно бути замкнутих циклів.

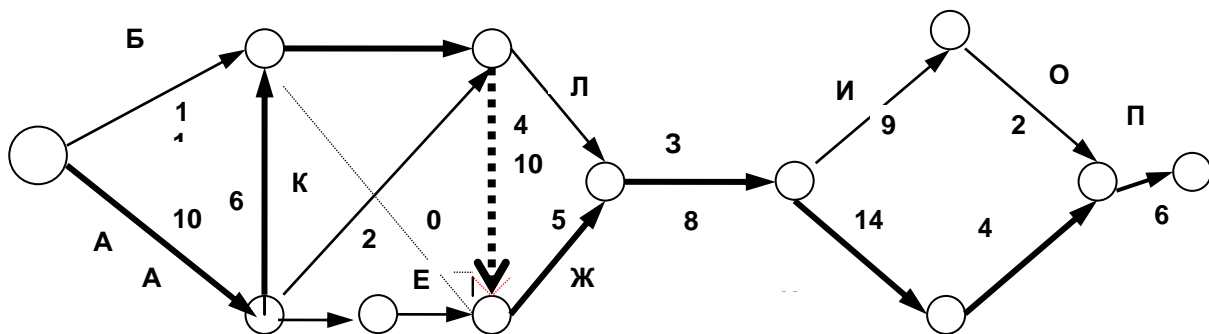


Рис. 1.46. Сітковий графік ремонту трубопроводу

Після побудови сіткової схеми проставляють тривалість робіт і отримують сітковий графік. Ланцюжків багато, у кожного своя тривалість. Найтриваліший шлях — критичний. Роботи на його шляху — критичні. Від їх довжини залежить загальна тривалість виконання робіт. Для робіт, що ми розглядаємо, критичний шлях становить 65 днів.

## 2. ЛАБОРАТОРНО–ПРАКТИЧНІ РОБОТИ

Дисципліна "Математичні моделі на ЕОМ" є продовженням та тісно пов'язана з дисципліною "Обчислювальна техніка, програмування, моделювання систем", що викладається студентам першого та другого років навчання механічного та будівельного факультетів Української державної академії залізничного транспорту.

Книга призначена для вивчення вище названого курсу й включає питання, що охоплюють 2-й семестр 3-го року підготовки.

*Методичні рекомендації з використання навчального посібника:*

- до виконання завдань варто приступати після вивчення теоретичної частини відповідного розділу;
- завдання рекомендується виконувати в тій послідовності, у якій вони наведені в посібнику;
- всі пункти пропонованих завдань необхідно виконувати та зберігати створені документи й програми;
- додаткові індивідуальні завдання призначені для поглибленого вивчення відповідних тем та для студентів заочної форми навчання.

*Вивчення теоретичного матеріалу*

Навчальний посібник містить весь матеріал, необхідний студенту для засвоєння даного розділу курсу. На початку навчального модуля студент отримує весь матеріал та самостійно вивчає теоретичні засади, розкриті в ньому (див. розділ 1). Передбачається, що студент, ознайомлюючись із навчальним матеріалом, самостійно готується до лекційних занять, вивчаючи навчально-методичну літературу та формулюючи список питань для обговорення з викладачем на лекції. У кінці збірника наводиться перелік навчально-методичної літератури, що буде корисною для підготовки до розв'язання задач.

Викладачі можуть проводити заняття з застосуванням комп'ютерних технічних засобів навчання, використовуючи теоретичний матеріал першого розділу навчального посібника.

### *Виконання лабораторних робіт*

У розділі 2 наведено перелік питань та завдання, що виконуються студентами при виконанні лабораторних робіт. Кожна робота присвячена певній темі.

Студент на основі вивченого теоретичного матеріалу виконує та оформлює звіт з лабораторної роботи відповідно до наведеного в додатку 2 зразка.

У процесі виконання лабораторної роботи студент здійснює такі види діяльності:

- відповідає на теоретичні (контрольні) питання;
- виконує практичне (обов'язкове) завдання відповідно до варіанта. Варіанти завдань визначає викладач.

Звіт надається викладачеві безпосередньо після виконання лабораторної роботи.

Після виконання лабораторної роботи студентові виставляється оцінка. Складові оцінки за виконання роботи розподілені таким чином: 0-60 балів — виконання індивідуального завдання, 0-30 балів — відповіді на теоретичні питання, 0-10 балів — варіативність, оптимізація розв'язання завдання, висновки з роботи.

Виконання кожної лабораторної роботи має свою питому вагу в загальній оцінці. Нижче наведені можливі вагові важелі для кожної лабораторної роботи: лабораторна робота № 1 — 0,2; лабораторна робота № 2 — 0,1; лабораторна робота № 3 — 0,1; лабораторна робота № 4 — 0,1; лабораторна робота № 5 — 0,1; лабораторна робота № 6 — 0,1; лабораторна робота № 7 — 0,1; лабораторна робота № 6 — 0,05; лабораторна робота № 7 — 0,05; лабораторна робота № 8 — 0,05; лабораторна робота № 9 — 0,05.

При недостатній сумарній кількості балів для одержання тієї підсумкової оцінки, що задовольняє студента, він має право набрати додаткові бали за рахунок виконання додаткових індивідуальних завдань.

### *Виконання індивідуальних завдань*

У додатку 2 навчального посібника розглянутий приклад моделювання роботи пункту перестановки вагонів та наведені варіанти індивідуальних завдань до розрахунково-графічної роботи.

### *Використання тестових завдань*

Тестові завдання призначені для контролю за навчальною діяльністю студента. Вони об'єднані в тематичні блоки. Наприкінці тематичного блоку студент проходить тестування. Для проходження тесту студент повинен розв'язати тестове завдання з використанням роздавального матеріалу з додатка 2. Оцінка з даного тематичного блоку виставляється викладачем за результатами виконання завдання.

Тестування з кожного тематичного блоку має свою питому вагу в загальній оцінці.

На початку тестування студент вказує свої прізвище, ініціали та номер групи або ставить особистий підпис. При захисті роботи викладач затверджує своїм підписом оцінку та дату захисту.

### *Загальна оцінка*

При викладанні дисципліни пропонується така система оцінювання знань студента. За модуль студент отримує загальну оцінку, що складається з середньозваженої оцінок за такі види діяльності, враховуючи вагові важелі:

<b>вид діяльності</b>	<b>вагові важелі</b>
виконання лабораторної роботи	0,75
тестування	0,25

Порядок проведення лабораторних робіт:

- 1** →  Вивчити теоретичний матеріал та відповісти на контрольні питання.
- 2**  Виконати індивідуальне завдання (згідно з варіантом).
- 3**  Результати роботи оформити у звіт (див. додаток 1).
- 4**  Захистити роботу у керівника заняття.



## РОБОТА № 1. ДОСЛІДЖЕННЯ НЕЛІНІЙНИХ МОДЕЛЕЙ

*Мета роботи* — придбання практичних навичок моделювання нелінійних систем, що описуються алгебраїчними та трансцендентними рівняннями.

→ **Вивчити** теоретичний матеріал та дати відповіді на контрольні питання.

### *Контрольні питання*

1. Які рівняння називаються трансцендентними?
2. З яких етапів складається процес знаходження коренів нелінійного рівняння?
3. Що означає ізолювати корінь рівняння?
4. Перечислити методи уточнення коренів рівняння.
5. Логічна умова наявності кореня на розглянутому інтервалі.
6. Коли завершується процес уточнення кореня методом ділення інтервалу навпіл?

▣ **Скласти, ввести та відлагодити програми, що реалізують:**

- ізоляцію коренів рівняння методом послідовного перебору (крок ізоляції вважати 0.1);
- уточнення коренів рівняння методом ділення інтервалу навпіл (точність уточнення вважати 0.0001).

▣ **Варіанти індивідуальних завдань**

Варіант	Рівняння для моделювання
1	2
1	$\cos(2x) - \sin(x) - \ln(x^2 + 0.1) = 0$
2	$\sin(2x) - \log(x) + 0.1 = 0$
3	$\ln(2x + 0.1) + 2\sqrt{2x} + 0.2 = 0$
4	$5\ln(x + 0.1) + 2\sin(2x) = 0$
5	$\cos(x^2) = \sqrt{2x}$
6	$3x - \sqrt{x^3} = \cos(x)$
7	$3\cos(2x) + \ln(x + 2) = 1.2$
8	$2x^2 - \cos(x) - 0.6 = 0$
9	$\ln(2x^2) - \sin(x) - \sqrt{2x} = 0$

1	2
10	$5x^2 + \sin(2x) - \sqrt{x} = 0$
11	$3 \sin 8x = 0.7x - 0.9$
12	$1.2 - \ln x = 4 \cos 2x$
13	$10 \cos x - 0.1 x^2 = 0$
14	$\sin x - 0.2 x = 0$
15	$2x - 2 \cos x = 0$

## РОБОТА № 2. ОБРОБКА РЕЗУЛЬТАТІВ ЕКСПЕРИМЕНТІВ

*Мета роботи* — придбання практичних навичок апроксимації результатів експериментів з використанням методу найменших квадратів.

→  Вивчити теоретичний матеріал та дати відповіді на контрольні питання.

*Контрольні питання*

1. У чому сутність апроксимації результатів експериментів?
2. Сутність методу найменших квадратів.
3. У чому сутність прямого ходу методу Гаусса?
4. У чому сутність оберненого ходу методу Гаусса?

**Скласти, ввести та відлагодити програми:**

- апроксимації результатів експериментів з використанням методу найменших квадратів;
- розв'язання системи лінійних алгебраїчних рівнянь методом Гаусса.

**Варіанти індивідуальних завдань**

1	N	1	2	3	4	5	6	7	8	9	10	11
	X	8,4	9,1	9,9	10	11	12	13	15	17		
	Y	3,8	4,8	5,3	5,8	6,3	6,3	6,7	6,6	5,6		
2	N	1	2	3	4	5	6	7	8	9	10	11
	X	9,4	9,9	10	11	12	13	14	14,5	15	15,5	17
	Y	6,3	5,8	4,8	4,1	3,3	3,3	3,3	2,8	2,6	4,3	6,3
3	N	1	2	3	4	5	6	7	8	9	10	11
	X	1	1,5	2	2,5	3,5	4	5	6	7,5	9	
	Y	2	1,5	1	7	5	8	1	1,3	2,4	3	

4	N	1	2	3	4	5	6	7	8	9	10	11
	X	1,5	2	3	3,5	4,5	5,5	6	6,5	7,5	8,3	9,5
	Y	1,5	2,7	3,5	4,5	5	5,5	5,8	5,8	5,5	5	4
5	N	1	2	3	4	5	6	7	8	9	10	11
	X	2	2,5	3	4	5	6	6,5	7	7,5	8	10
	Y	4	3,5	2,5	1,8	1	1	1	5	3	2	4
6	N	1	2	3	4	5	6	7	8	9	10	11
	X	1	1,5	2	2,5	3	3,5	4	5	9		
	Y	4	2,8	2	1,5	1,5	1,3	1,4	1	8		
7	N	1	2	3	4	5	6	7	8	9	10	11
	X	1	1,7	2,5	3	4	5	5,5	7,5	10		
	Y	1,5	2,5	3	3,5	4	4	4,4	4,3	3,3		
8	N	1	2	3	4	5	6	7	8	9	10	11
	X	3	3,5	4,5	5	6	7	7,5	8	9	9,8	11
	Y	6	7,2	8	9	9,5	10	10	10	10	9,5	8,5
9	N	1	2	3	4	5	6	7	8	9	10	11
	X	3,5	4	4,5	5,5	6,5	7,5	8,5	8,6	9	9,5	12
	Y	8,5	8	7	6,3	5,5	5,5	5,5	5	4,8	6,5	8,5
10	N	1	2	3	4	5	6	7	8	9	10	11
	X	2,5	3	3,5	4	5	5,5	6,5	7,5	9	11	
	Y	6,5	6	5,5	5,2	5	5,3	5,5	5,8	6,9	7,5	
11	N	1	2	3	4	5	6	7	8	9	10	11
	X	2,5	3,2	4	4,5	5,5	6,5	7	9	12		
	Y	6	7	7,5	8	8,5	8,5	8,9	8,8	7,8		
12	N	1	2	3	4	5	6	7	8	9	10	11
	X	8,9	9,4	10	11	12	13	13,5	14	15	16	17
	Y	3,8	5	5,8	6,8	7,3	7,8	8,1	7,8	7,8	7,3	6,3
13	N	1	2	3	4	5	6	7	8	9	10	11
	X	8,4	8,9	9,4	9,9	10	11	11,5	12	16		
	Y	6,3	5,1	4,3	3,8	3,9	3,6	3,7	3,3	3,1		
14	N	1	2	3	4	5	6	7	8	9	10	11
	X	8,4	8,9	9,4	9,9	11	11,5	12	13	15	16	
	Y	4,3	3,8	3,3	3	2,8	3,1	3,3	3,6	4,7	5,3	

☐ Оформити результати моделювання на ЕОМ у звіті до роботи в табличній та графічній формах.

## **РОБОТА № 3. МОДЕЛЮВАННЯ ПРОЦЕСІВ РОЗХОДУ ПАЛИВА**

*Мета роботи* — придбання практичних навичок моделювання процесів з використанням чисельних методів наближених обчислень визначених інтегралів на прикладі розрахунку розходу палива крізь трубу.

↪ **Вивчити** теоретичний матеріал та дати відповіді на контрольні питання.

*Контрольні питання*

1. Геометрична сутність визначеного інтеграла  $\int_a^b f(x)dx$ .
  2. Які чисельні методи використовуються для обчислення визначених інтегралів?
  3. Від чого залежить точність обчислення визначеного інтеграла в межах одного методу?
  4. У чому джерело похибок при обчисленні визначеного інтеграла для методу трапецій?
  5. Які границі інтегрування використовуються для визначення розходу палива крізь трубу?
  6. Які геометричні фігури використовуються для обчислення визначеного інтеграла методом Симпсона?
  7. Назвіть вимоги до числа підінтервалів розбиття для методу Симпсона.
  8. Який з відомих вам чисельних методів обчислення визначених інтегралів є найточнішим?
  9. Наведіть опис постановки задачі розходу палива крізь трубу.
- ▣** Описати постановку задачі моделювання розходу палива.  
**▣** Скласти, ввести та відлагодити програми моделювання по методах прямокутників, трапецій і Симпсона.  
**▣** Отримати аналітичні рішення задачі та оцінити помилки чисельного методу розв'язання задачі.

**Варіанти індивідуальних завдань**  
**(вхідні дані для моделювання розходу палива)**

<b>Варіант</b>	<b>a</b>	<b>b</b>	<b>c</b>	<b>r</b>	<b>N</b>
<b>1</b>	<b>0.5</b>	<b>1.2</b>	<b>2.2</b>	<b>0.25</b>	<b>150</b>
<b>2</b>	<b>0.6</b>	<b>2.4</b>	<b>2.4</b>	<b>0.45</b>	<b>120</b>
<b>3</b>	<b>0.7</b>	<b>3.1</b>	<b>1.8</b>	<b>0.55</b>	<b>130</b>
<b>4</b>	<b>0.8</b>	<b>2.3</b>	<b>2.6</b>	<b>0.6</b>	<b>140</b>
<b>5</b>	<b>0.9</b>	<b>3.2</b>	<b>2.2</b>	<b>1.2</b>	<b>170</b>
<b>6</b>	<b>1.0</b>	<b>1.8</b>	<b>2.5</b>	<b>0.8</b>	<b>110</b>
<b>7</b>	<b>1.2</b>	<b>1.9</b>	<b>3.1</b>	<b>0.9</b>	<b>124</b>
<b>8</b>	<b>1.5</b>	<b>2.2</b>	<b>3.7</b>	<b>1.1</b>	<b>144</b>
<b>9</b>	<b>1.9</b>	<b>2.5</b>	<b>2.8</b>	<b>0.35</b>	<b>150</b>
<b>10</b>	<b>3.2</b>	<b>2.8</b>	<b>2.7</b>	<b>0.30</b>	<b>160</b>
<b>11</b>	<b>3.6</b>	<b>2.9</b>	<b>3.2</b>	<b>0.40</b>	<b>170</b>
<b>12</b>	<b>3.9</b>	<b>3.1</b>	<b>2.6</b>	<b>0.50</b>	<b>180</b>
<b>13</b>	<b>4.2</b>	<b>2.4</b>	<b>1.9</b>	<b>0.65</b>	<b>190</b>
<b>14</b>	<b>4.4</b>	<b>1.7</b>	<b>2.0</b>	<b>0.75</b>	<b>200</b>
<b>15</b>	<b>4.7</b>	<b>2.7</b>	<b>3.1</b>	<b>0.85</b>	<b>210</b>

***РОБОТА № 4. МОДЕЛЮВАННЯ ДИНАМІЧНИХ  
ДЕТЕРМІНОВАНИХ СИСТЕМ***

*Мета роботи* — придбання практичних навичок моделювання динамічних детермінованих систем, опис системами звичайних диференціальних рівнянь (ЗДР).

▣ **Ознайомитися з постановкою задачі моделювання нестационарного розподілу температури в однорідних будівельних конструкціях.**

*Контрольні питання*

1. Назвіть властивості динамічної детермінованої системи.
2. Що таке результат чисельного розв'язання диференціального рівняння?
3. У чому джерело похибок при інтегруванні ЗДР методом Ейлера?
4. Які похибки методів Ейлера та Рунге-Кута?

▣ Скласти, ввести та відлагодити програми моделювання за методами Ейлера і Рунге-Кутта.

▣ Варіанти індивідуальних завдань

Варіант	F1(x, y, z)	F2(x, y, z)	X0	Y0	Z0	h	N
1	$2x - 3y + 2z$	$3x - 2y + z$	0	1	0	0.1	10
2	$x - 2y + z$	$2x + y - 3z$	1	1	0	$\frac{0.1}{5}$	10
3	$2x - y + 3z$	$4x - 2y + 3z$	0	1	1	0.1	10
4	$3x + y - 4z$	$2x + 2y - 2z$	1	0	1	$\frac{0.1}{5}$	10
5	$2x - 2y + 3z$	$5x - 3y + 3z$	0	0	0	0.2	10
6	$3x - 2y + 4z$	$2x + 3y - 3z$	1	1	1	0.1	10
7	$2x + 2y - 3z$	$2x - 2y + 3z$	0	1	0	$\frac{0.1}{5}$	10
8	$3x - 2y + z$	$x + 3y - 3z$	1	0	1	0.2	10
9	$2x + 3y - 5z$	$x - 5y + 2z$	0	1	1	$\frac{0.1}{5}$	10
10	$2x + 2y - 4z$	$3x - 3y - z$	1	1	0	0.1	10
11	$3x + y + 3z$	$4x - 3y + 2z$	0	0	1	0.2	10
12	$3x - y + z$	$2x + 5y - 3z$	1	0	1	$\frac{0.1}{5}$	10
13	$2x + 3 + z$	$0.5x - 3y + 2z$	0	1	1	0.1	10
14	$3x - y - 2z$	$2x + 3y - 0.5z$	1	0	1	0.2	10
15	$x + 2y - 3z$	$1.5x - 3y - 2z$	0	1	0	0.1	10

▣ Оформити результати моделювання на ЕОМ у звіті до роботи в табличній і графічній формах.

## **РОБОТА № 5. МЕТОД ВИПАДКОВОГО ПОШУКУ В ЗАДАЧАХ ОДНОВИМІРНОЇ ОПТИМІЗАЦІЇ**

*Мета роботи* — придбання практичних навичок розв'язання задач одновимірної оптимізації методом випадкового пошуку.

☞ **Вивчити** теоретичний матеріал та дати відповіді на контрольні питання.

*Контрольні питання*

1. Що є результатом розв'язання оптимізаційної задачі?
2. Які числа обирає генератор випадкових чисел RND?
3. Який вигляд має формула для генерування випадкових дійсних чисел  $x$  з діапазону  $[a; b]$ ?
4. Яка цільова функція є унімодальною?
5. У чому перевага методу випадкового пошуку?
6. У чому недолік методу випадкового пошуку?
7. Від чого залежить точність обчислення оптимального значення цільової функції для методу випадкового пошуку?
8. Що означає табулювання функції в заданому діапазоні?
9. Що таке проектні параметри в задачах оптимізації?
10. Що таке цільова функція?

☞ **Скласти, ввести та відлагодити** програму дослідження заданої цільової функції.

☞ **Варіанти індивідуальних завдань**

Варіант	Цільова функція $F(x)$	XN	XK	N
1	2	3	4	5
1	$(15-2*x*(3-x))*EXP(-X/3)$	0	10	100
2	$(10-2*x*(8-x))*EXP(-X/3)$	0	10	200
3	$(40-2*x*(8-x))*EXP(-X/5)$	1	11	120
4	$(25-2*x*(8-x))*EXP(-X/4)$	2	12	140
5	$(22-2*x*(7-x))*EXP(-X/2)$	1	10	150
6	$(18-2*x*(1-x))*EXP(-X/3)$	1	9	160
7	$(9-2*x*(2-x))*EXP(-X/3)$	2	11	170
8	$(4-2*x*(6-x))*EXP(-X/7)$	2	10	180
9	$(34-2*x*(9-x))*EXP(-X/5)$	2	12	190
10	$(30-2*x*(8-x))*EXP(-X/3)$	3	9	210

1	2	3	4	5
11	$(25-2*x*(5-x))*EXP(-X/4)$	4	12	220
12	$(15-2*x*(15-x))*EXP(-X/4)$	3	15	230
13	$(20-2*x*(10-x))*EXP(-X/2)$	4	9	240
14	$(24-2*x*(9-x))*EXP(-X/3)$	5	10	180
15	$(15-2*x*(3-x))*EXP(-X/3)$	5	10	190

## **РОБОТА № 6. МЕТОД ЗОЛОТОГО ПЕРЕТИНУ В ЗАДАЧАХ ОДНОВИМІРНОЇ ОПТИМІЗАЦІЇ**

*Мета роботи* — придбання практичних навичок розв'язання задач одновимірної оптимізації методом золотого перетину.

↪ **Вивчити** теоретичний матеріал та дати відповіді на контрольні питання.

*Контрольні питання*

1. Назвіть властивості золотого перетину для відрізка довжиною L.
2. Який алгоритм визначення чисел Фібоначчі? Запишіть перші десять чисел.
3. У чому перевага методу золотого перетину?
4. У чому недолік методу золотого перетину?
5. Якими операторами визначається звертання до підпрограми методу золотого перетину та вихід з неї?
6. Які операції в задачах оптимізації необхідно виконати для переходу від пошуку min цільової функції до max та навпаки?
7. Що таке локальний та глобальний екстремуми функції, що оптимізується?
8. Як здійснюється вибір пошуку min або max цільової функції в основній програмі оптимізації?
9. Як пов'язане між собою число ітерацій та точність обчислення значень цільової функції?

▣ **Скласти, ввести та відлагодити** програму реалізації методу золотого перетину у відповідності з наведеним алгоритмом для дослідження заданої цільової функції, оформивши її як процедуру-підпрограму.



▣ Змінюючи точність обчислень значень цільової функції, дослідити залежність числа ітерацій від точності.

### Варіанти індивідуальних завдань

Варіант	Цільова функція F(x)	A	B
1	$Y=(15-2*X*(3-X))*EXP(-X/3)$	0	10
2	$Y=(10-2*X*(8-X))*EXP(-X/3)$	0	10
3	$Y=(40-2*X*(8-X))*EXP(-X/5)$	0	10
4	$Y=(25-2*X*(8-X))*EXP(-X/4)$	0	15
5	$Y=(22-2*X*(7-X))*EXP(-X/2)$	0	15
6	$Y=(15-2*X*(3-X))*EXP(-X/3)$	0	15
7	$Y=(9-2*X*(1-X))*EXP(-X/3)$	2	12
8	$Y=(4-2*X*(2-X))*EXP(-X/7)$	2	12
9	$Y=(34-2*X*(6-X))*EXP(-X/5)$	2	12
10	$Y=(30-2*X*(9-X))*EXP(-X/3)$	0	20
11	$Y=(25-2*X*(8-X))*EXP(-X/4)$	0	20
12	$Y=(5-2*X*(5-X))*EXP(-X/4)$	0	20
13	$Y=(15-2*X*(15-X))*EXP(-X/4)$	5	10
14	$Y=25-2*X*(10-X)*(EXP(-X/2))$	5	10
15	$Y=(24-2*X*(9-X))*EXP(-X/3)$	5	10

▣ Побудувати графічну залежність числа ітерацій від точності обчислень.

## **РОБОТА № 7. МЕТОД ПОКООРДИНАТНОГО ПІДЙОМУ В ЗАДАЧАХ БАГАТОВИМІРНОЇ ОПТИМІЗАЦІЇ**

*Мета роботи* — придбання практичних навичок розв’язання задач двовимірної оптимізації методом покоординатного підйому (спуску).

### *Контрольні питання*

1. Що є результатом розв’язання багатопараметричної оптимізаційної задачі?
2. Які ви знаєте методи багатовимірної оптимізації?

3. Які обмеження накладаються на цільову функцію в методі покоординатного підйому (спуску)?
4. Від чого залежить точність обчислення оптимального значення цільової функції для методу покоординатного підйому (спуску)?
5. Як табулювати функцію 2-х змінних у заданому діапазоні?
6. Які алгоритми одновимірного пошуку можна використовувати для методу покоординатного підйому (спуску)?

▣ Скласти, ввести та відлагодити програму знаходження мінімуму заданої цільової функції. Крок змінення параметрів обчислюється як довжина інтервалу дослідження, поділена на 10. Точність  $\xi = 0.00001$ .

### Варіанти індивідуальних завдань

Варіант	Цільова функція F(x)	XN	XK	YN	YK
1	2	3	4	5	6
1	$(15-2*x*(3-Y))*EXP(-X/3)$	0	17	2	10
2	$(10-2*x*(8-Y))*EXP(-X/3)$	3	10	4	10
3	$(40-2*Y*(8-x))*EXP(-X/5)$	1	13	1	11
4	$(25-2*x*(8-Y))*EXP(-X/4)$	5	12	2	12
5	$(22-2*Y*(7-x))*EXP(-X/2)$	1	10	3	10
6	$(18-2*x*(1-Y))*EXP(-X/3)$	1	9	5	9
7	$(9-2*x*(2-Yx))*EXP(-X/3)$	7	11	6	15
8	$(4-2*Y*(6-x))*EXP(-X/7)$	2	16	7	20
9	$(34-2*x*(9-Y))*EXP(-X/5)$	4	12	1	12
10	$(30-2*x*(8-x))*EXP(-X/3)$	2	19	2	9
11	$(25-2*Y*(5-x))*EXP(-X/4)$	4	12	4	12
12	$(15-2*Y*(15-x))*EXP(-X/4)$	3	15	1	17
13	$(20-2*x*(10-Y))*EXP(-X/2)$	4	9	1	9
14	$(24-2*Y*(9-Y))*EXP(-X/3)$	5	10	4	10
15	$(15-2*x*(3-x))*EXP(-X/3)$	5	10	5	23

▣ Оформити результати моделювання на ЕОМ у звіті до роботи в табличній та графічній формах.

## **РОБОТА № 8. СИМПЛЕКС–МЕТОД ЛІНІЙНОГО ПРОГРАМУВАННЯ В ЗАДАЧАХ ОПТИМІЗАЦІЇ**

*Мета роботи* — придбання практичних навичок розв’язання задач оптимізації симплекс-методом.

→ **Вивчити** теоретичний матеріал та дати відповіді на контрольні питання.

### *Контрольні питання*

1. Формулювання задачі лінійного програмування.
2. Що таке цільова функція?
3. Який план називається оптимальним?
4. Стандартна та канонічна форми задачі лінійного програмування.
5. Основні методи розв’язання задач лінійного програмування.

☐ **Згідно з варіантом завдання записати цільову функцію та систему обмежень для розв’язання задачі використання ресурсів.**

☐ **Скласти, ввести та відлагодити програму розв’язання задачі оптимізації симплекс-методом.**

### **Варіанти індивідуальних завдань<sup>2</sup>**

Варіант	Наявність техніки				Комплекс 1-го типу				Комплекс 2-го типу			
	С	Е	Б	А	С	Е	Б	продуктивність	С	Е	А	продуктивність
<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>10</b>	<b>11</b>	<b>12</b>	<b>13</b>
<b>1</b>	<b>10</b>	<b>6</b>	<b>10</b>	<b>12</b>	<b>2</b>	<b>3</b>	<b>3</b>	<b>10</b>	<b>22</b>	<b>1</b>	<b>3</b>	<b>12</b>
<b>2</b>	<b>12</b>	<b>8</b>	<b>8</b>	<b>6</b>	<b>3</b>	<b>1</b>	<b>3</b>	<b>20</b>	<b>3</b>	<b>2</b>	<b>3</b>	<b>18</b>
<b>3</b>	<b>16</b>	<b>6</b>	<b>16</b>	<b>6</b>	<b>4</b>	<b>1</b>	<b>2</b>	<b>15</b>	<b>2</b>	<b>1</b>	<b>2</b>	<b>18</b>
<b>4</b>	<b>12</b>	<b>12</b>	<b>12</b>	<b>12</b>	<b>5</b>	<b>3</b>	<b>2</b>	<b>12</b>	<b>4</b>	<b>2</b>	<b>3</b>	<b>14</b>
<b>5</b>	<b>10</b>	<b>4</b>	<b>8</b>	<b>10</b>	<b>4</b>	<b>2</b>	<b>4</b>	<b>22</b>	<b>3</b>	<b>2</b>	<b>2</b>	<b>18</b>
<b>6</b>	<b>16</b>	<b>8</b>	<b>10</b>	<b>8</b>	<b>3</b>	<b>2</b>	<b>1</b>	<b>16</b>	<b>2</b>	<b>1</b>	<b>3</b>	<b>15</b>

<sup>2</sup> Примітка. С — самоскид, Б — бульдозер, Е — екскаватор, А — автогрейдер.

1	2	3	4	5	6	7	8	9	10	11	12	13
7	10	6	10	4	4	2	2	18	4	2	1	20
8	12	10	10	12	4	3	2	12	3	2	2	14
9	8	8	8	8	2	2	2	14	3	1	2	16
10	16	10	8	4	4	2	2	24	2	2	2	20
11	20	12	12	6	5	2	3	16	4	2	2	15
12	14	14	10	12	3	2	2	18	4	1	1	20
13	18	6	6	6	4	2	2	30	4	2	4	32
14	12	14	20	14	2	2	4	36	4	2	1	40
15	8	6	10	4	4	2	4	42	4	2	1	38

☐ Оформити результати моделювання на ЕОМ у звіті до роботи в табличній та графічній формах.

### ***РОБОТА № 9. МЕТОД ДИНАМІЧНОГО ПРОГРАМУВАННЯ***

*Мета роботи — придбання практичних навичок розв’язання оптимізаційних задач методом динамічного програмування на прикладі пошуку найкоротшого шляху.*

☐ Вивчити теоретичний матеріал та дати відповіді на контрольні питання.

*Контрольні питання*

1. Які задачі належать до задач динамічного програмування?
2. У чому сутність принципу, сформульованого Беллманом?
3. Яке співвідношення називається рекурентним? Навести приклад.
4. Як позначається відсутність зв’язку між пунктами в алгоритмі та програмі?

☐ Скласти транспортну сітку у відповідності з варіантом індивідуального завдання.

☐ Скласти, ввести та відлагодити програму реалізації методу динамічного програмування для визначення найкоротшого шляху з пункту «1» до пункту «10».

**Варіанти індивідуальних завдань (транспортна сітка задана відстанями між вершинами)**

<b>1</b>	<b>a<sub>1,2</sub>=6, a<sub>1,3</sub>=7, a<sub>1,4</sub>=12, a<sub>2,4</sub>=6, a<sub>3,5</sub>=8, a<sub>4,6</sub>=14, a<sub>5,6</sub>=10, a<sub>5,7</sub>=9, a<sub>6,8</sub>=4, a<sub>6,9</sub>=5, a<sub>7,10</sub>=10, a<sub>8,9</sub>=5, a<sub>9,10</sub>=6.</b>
<b>2</b>	<b>a<sub>1,2</sub>=5, a<sub>1,3</sub>=9, a<sub>1,5</sub>=6, a<sub>2,4</sub>=5, a<sub>3,6</sub>=6, a<sub>4,7</sub>=6, a<sub>4,10</sub>=9, a<sub>5,7</sub>=5, a<sub>5,8</sub>=3, a<sub>6,8</sub>=8, a<sub>7,10</sub>=7, a<sub>8,9</sub>=4, a<sub>9,10</sub>=7.</b>
<b>3</b>	<b>a<sub>1,2</sub>=5, a<sub>1,3</sub>=6, a<sub>2,4</sub>=4, a<sub>3,6</sub>=4, a<sub>4,5</sub>=4, a<sub>4,6</sub>=7, a<sub>4,7</sub>=8, a<sub>4,9</sub>=10, a<sub>5,7</sub>=5, a<sub>6,8</sub>=2, a<sub>7,10</sub>=10, a<sub>8,9</sub>=10, a<sub>9,10</sub>=5.</b>
<b>4</b>	<b>a<sub>1,2</sub>=6, a<sub>1,3</sub>=10, a<sub>2,4</sub>=7, a<sub>3,5</sub>=5, a<sub>3,7</sub>=10, a<sub>4,6</sub>=7, a<sub>4,7</sub>=6, a<sub>5,7</sub>=6, a<sub>5,8</sub>=6, a<sub>6,9</sub>=5, a<sub>7,10</sub>=5, a<sub>8,10</sub>=6, a<sub>9,10</sub>=5.</b>
<b>5</b>	<b>a<sub>1,2</sub>=6, a<sub>1,3</sub>=5, a<sub>2,4</sub>=7, a<sub>3,5</sub>=6, a<sub>3,6</sub>=6, a<sub>4,7</sub>=8, a<sub>5,7</sub>=10, a<sub>5,8</sub>=9, a<sub>6,8</sub>=8, a<sub>7,9</sub>=6, a<sub>7,10</sub>=10, a<sub>8,10</sub>=10, a<sub>9,10</sub>=6.</b>
<b>6</b>	<b>a<sub>1,2</sub>=9, a<sub>1,3</sub>=5, a<sub>1,4</sub>=18, a<sub>2,4</sub>=8, a<sub>3,5</sub>=10, a<sub>4,6</sub>=7, a<sub>5,6</sub>=9, a<sub>5,7</sub>=10, a<sub>6,8</sub>=3, a<sub>6,9</sub>=6, a<sub>7,10</sub>=10, a<sub>8,9</sub>=4, a<sub>9,10</sub>=6.</b>
<b>7</b>	<b>a<sub>1,2</sub>=4, a<sub>1,3</sub>=4, a<sub>1,5</sub>=7, a<sub>2,4</sub>=5, a<sub>3,6</sub>=6, a<sub>4,7</sub>=9, a<sub>4,10</sub>=11, a<sub>5,7</sub>=6, a<sub>5,8</sub>=4, a<sub>6,8</sub>=8, a<sub>7,10</sub>=7, a<sub>8,9</sub>=4, a<sub>9,10</sub>=6.</b>
<b>8</b>	<b>a<sub>1,2</sub>=3, a<sub>1,3</sub>=8, a<sub>2,4</sub>=6, a<sub>3,6</sub>=5, a<sub>4,5</sub>=9, a<sub>4,6</sub>=8, a<sub>4,7</sub>=9, a<sub>4,9</sub>=12, a<sub>5,7</sub>=10, a<sub>6,8</sub>=8, a<sub>7,10</sub>=4, a<sub>8,9</sub>=9, a<sub>9,10</sub>=6.</b>
<b>9</b>	<b>a<sub>1,2</sub>=6, a<sub>1,3</sub>=5, a<sub>2,4</sub>=7, a<sub>3,5</sub>=6, a<sub>3,7</sub>=10, a<sub>4,6</sub>=4, a<sub>4,7</sub>=3, a<sub>5,7</sub>=5, a<sub>5,8</sub>=7, a<sub>6,9</sub>=5, a<sub>7,10</sub>=10, a<sub>8,10</sub>=8, a<sub>9,10</sub>=5.</b>
<b>10</b>	<b>a<sub>1,2</sub>=6, a<sub>1,3</sub>=5, a<sub>2,4</sub>=7, a<sub>3,5</sub>=8, a<sub>3,6</sub>=7, a<sub>4,7</sub>=6, a<sub>5,7</sub>=8, a<sub>5,8</sub>=9, a<sub>6,8</sub>=8, a<sub>7,9</sub>=4, a<sub>7,10</sub>=8, a<sub>8,10</sub>=8, a<sub>9,10</sub>=8.</b>
<b>11</b>	<b>a<sub>1,2</sub>=7, a<sub>1,3</sub>=4, a<sub>1,4</sub>=10, a<sub>2,4</sub>=6, a<sub>3,5</sub>=7, a<sub>4,6</sub>=12, a<sub>5,6</sub>=10, a<sub>5,7</sub>=9, a<sub>6,8</sub>=3, a<sub>6,9</sub>=6, a<sub>7,10</sub>=12, a<sub>8,9</sub>=7, a<sub>9,10</sub>=6.</b>

12	$a_{1,2}=8,$ $a_{1,3}=7,$ $a_{1,5}=6,$ $a_{2,4}=6,$ $a_{3,6}=6,$ $a_{4,7}=5,$ $a_{4,10}=9,$ $a_{5,7}=8,$ $a_{5,8}=7,$ $a_{6,8}=8,$ $a_{7,10}=7,$ $a_{8,9}=4,$ $a_{9,10}=7.$
13	$a_{1,2}=7,$ $a_{1,3}=8,$ $a_{2,4}=6,$ $a_{3,6}=8,$ $a_{4,5}=9,$ $a_{4,6}=7,$ $a_{4,7}=8,$ $a_{4,9}=10,$ $a_{5,7}=6,$ $a_{6,8}=2,$ $a_{7,10}=10,$ $a_{8,9}=10,$ $a_{9,10}=8.$
14	$a_{1,2}=6,$ $a_{1,3}=10,$ $a_{2,4}=7,$ $a_{3,5}=5,$ $a_{3,7}=10,$ $a_{4,6}=5$ , $a_{4,7}=6,$ $a_{5,7}=6,$ $a_{5,8}=9,$ $a_{6,9}=7,$ $a_{7,10}=5,$ $a_{8,10}=6,$ $a_{9,10}=6.$
15	$a_{1,2}=6,$ $a_{1,3}=8,$ $a_{2,4}=7,$ $a_{3,5}=12,$ $a_{3,6}=6,$ $a_{4,7}=8,$ $a_{5,7}=10,$ $a_{5,8}=9,$ $a_{6,8}=8,$ $a_{7,9}=9,$ $a_{7,10}=10,$ $a_{8,10}=13,$ $a_{9,10}=9.$

**▣ Отримати результат розв'язання задачі на ПК и порівняти його з результатом ручного розв'язання.**

## *Список літератури*

1. Turbo Pascal: Алгоритми і програми: Чисельні методи в фізиці та математиці: Навч. посібник /А.Б.Бартків та ін. — К.: Вища школа, 1992.

2. Бережная Е.В., Бережной В.И. Математические методы моделирования экономических систем. — М.: Финансы и статистика, 2003. — 368 с.

3. Боглаев Ю.П. Вычислительная математика и программирование. — М.: Высш. школа, 1990.

4. Вентцель Е.С. Исследование операций. — М.: Наука, 1972.

5. Глухов В.В., Медников М.Д., Коробко С.Б. Математика в бизнесе: Учеб. для вузов. Специальная литература. — М.: Лань, 2005. — 528 с.

6. Джашитов В.Э., Панкратов В.М. Математические модели теплового дрейфа гироскопических датчиков инерциальных систем / Под общей ред. академика РАН В.Г.Пешехонова. — СПб.: ГНЦ РФ - ЦНИИ "Электроприбор", 2001. - 150 с.

7. Катулев А.Н., Северцев Н.А. Математические методы в системах поддержки принятия решений. — М.: Высшая школа, 2005. — 311 с.

8. Комп'ютерна техніка і програмування. Методичні вказівки до розрахунково-графічної роботи з дисципліни «Комп'ютерна техніка і програмування» / С.Є.Бантюков, О.Є.Пенкіна, С.О.Бантюкова. — Харків: УкрДАЗТ, 2006. — 50 с.

9. Красс М.С. Математика в экономике. Математические модели и методы. — М.: Финансы и статистика, 2007. — 544 с.

10. Маликов В.Т., Кветный Р.Н. Вычислительные методы и применение ЭВМ. — К.: Вища школа, 1989.

11. Методические указания к лаб. работам № 1-4 по дисциплине "Математические методы и модели в расчетах на ЭВМ" (исследование детерминированных моделей). — Харьков: ХарГАЖТ, 1995.

12. Методы и модели оптимизации: Методические указания к лабораторным работам. — Харьков: ХИИТ, 1993.

13. Методы классической и современной теории автоматического управления. Т.1. Математические модели, динамические характеристики и анализ систем автоматического управления: Учебник / Под ред. К.А.Пупкова. — М.: Высш. школа, 2004. — 322 с.

14. Невежин В. Сборник задач по курсу "Экономико-математическое моделирование". — М.: Городец, 2005. — 320 с.

15. Петров А.В. и др. Вычислительная техника в инженерных и экономических расчетах. — М.: Высш.школа, 1984.

16. Советов Б.Я., Яковлев С.А. Моделирование систем. — М.: Высш. школа, 1985.

17. Філіппенко І.Г., Гончаров В.О. Меркулов В.С. Основи проектування технічних систем. Класифікація моделей. Основи теорії оптимізації: Конспект лекцій. — Ч. 1. — Харків: ХарДАЗТ, 2001.

18. Шелобаев С.И. Математические методы и модели. — М.: ЮНИТИ, 2001. — 367 с.

19. Шенон Р. Имитационное моделирование систем — искусство и наука. — М.: 1978.

20. Щуп Т. Решение инженерных задач с использованием ЭВМ: Практическое руководство. — М.: Мир, 1982.



## ЗРАЗОК ОФОРМЛЕННЯ ЗВІТУ З РОБОТИ

## ЛАБОРАТОРНА РОБОТА № 1 ДОСЛІДЖЕННЯ НЕЛІНІЙНИХ МОДЕЛЕЙ

**1. Мета роботи** — придбання практичних навичок моделювання нелінійних систем, що описуються алгебраїчними й трансцендентними рівняннями.

2. Завдання й порядок виконання

2.1. Вивчити навчальний матеріал до лабораторної роботи і записати відповіді на контрольні питання.

2.2. Скласти Basic-програми для ізоляції коренів рівняння методом послідовного перебору і їхнього уточнення методом ділення інтервалу навпіл відповідно до варіанта завдання.

2.3. Ввести й відлагодити програми:

- ізоляції коренів рівняння методом послідовного перебору (крок ізоляції прийняти рівним 0.1);
- уточнення кореня рівняння методом ділення інтервалу навпіл.

2.4. Оформити результати моделювання на ЕОМ у звіті до лабораторної роботи.

2.5. Захистити лабораторну роботу в керівника занять.

3. Контрольні питання

3.1. Які рівняння називаються трансцендентними?

*Нелінійні рівняння з одним невідомим поділяються на алгебраїчні і трансцендентні. Рівняння вигляду  $f(x)=0$  називається алгебраїчним, якщо функція є алгебраїчною функцією.*

*Наприклад,  $3X^4 + 2X^3 + 2X^2 - 4X - 5.2 = 0$ .*

*Якщо функція  $f(x)$  не є алгебраїчною і містить у лівій частині такі функції, як EXP, LOG, SIN і т. д., то рівняння називається трансцендентним.*

*Наприклад,  $X - 10 \sin x = 0$ .*

3.2. З яких етапів складається процес знаходження кореня нелінійного рівняння? *Ізоляція або відділення коренів та уточнення коренів.*

3.3. Що означає ізолювати корінь рівняння? *Визначити найменші інтервали, на яких міститься один корінь.*

3.4. Перелічити методи уточнення коренів рівнянь. *Найбільшого поширення серед чисельних методів уточнення значень коренів знайшли: метод ділення навпіл; метод хорд; метод дотичних (Ньютона); комбінований метод; метод простих ітерацій.*

3.5. Логічна умова наявності кореня в розглянутому інтервалі. *Якщо безперервна та монотонна функція на кінцях інтервалу  $[a, b]$  має значення різних знаків, то на цьому інтервалі міститься хоча б один корінь. Умовою наявності кореня буде  $f(a)*f(b) \leq 0$ .*

3.6. Коли завершується процес уточнення кореня методом ділення інтервалу навпіл?

*Аналізуємо виконання умови  $|b-a| < E$ , де  $[a, b]$  — інтервал дослідження;  $E$  — точність уточнення кореня. Якщо умова виконується, закінчуємо алгоритм.*

3.7. Скласти Basic-програму для ізоляції коренів заданого рівняння методом послідовного перебору на підставі алгоритму на рис.Д.1.1.

3.8. Скласти Basic-програму для уточнення кореня методом ділення інтервалу навпіл на підставі алгоритму на рис.Д.1.2.

Текст програми:

**REM Ізоляція коренів трансцендентного рівняння**

**REM при моделюванні статичної системи**

**REM Програму склав студент .....**

**CLS**

**DEF fnf (x) =**

**INPUT “xp,xk,h”; xp,xk,h**

**a=xp**

**1 b=a+h**

**IF fnf(a)\*fnf(b)<=0 then PRINT “a=”;a, “b=”;b**

**a=a+h**

**IF a<xk then 1**

**Вихідні дані:**

**xp=.....,xk=.....,h=.....**

**Результат**

**a=..., b=.....**

```
REM Уточнення кореня трансцендентного рівняння
REM при моделюванні статичної системи
REM Програму склав студент .....
CLS
DEF fnf (x) = .....
INPUT "a,b,e"; a,b,e
4 t=(a+b)/2
IF fnf(t)=0 then x=t:GOTO 5
IF fnf(a)*fnf(t)>0 then a=t ELSE b=t
IF b-a>e then 4
  x=(a+b)/2
5 PRINT "x=";x
```

Вихідні дані:

a=...,b=...,e=.....

Результат

x=...

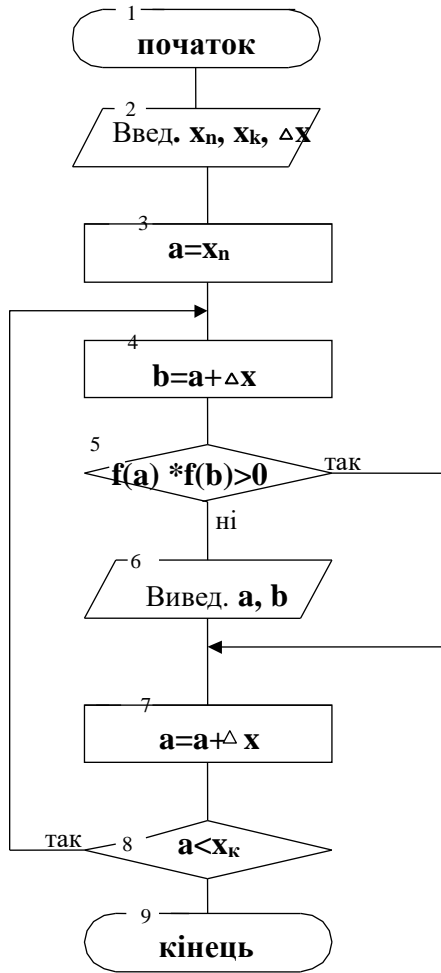


Рис. Д.1.1. Алгоритм відділення коренів (послідовний перебір)

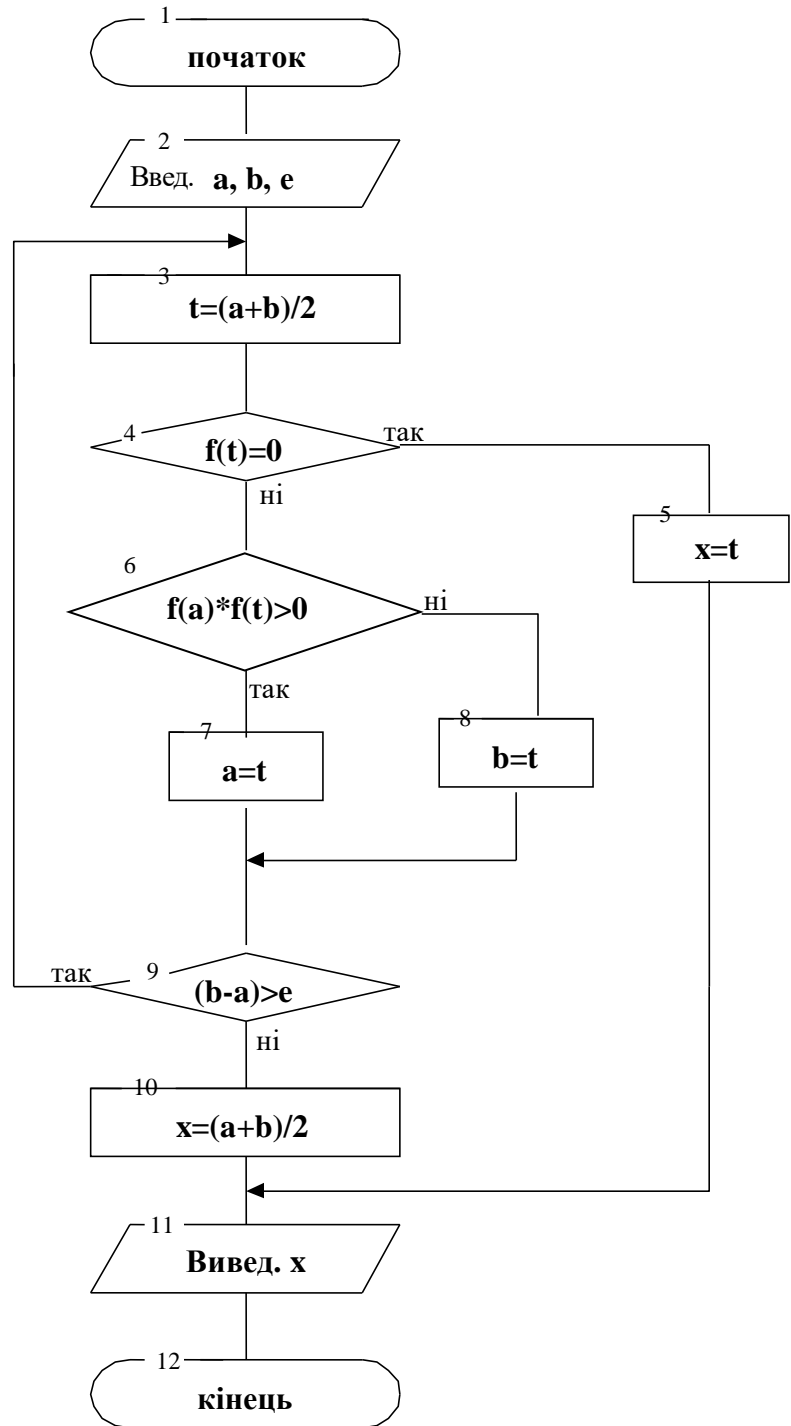


Рис. Д.1.2. Алгоритм уточнення кореня (ділення інтервалу навпіл)

## ВАРІАНТИ ДОДАТКОВИХ ІНДИВІДУАЛЬНИХ ЗАВДАНЬ

*Тема "Чисельні методи розв'язання задач моделювання"*

Для застосування кількісних методів дослідження в будь-якій області завжди потрібно побудувати ту або іншу математичну модель технологічного процесу. При побудові математичної моделі процес спрощується, схематизується із безлічі факторів, що впливають на процес, виділяються найважливіші, і одержана схема описується за допомогою того або іншого математичного апарату. У результаті отримуємо систему рівнянь — кількісний зв'язок між факторами.

Чим вдаліше підібрана модель, тим краще вона відображає характерні риси процесу, тим успішніше буде дослідження і корисніші рекомендації щодо удосконалення процесу.

Загальних способів побудови моделей не існує і вимоги до моделі суперечні: з одного боку, модель повинна бути досить повною — враховувати всі найважливіші фактори, з іншого боку, досить простою, щоб встановити реальні (аналітичні) залежності між параметрами.

Побудова математичної моделі — найбільш важлива і відповідальна частина дослідження, що вимагає глибоких знань не стільки математичних, скільки про модельовані процеси. Вдало побудована модель може знайти застосування для різних процесів. Наприклад, математичні моделі масового обслуговування знайшли застосування при дослідженні надійності технічних пристроїв, організації автоматизованого виробництва, розв'язання задач ППО та ін. [4].

Математичні моделі можна розділити на два класи: *аналітичні* і *статистичні*. Для аналітичних характерне встановлення формальних, аналітичних залежностей між параметрами задачі, записаних у будь-якому вигляді: система алгебраїчних рівнянь, звичайних диференціальних рівнянь, рівняння із першими похідними і т. ін. — використовуються для опису порівняно простих процесів із малою кількістю факторів. Статистичні моделі призначені для моделювання складних процесів із високою точністю і дозволяють враховувати набагато більшу кількість факторів.

Наприклад моделювання роботи пункту перестановки вагонів. Розглянемо процес заміни візків у пункті перестановки вагонів. У пункті перестановки вагонів для заміни візків состави подаються за нормальним законом розподілу з математичним очікуванням 60 хв та середнім квадратичним відхиленням 10 хв. Заміна візків проводиться одночасно для всього состава. Час, витрачений бригадою на заміну візків для одного состава, рівномірно розподілений на інтервалі 40-60 хв. У пункті перестановки вагонів працює одна бригада, состави подаються на одну колію. Розробити модель заміни візків составів та проаналізувати роботу пункту перестановки вагонів, виходячи з того, що тривалість зміни складає 10 год.

Задача полягає в синхронізації двох процесів: перший процес — подача составів до пункту перестановки вагонів  $t_1$ , другий процес — безпосередньо заміна візків  $t_2$ .

*Розглянемо фактори, що забезпечують роботу пункту перестановки вагонів, опишемо для них систему рівнянь, складемо алгоритм та програму.*

*Перший фактор* —  $k$  — тривалість зміни.

Розраховується за формулою  $k = 10 \text{ год} = 10 * 60 \text{ хв}$ .

Використовується для формування умови щодо закінчення процесу моделювання.

*Другий фактор* —  $n$  — кількість составів, що можливо обробити у пункті перестановки вагонів, якщо не враховувати другий процес.

Розраховується за формулою  $n = k / 40$  як тривалість зміни, поділена на мінімальний час одного з процесів.

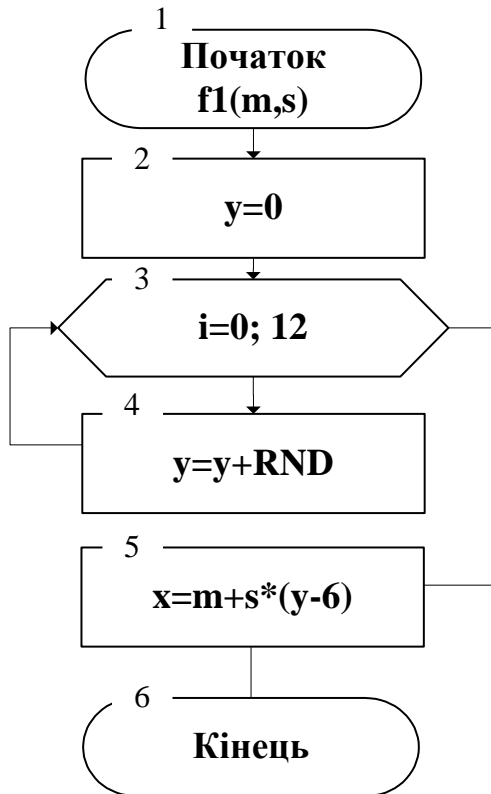
*Третій фактор* —  $t_1(i)$ ,  $i=1, \infty$  — час надходження кожного із составів до пункту перестановки вагонів. Состави надходять за нормальним законом розподілу з математичним очікуванням 60 та середнім квадратичним відхиленням 10. Для формування часу надходження кожного состава до пункту перестановки вагонів  $t_1(i)$ ,  $i=1, \infty$  потрібно згенерувати масив випадкових чисел. Для формування випадкових значень для цього масиву використаємо процедуру-функцію  $f_1$ , для якої вхідними параметрами виступають математичне очікування та середнє квадратичне відхилення (рис. Д.2.2).

На практиці для отримання дійсного значення випадкової величини використовують формулу

$$x = m + \left( \sum_{i=1}^{12} RND - 6 \right) s,$$

де RND – випадково згенероване дійсне число від 0 до 1; m – математичне очікування; s – середнє квадратичне відхилення (у нашому випадку m=60 хв, s=10 хв).

Схема алгоритму для визначення випадкової величини за нормальним законом розподілу наведена на рис. Д.2.1.



Текст процедури-функції f1(m, s):

```

FUNCTION f1% (m, s)
  y = 0
  DIM x AS INTEGER
  FOR i = 0 TO 12 - 1
    y = y + RND
  NEXT i
  x = m + s * (y - 6)
  f1% = x
END FUNCTION
  
```

Рис. Д.2.1. Схема алгоритму функції f1(m,s)

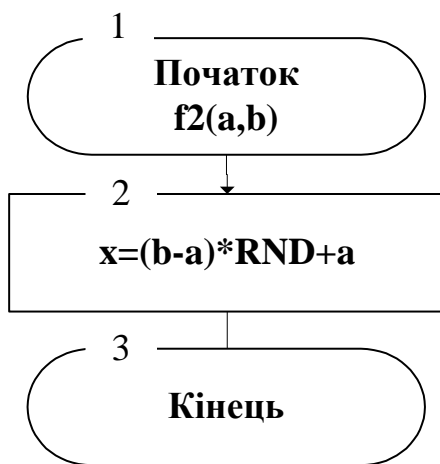
Четвертий фактор —  $t_2(i)$  — час, що необхідно затратити на заміну візків. Процес заміни візка підпорядкований рівномірному закону розподілу на інтервалі від 40 хв до 60 хв. Для формування часу заміни візків  $t_2(i)$ ,  $i=1, \infty$  потрібно сформуувати послідовність випадкових чисел за рівномірним законом розподілу. Для формування випадкових значень цього масиву використаємо процедуру-функцію  $f_2$ , для якої вхідними параметрами виступають початкове та кінцеве значення інтервалу розподілу.

На практиці для отримання дійсного значення випадкової величини, рівномірно розподіленої на інтервалі від  $a$  до  $b$ , використовують формулу

$$x = (b - a) * \text{RND} + a,$$

де  $a$ ,  $b$  — відповідно початок і кінець інтервалу розподілу (у нашому випадку  $a=40$  хв,  $b=60$  хв).

Схема алгоритму для визначення випадкової величини за рівномірним законом розподілу наведена на рис. Д.2.2.



Текст процедури-функції  $f_2(a, b)$ :

```

FUNCTION f2% (a, b)
DIM x AS INTEGER
  x = (b - a) * RND + a
  f2% = x
END FUNCTION
  
```

Рис. Д.2.2. Схема алгоритму функції  $f_2(a, b)$



Для побудови моделі роботи пункту перестановки вагонів треба вирахувати:

П'ятий фактор —  $t(i)$ ,  $i=1, \infty$  — час прибуття составів на заміну візків. Для формування часу прибуття составів на заміну візків використовують формулу

$$t(i) = t(i - 1) + t1(i - 1) .$$

Шостий фактор —  $tn(i)$ ,  $i=1, \infty$  — час початку заміни візків у пункті перестановки вагонів дорівнює часу надходження состава, якщо попередній состав завершив заміну візків, або часу закінчення обслуговування попереднього состава та розраховується за формулою

$$tn(i) = \begin{cases} tk(i - 1); & \text{якщо } t(i) < tk(i - 1); \\ t(i); & \text{якщо } t(i) \geq tk(i - 1). \end{cases}$$

Сьомий фактор —  $tk(i)$ ,  $i=1, \infty$  — час завершення заміни візків у пункті перестановки вагонів, розраховується за формулою

$$tk(i) = tn(i) + t2(i) .$$

Восьмий фактор —  $prt1(i)$ ,  $i=1, \infty$  — час простою составів у пункті перестановки вагонів виникає, якщо бригада ще не завершила заміну колісної пари попереднього состава, і розраховується за формулою

$$prt1(i) = \begin{cases} 0, & \text{якщо } t1(i) \geq tk(i - 1); \\ tk(i - 1) - t(i), & \text{якщо } t1(i) < tk(i - 1). \end{cases}$$

Дев'ятий фактор —  $prt2(i)$ ,  $i=1, \infty$  — час простою бригади в пункті перестановки вагонів виникає, якщо состав ще не подано на заміну колісної пари, і розраховується за формулою

$$prt2(i) = \begin{cases} t1(i) - tk(i - 1), & \text{якщо } t(i) > tk(i - 1); \\ 0, & \text{якщо } t1(i) \leq tk(i - 1). \end{cases}$$

Складемо перелік ідентифікаторів, що необхідні для побудови моделі (табл. Д.2.1).

Таблиця Д.2.1  
Ідентифікатори для побудови моделі

Ідентифікатор	Опис ідентифікатора	Тип даних	Формат виведення
m,s	математичне очікування та середнє квадратичне відхилення часу надходження состава до пункту перестановки вагонів	INTEGER, INTEGER	##, ##
a,b	граничні значення часу заміни візків	INTEGER, INTEGER	##, ##
n	планова кількість составів	INTEGER	##
k	тривалість зміни	INTEGER	###
i	номер оброблюваного состава, $i=1,n$	INTEGER	##
t1(i)	час подачі кожного состава до пункту перестановки вагонів	INTEGER	##
t2(i)	час заміни візків для кожного состава	INTEGER	##
t(i)	час прибуття состава на заміну візків за планом	INTEGER	###
tn(i)	час початку заміни візків	INTEGER	###
tk(i)	час закінчення заміни візків	INTEGER	###
prt1(i)	час простою состава	INTEGER	##
prt2(i)	час простою бригади	INTEGER	##
kol	кількість составів у даному модельному процесі	INTEGER	##

Блоки 2-32 схеми алгоритму на рис. Д.2.3 реалізують розрахунок наведених вище факторів.

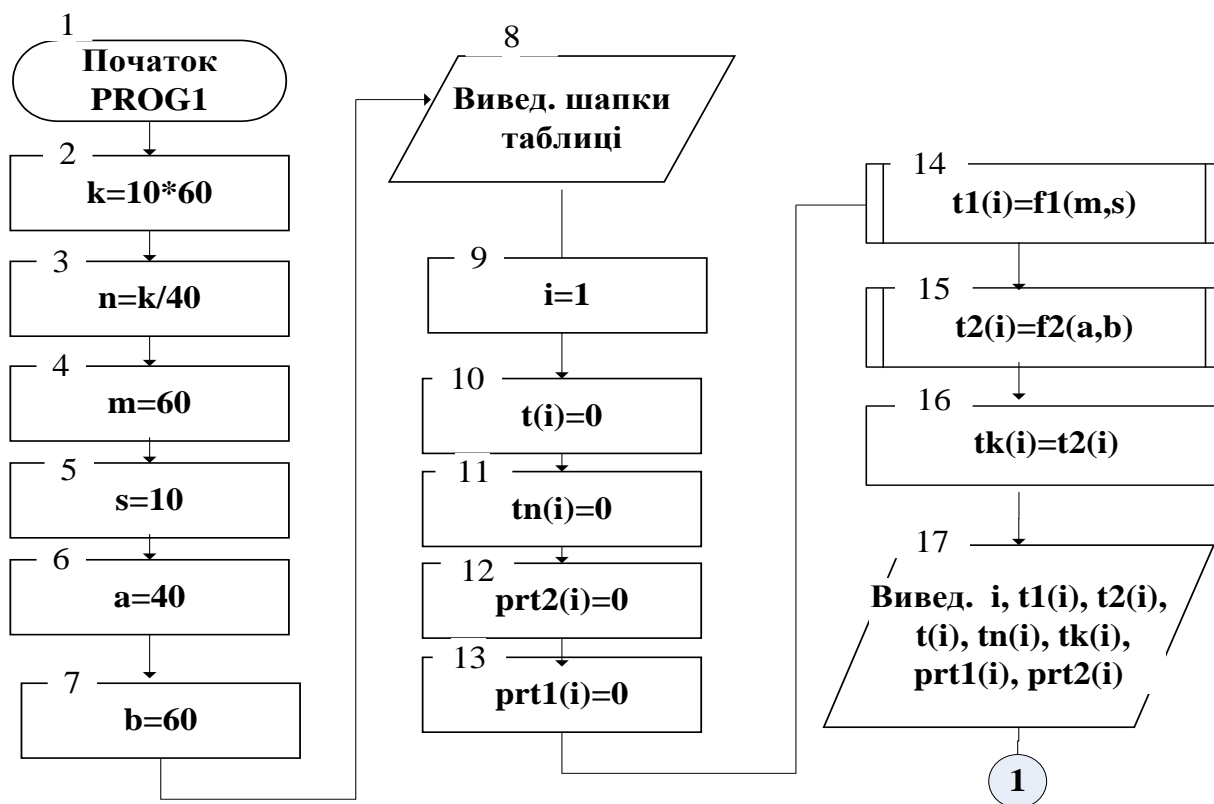


Рис. Д.2.3. Схема алгоритму підпрограми PROG1  
(див. також с.172,173)

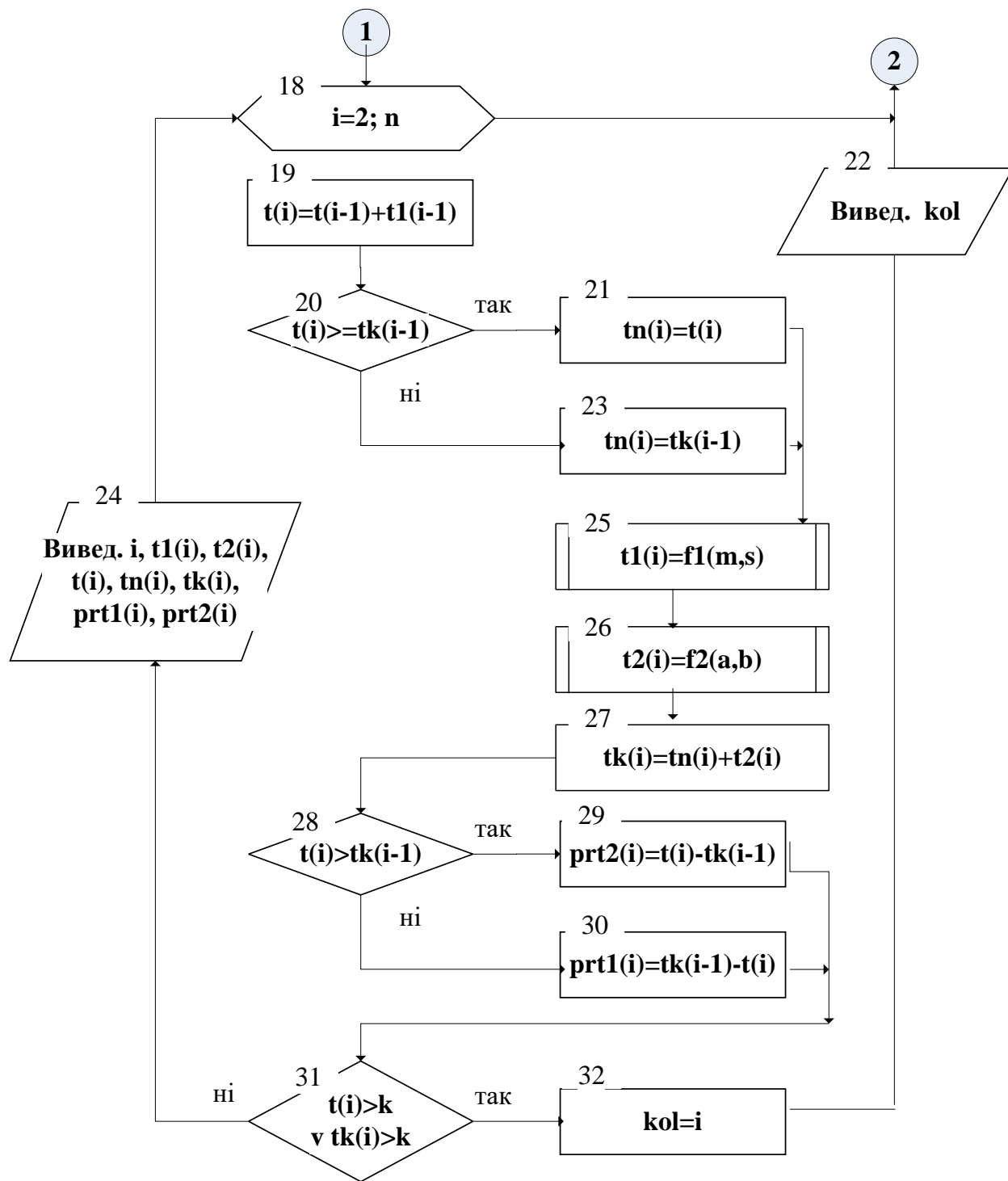


Рис. Д.2.3. Продовження

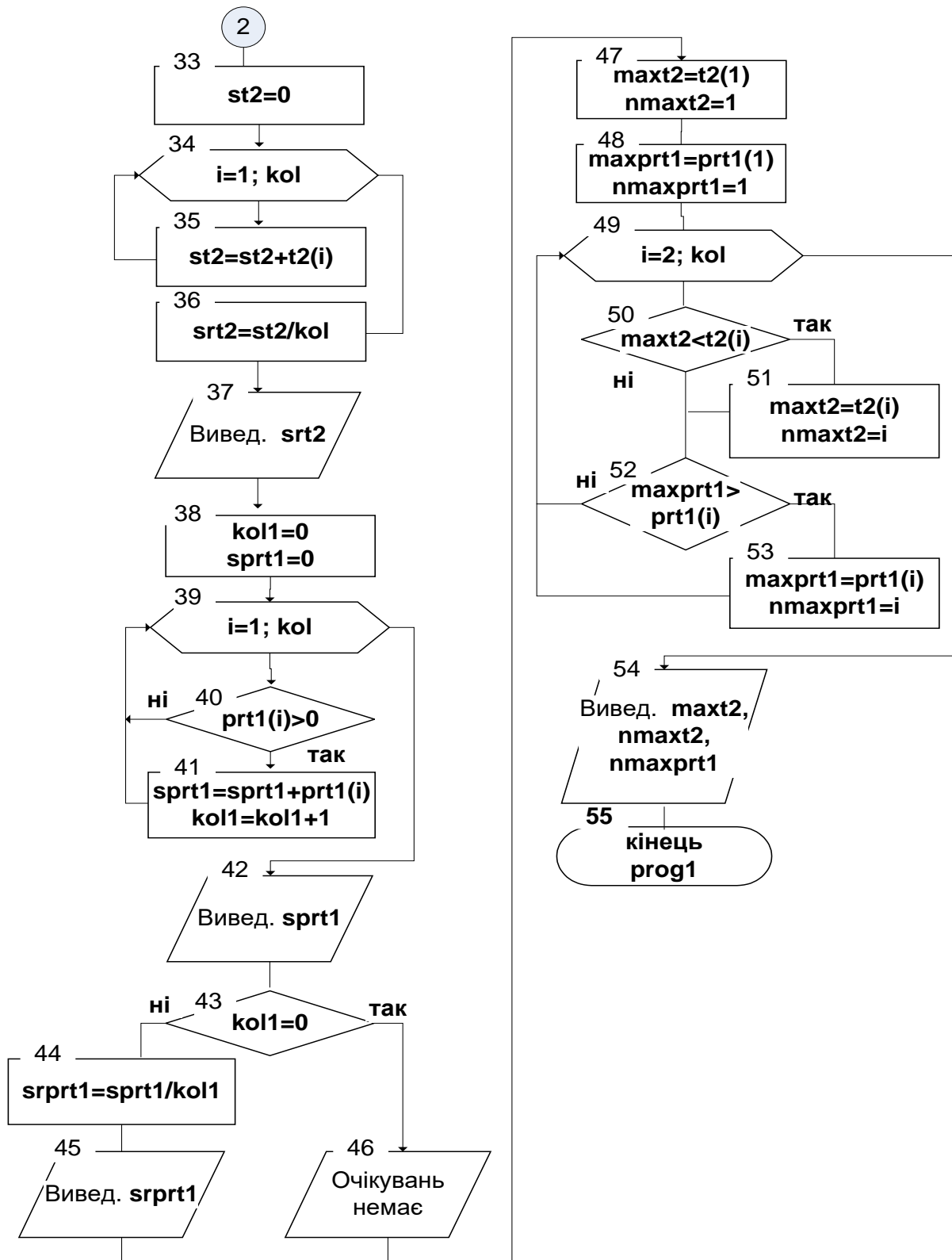


Рис. Д.2.3. Закінчення

*Опис алгоритму:*

- 4, 5, 9, 10, 14, 17, 18, 25, 24 – формування випадкових величини — моментів прибуття составів до пункту перестановки вагонів
- 6, 7, 9, 15, 17, 18, 26, 24 – формування випадкової величини — часу заміни візків кожного состава
- 2, 3, 31, 22, 32 – визначення тривалості зміни в хвиликах і планової кількості составів
- 10, 17, 18, 19, 24 – визначення часу надходження составів
- 11, 16, 17, 18, 20, 21, 23, 27, 24 – визначення часу початку та закінчення заміни візків поточного состава
- 12, 13, 17, 18, 28, 29, 30, 24 – визначення часу простою бригади або очікування заміни составом
- 33, 34, 35, 36, 37 – накопичення суми часу заміни візків, вирахування та виведення на друк середнього часу заміни візків
- 38, 39, 40, 41, 42 – накопичення та виведення на друк суми часу очікування заміни візків та їх кількості
- 38, 39, 40, 41, 42 – накопичення та виведення на друк загальної суми часу очікування заміни візків та їх кількості
- 43, 44, 45, 46 – аналіз та вирахування середнього часу очікування заміни візків
- 47, 49, 50, 51, 54 – вирахування найбільшого часу заміни візків та номера состава, що заміняв візки довше за всіх
- 48, 49, 52, 53, 54 – вирахування номера состава, що очікував заміну візків довше за всіх

## Текст підпрограми PROG1:

```

SUB prog1 (kol, t1(), t2(), t(), tn(), tk(), prt1(), prt2(), k, n)
m = 60: s = 10
a = 40: b = 60
'INPUT "Vvedite m,s", m, s
'INPUT "Vvedite a,b", a, b
' RANDOMIZE TIMER

CLS
PRINT "  Результати моделювання роботи"
PRINT "    пункту перестановки вагонів."
PRINT "  Дослідження проводив Іванов І.І. гр. 3-Ш-Л"
PRINT "-----"
PRINT "| N | час   | час   | час   | початок|завер-|очікування|      |"
PRINT "| з.п|подачі |заміни|прибут.|заміни |шення|составом |очікування|"
PRINT "|   |состава| візків|за пл.  |       |заміни|          |бригадою |"
PRINT "| i | t1(i)  | t2(i) | t(i)   | tn(i)  | tk(i) | prt1(i)  | prt2(i) |"
PRINT "-----"

    i = 1
    t1(i) = f1%(m, s)
    t2(i) = f2%(a, b)
    tk(i) = t2(i)
    t(i) = 0: tn(i) = 0: prt1(i) = 0: prt2(i) = 0
PRINT USING "|##| ## |## | ###| ### | ###| ##| ##|"; i; t1(i); t2(i); t(i); tn(i);
tk(i); prt1(i); prt2(i)

    FOR i = 2 TO n
        t(i) = t(i - 1) + t1(i - 1)
    IF t(i) >= tk(i - 1) THEN tn(i) = t(i) ELSE tn(i) = tk(i - 1)
        t1(i) = f1%(m, s)
        t2(i) = f2%(a, b)
        tk(i) = tn(i) + t2(i)
    IF t(i) > tk(i - 1) THEN prt2(i) = t(i) - tk(i - 1) ELSE prt1(i) = tk(i - 1) - t(i)
    IF t(i) >= k OR tk(i) >= k THEN kol = i: PRINT USING "|##| ## |## | ###| ###
| ###| ##| ##|"; i; t1(i); t2(i); t(i); tn(i); tk(i); prt1(i); prt2(i): EXIT FOR
PRINT USING "|##| ## |## | ###| ### | ###| ##| ##|"; i; t1(i); t2(i); t(i); tn(i);
tk(i); prt1(i); prt2(i)

    IF i = 12 OR i = 34 OR i = 56 OR i = 78 THEN INPUT "", qq
    NEXT i
PRINT "-----"
INPUT ""; qq

```

```

PRINT "Проведемо аналіз результатів моделювання:"
st2 = 0: kol1 = 0: sprt1 = 0
FOR i = 1 TO kol
    st2 = st2 + t2(i)
    IF prt1(i) > 0 THEN sprt1 = sprt1 + prt1(i): kol1 = kol1 + 1
NEXT i
srt2 = st2 / kol

maxt2 = t2(1): nmaxt2 = 1: maxprt1 = t1(1): nmaxprt1 = 1
FOR i = 2 TO kol
    IF t2(i) > maxt2 THEN maxt2 = t2(i): nmaxt2 = i
    IF prt1(i) > maxprt1 THEN maxprt1 = prt1(i): nmaxprt1 = i
NEXT i

PRINT USING "Кількість оброблених составів ## "; kol
PRINT USING "Середній час заміни візків ##.# "; srt2
PRINT USING "Загальний час очікування заміни ##### "; sprt1
IF kol1=0 THEN
PRINT "очікувань заміни візків - немає"
ELSE
srprt1=sprt1/kol1
PRINT USING "Середній час очікування заміни візків ##.#"; srprt1
END IF
PRINT USING "Найбільший час заміни візків ## "; maxt2
PRINT "Номер состава, що заміняв візки "
PRINT USING "довше за всіх ##"; nmaxt2
PRINT "Номер состава, що очікував заміну візків"
PRINT USING "довше за всіх ## "; nmaxprt1
INPUT ""; qqq

LOCATE 23, 1: INPUT "Якщо побудувати графік, натисніть 1", qqq
IF qqq = 1 THEN CALL PROG2(kol, t1(), t2(), t(), tn(), tk(), prt1(), prt2(), k)
    SCREEN 0
    INPUT ""; qqq
END SUB

```



Проведемо візуалізацію даних.

Схема алгоритму для побудови графіка наведена на рис. Д.2.4.

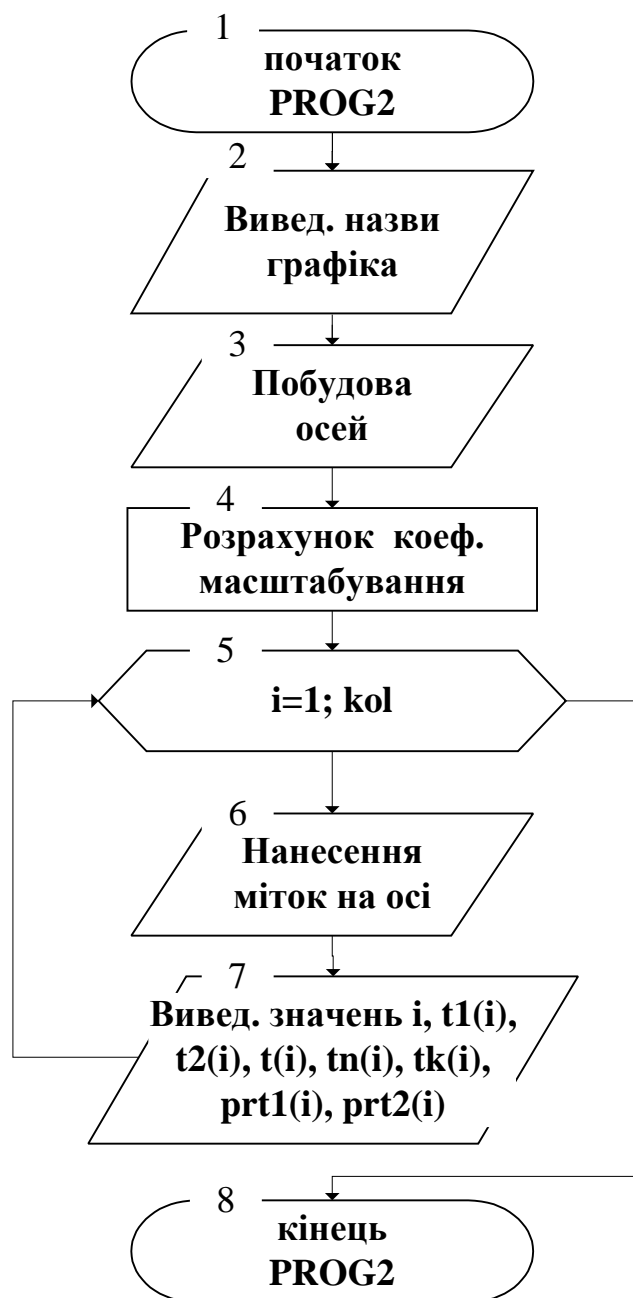


Рис. Д.2.4. Схема алгоритму підпрограми PROG2

*Текст підпрограми PROG2:*

```

SUB PROG2 (kol, t1(), t2(), t(), tn(), tk(), prt1(), prt2(), k)
  SCREEN 12
  DIM i AS INTEGER
  PRINT "      Графік "
  ' gorizontal:
  ' t, t1
  LINE (10, 40)-(600, 40) : LINE (630, 40)-(600, 30) : LINE (630, 40)-(600, 50)
  ' tn,t2,tk
  LINE (10, 90)-(600, 90) : LINE (630, 90)-(600, 80) : LINE (630, 90)-(600, 100)
  ' prt1
  LINE (10, 145)-(600, 145) : LINE (630, 145)-(600, 135) : LINE (630, 145)-(600, 155)
  ' prt1
  LINE (10, 195)-(600, 195) : LINE (630, 195)-(600, 185) : LINE (630, 195)-(600, 205)
  i = 1
  ' vertikal:
  LINE (10, 30)-(10, 50)
  ' tn,t2,tk
  LINE (10, 90)-(10, 105 + i) : LINE (tk(i), 90 + i)-(tk(i), 100 + i)
  LINE (10, 100 + i)-(10 + t2(i), 100 + i)
  ' prt1, prt2
  IF t(2) < tk(1) THEN LINE (t(2), 150)-(t(2) + prt1(2), 150)
  IF t(2) > tk(1) THEN LINE (tk(1), 200)-(tk(1) + prt2(1), 200)
  ' text
  xdtex = FIX(76 * 10 / k) : LOCATE 4, xdtex: PRINT t(i)
  FOR i = 2 TO kol
  ' vertikal: t, t1
  LINE (t(i), 30)-(t(i), 50)
  ' tn,t2,tk
  LINE (tn(i), 90)-(tn(i), 105 + i) : LINE (tk(i), 90)-(tk(i), 100 + i)
  LINE (tn(i), 100 + i)-(tn(i) + t2(i), 100 + i)
  ' prt1, prt2
  IF t(i) < tk(i - 1) THEN LINE (t(i), 150)-(tk(i - 1), 150)
  IF t(i) > tk(i - 1) THEN LINE (tk(i - 1), 200)-(tn(i), 200)
  ' text t
  xdtex = FIX(76 * t(i) / k) : LOCATE 4, xdtex: PRINT t(i)
  ' text prt1, prt2
  xdtex = FIX(76 * t(i) / k)
  IF t(i) < tk(i - 1) THEN LOCATE 11, xdtex: PRINT prt1(i)
  IF t(i) > tk(i - 1) THEN LOCATE 15, xdtex: PRINT prt2(i)
  NEXT i : SLEEP : SCREEN 0 :
END SUB

```

Нанесемо дані на відповідні осі — складемо модель. На часових осях відображена послідовність подій, що моделюються (рис. Д.2.5).

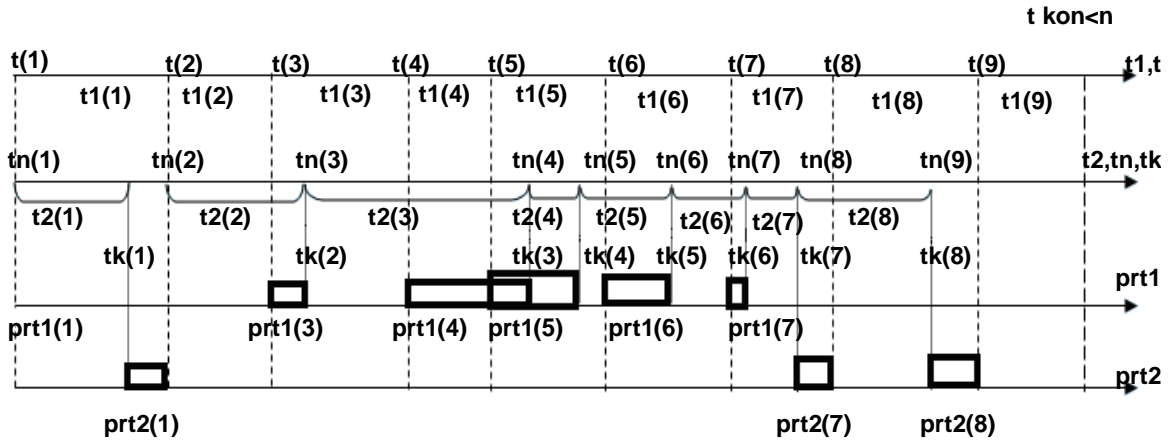


Рис. Д.2.5. Заміна візків у пункті перестановки вагонів

Проведемо аналіз отриманих даних. Для ефективного управління процесом заміни візків треба визначити такі значення:

—  $maxt2$  — найбільший час заміни візків у пункті перестановки вагонів, що розраховується за формулою

$$maxt2 = \{t2(i)\}_{\max};$$

—  $nmact2$  — номер состава, що заміняв візки в пункті перестановки вагонів довше за всіх. Розраховується за формулою

$$nmact2 = \{t2(i), i\}_{\max};$$

—  $nmaxprt1$  — номер состава, що очікував заміну візків у пункті перестановки вагонів довше за всіх. Розраховується за формулою

$$nmaxprt1 = \{prt1(i), i\}_{\max};$$

—  $srt2$  — середній час заміни візків у пункті перестановки вагонів. Розраховується за формулою

$$srt2 = \frac{1}{kol} \sum_{i=0}^{kol} t2(i) \quad ;$$

—  $srprt1$  — середній час очікування заміни візків у пункті перестановки вагонів. Розраховується за формулою

$$srprt1 = \frac{1}{kol} \sum_{i=0}^{kol} prt1(i);$$

—  $sprt1$  — загальний час очікування заміни візків. Розраховується за формулою

$$sprt1 = \sum_{i=0}^{kol} prt1(i);$$

—  $kol$  — кількість составів, що надійшли до пункту перестановки вагонів за зміну.

Складемо перелік ідентифікаторів, що необхідні для аналізу моделі (табл. Д.2.2).

Таблиця Д.2.2

## Ідентифікатори для аналізу моделі

Ідентифікатор	Опис ідентифікатора	Тип даних	Формат виведення
$maxt2$	найбільший час заміни візків	INTEGER	##
$nmaxt2$	номер состава, що заміняв візки в пункті перестановки вагонів довше за всіх	INTEGER	##
$nmaxprt1$	номер состава, що очікував заміну візків у пункті перестановки вагонів довше за всіх	INTEGER	##
$srt2$	середній час заміни візків	SINGLE	##.#
$srprt1$	середній час очікування заміни візків	SINGLE	##.#
$sprt1$	загальний час очікування заміни візків	INTEGER	####
$kol1$	кількість составів, що очікували заміну візків	INTEGER	##
$kol$	загальна кількість составів	INTEGER	##

*Текст основної програми:*

```

DECLARE FUNCTION f1% (m, s)
DECLARE FUNCTION f2% (a, b)
DECLARE SUB PROG2 (kol, t1(), t2(), t(), tn(), tk(), prt1(), prt2(), k)
DECLARE SUB prog1 (kol, t1(), t2(), t(), tn(), tk(), prt1(), prt2(), k, n)
k = 10 * 60 : n = FIX(k / 40)
DIM t1(n) : DIM t2(n) AS SINGLE
DIM t(n) AS SINGLE : DIM tn(n) AS SINGLE
DIM tk(n) AS SINGLE : DIM prt1(n) AS SINGLE
DIM prt2(n) AS SINGLE : DIM i AS INTEGER
DIM a AS INTEGER : DIM b AS INTEGER
DIM m AS INTEGER : DIM s AS INTEGER
DO
    CLS
    LOCATE 2, 9: PRINT "Натисніть 1 для розрахунків"
    LOCATE 3, 9: PRINT "Натисніть 2 для виходу"
    LOCATE 4, 9: INPUT ""; v
    SELECT CASE v
        CASE 1 : CALL prog1(kol, t1(), t2(), t(), tn(), tk(), prt1(), prt2(), k, n)
        CASE 2 : GOTO m1
    END SELECT
LOOP WHILE (1)
m1: CLS : PRINT "Програма завершила роботу"
    Модульна схема програми наведена на рис. Д.2.6.
  
```

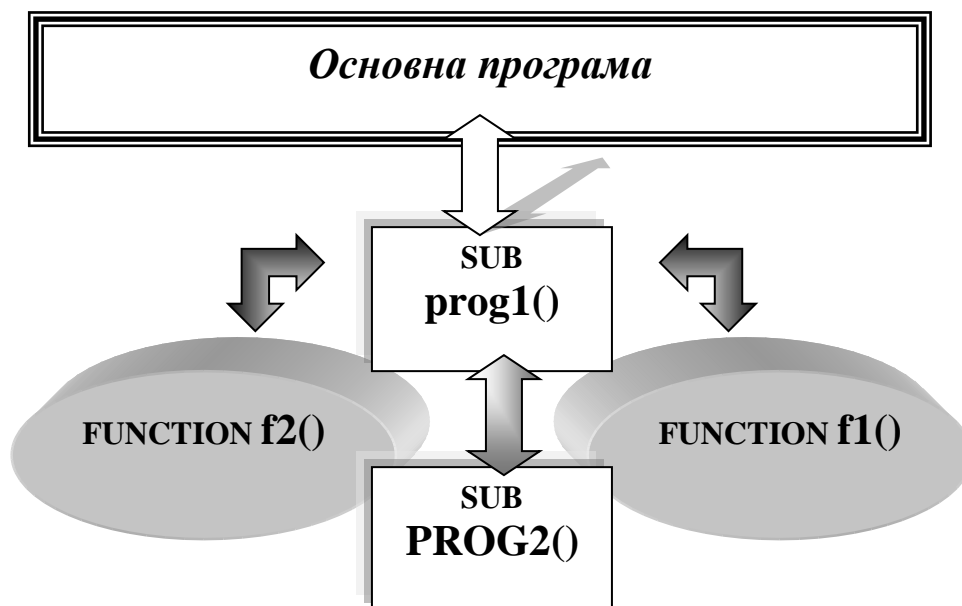


Рис. Д.2.6. Модульна структура програми

Результати досліджень. Результат моделювання наведений у табл. Д.2.3.

Таблиця Д.2.3

**Результати моделювання роботи  
пункту перестановки вагонів.  
Дослідження проводив Іванов І.І. гр. 3-Ш-Л"**

<b>N</b> <b>з.п</b>	<b>час</b> <b>подачі</b> <b>состава</b>	<b>час</b> <b>заміни</b> <b>візків</b>	<b>час</b> <b>прибут.</b> <b>за пл.</b>	<b>початок</b> <b>заміни</b>	<b>завер-</b> <b>шення</b>	<b>очікування</b> <b>составом</b>	<b>очікування</b> <b>заміни</b> <b>бригадою</b>
<b>i</b>	<b>t1(i)</b>	<b>t2(i)</b>	<b>t(i)</b>	<b>tn(i)</b>	<b>tk(i)</b>	<b>prt1(i)</b>	<b>prt2(i)</b>
1	60	40	0	0	40	0	0
2	61	60	60	60	120	20	0
3	30	55	121	121	176	1	0
4	61	58	151	176	234	0	25
5	55	57	212	234	291	0	22
6	40	59	267	291	374	0	24
7	60	49	307	350	399	0	43
8	62	45	367	399	444	0	32
9	63	47	429	444	491	0	15
10	67	41	492	492	533	1	0
11	58	51	559	559	610	26	0

**Проведемо аналіз результатів моделювання:**

**Кількість оброблених составів 11**

**Середній час заміни візків 51.09**

**Загальний час очікування заміни 47**

**Середній час очікування заміни візків 15,66**

**Найбільший час заміни візків 60**

**Номер состава, що заміняв візки**

**довше за всіх 2**

**Номер состава, що очікував заміну візків**

**довше за всіх 11**

*Варіанти індивідуальних завдань до роботи*

Провести моделювання роботи пункту перестановки вагонів, якщо:

1. У пункті перестановки вагонів для заміни візків состави подаються за нормальним законом розподілу з математичним очікуванням 30 хв та середнім квадратичним відхиленням 5 хв. Час, витрачений бригадою на заміну візків для одного состава, розподілений за нормальним законом розподілу з математичним очікуванням 45 хв та середнім квадратичним відхиленням 5 хв. Проаналізувати роботу пункту перестановки вагонів, виходячи з того, що тривалість зміни складає 12 год.

2. У пункті перестановки вагонів для заміни візків состави подаються за нормальним законом розподілу з математичним очікуванням 40 хв та середнім квадратичним відхиленням 15 хв. Час, витрачений бригадою на заміну візків для одного состава, рівномірно розподілений на інтервалі 50-70 хв. Проаналізувати роботу пункту перестановки вагонів, виходячи з того, що тривалість зміни складає 18 год.

3. У пункті перестановки вагонів для заміни візків час подачі состава рівномірно розподілений на інтервалі 55-65 хв. Час, витрачений бригадою на заміну візків для одного состава, розподілений за нормальним законом розподілу з математичним очікуванням 50 хв та середнім квадратичним відхиленням 20 хв. Проаналізувати роботу пункту перестановки вагонів, виходячи з того, що тривалість зміни складає 10 год.

4. У пункті перестановки вагонів для заміни візків час подачі состава рівномірно розподілений на інтервалі 55-60 хв. Час, витрачений бригадою на заміну візків для одного состава, рівномірно розподілений на інтервалі 40-65 хв. Проаналізувати роботу пункту перестановки вагонів, виходячи з того, що тривалість зміни складає 8 год.

5. У пункті перестановки вагонів для заміни візків состави подаються за нормальним законом розподілу з математичним очікуванням 55 хв та середнім квадратичним відхиленням 20 хв. Час, витрачений бригадою на заміну візків для одного состава, розподілений за нормальним законом розподілу з математичним очікуванням 55 хв та середнім квадратичним відхиленням 10 хв. Проаналізувати роботу пункту перестановки вагонів, виходячи з того, що тривалість зміни складає 12 год.

6. У пункті перестановки вагонів для заміни візків состави подаються за нормальним законом розподілу з математичним очікуванням 55 хв та середнім квадратичним відхиленням 10 хв. Час, витрачений бригадою на заміну візків для одного состава, рівномірно розподілений на інтервалі 55-70 хв. Проаналізувати роботу пункту перестановки вагонів, виходячи з того, що тривалість зміни складає 12 год.

7. У пункті перестановки вагонів для заміни візків час подачі состава рівномірно розподілений на інтервалі 20-60 хв. Час, витрачений бригадою на заміну візків для одного состава, розподілений за нормальним законом розподілу з математичним очікуванням 60 хв та середнім квадратичним відхиленням 15 хв. Проаналізувати роботу пункту перестановки вагонів, виходячи з того, що тривалість зміни складає 18 год.

8. У пункті перестановки вагонів для заміни візків час подачі состава рівномірно розподілений на інтервалі 30-50 хв. Час, витрачений бригадою на заміну візків для одного состава, рівномірно розподілений на інтервалі 45-70 хв. Проаналізувати роботу пункту перестановки вагонів, виходячи з того, що тривалість зміни складає 18 год.

9. У пункті перестановки вагонів для заміни візків состави подаються за нормальним законом розподілу з математичним очікуванням 45 хв та середнім квадратичним відхиленням 15 хв. Час, витрачений бригадою на заміну візків для одного состава, розподілений за нормальним законом розподілу з математичним очікуванням 58 хв та середнім квадратичним відхиленням 10 хв. Проаналізувати роботу пункту перестановки вагонів, виходячи з того, що тривалість зміни складає 8 год.

10. У пункті перестановки вагонів для заміни візків состави подаються за нормальним законом розподілу з математичним очікуванням 60 хв та середнім квадратичним відхиленням 10 хв. Час, витрачений бригадою на заміну візків для одного состава, рівномірно розподілений на інтервалі 30-50 хв. Проаналізувати роботу пункту перестановки вагонів, виходячи з того, що тривалість зміни складає 10 год.



11. У пункті перестановки вагонів для заміни візків час подачі состава рівномірно розподілений на інтервалі 40-50 хв. Час, витрачений бригадою на заміну візків для одного состава, розподілений за нормальним законом розподілу з математичним очікуванням 40 хв та середнім квадратичним відхиленням 17 хв. Проаналізувати роботу пункту перестановки вагонів, виходячи з того, що тривалість зміни складає 6 год.

12. У пункті перестановки вагонів для заміни візків час подачі состава рівномірно розподілений на інтервалі 20-50 хв. Час, витрачений бригадою на заміну візків для одного состава, рівномірно розподілений на інтервалі 45-55 хв. Проаналізувати роботу пункту перестановки вагонів, виходячи з того, що тривалість зміни складає 6 год.

13. У пункті перестановки вагонів для заміни візків состави подаються за нормальним законом розподілу з математичним очікуванням 30 хв та середнім квадратичним відхиленням 10 хв. Час, витрачений бригадою на заміну візків для одного состава, розподілений за нормальним законом розподілу з математичним очікуванням 65 хв та середнім квадратичним відхиленням 5 хв. Проаналізувати роботу пункту перестановки вагонів, виходячи з того, що тривалість зміни складає 18 год.

14. У пункті перестановки вагонів для заміни візків состави подаються за нормальним законом розподілу з математичним очікуванням 20 хв та середнім квадратичним відхиленням 10 хв. Час, витрачений бригадою на заміну візків для одного состава, рівномірно розподілений на інтервалі 55-65 хв. Проаналізувати роботу пункту перестановки вагонів, виходячи з того, що тривалість зміни складає 18 год.

15. У пункті перестановки вагонів для заміни візків час подачі состава рівномірно розподілений на інтервалі 20-50 хв. Час, витрачений бригадою на заміну візків для одного состава, розподілений за нормальним законом розподілу з математичним очікуванням 48 хв та середнім квадратичним відхиленням 12 хв. Проаналізувати роботу пункту перестановки вагонів, виходячи з того, що тривалість зміни складає 10 год.

## Тема "Побудова матричних моделей"

**Задача 1**

Заданий двовимірний масив числових значень -  $\{s_{ij}\}$ ,  $i=1,2,3$ ;  $j=1,2,3,4$ .

1. У **головній програмі** здійснити введення елементів  $s_{ij}$ , виклик необхідних процедур і виведення результатів.
2. За допомогою **процедури-підпрограми** сформувати  $\{x_j\}$ ,  $j=1,2,3,4$  з максимальних елементів кожного стовпця масиву  $\{s_{ij}\}$ .
3. За допомогою **процедури-підпрограми** транспонувати (поміняти місцями рядки і стовпці) масив  $\{s_{ij}\}$ .
4. За допомогою **процедури-функції** підрахувати суму всіх додатних елементів масиву  $\{s_{ij}\}$ .

**Задача 2**

Заданий двовимірний масив числових значень –  $\{d_{ij}\}$ ,  $i=1,2,\dots,n$ ;  $j=1,2,\dots,m$ .

1. У **головній програмі** здійснити введення елементів  $\{d_{ij}\}$ , виклик необхідних процедур і виведення результатів.
2. За допомогою **процедури-підпрограми** сформувати масив  $\{c_i\}$ ,  $i=1,2,\dots,n$  з середніх арифметичних елементів кожного рядка масиву  $\{d_{ij}\}$ .
3. За допомогою **процедури - підпрограми** сформувати масив  $\{y_t\}$  з елементів  $-12 < d_{ij} < 25$  та підрахувати кількість елементів у  $\{y_t\}$ .
4. За допомогою **процедури - функції** знайти мінімальне значення серед елементів масиву  $\{d_{ij}\}$ .

**Задача 3**

Задані двовимірні масиви числових значень –  $\{a_{ij}\}$  та  $\{b_{ij}\}$ ,  $i=1,2,\dots,n$ ;  $j=1,2,\dots,m$ .

1. У **головній програмі** здійснити введення елементів  $a_{ij}$  та  $b_{ij}$ , виклик необхідних процедур і виведення результатів.
2. За допомогою **процедури-підпрограми** сформувати масив  $\{c_{ij}\}$ ,  $i=1,n$ ;  $j=1,m$ , де  $\{c_{ij}\} = \max(a_{ij}, b_{ij})$  ( $c_{11} = \max(a_{11}, b_{11})$   $c_{12} = \max(a_{12}, b_{12}), \dots$ ).
3. За допомогою **процедури-функції** підрахувати кількість елементів парних рядків масиву  $\{c_{ij}\}$ , що задовольняють умову  $f < c_{ij} < d$  ( $f, d$ - довільні числа,  $f < d$ ).
4. За допомогою **процедури-функції** знайти різницю між найбільшим і найменшим значеннями масиву  $\{b_{ij}\}$ .

**Задача 4**

Заданий двовимірний масив числових значень –  $\{f_{ij}\}$ ,  $i=1,2,\dots,m$ ;  
 $j=1,2,\dots,m$ .

1. У **головній програмі** здійснити введення елементів  $f_{ij}$ , виклик необхідних процедур і виведення результатів.
2. За допомогою **процедури-підпрограми** знайти мінімальне та максимальне значення серед елементів  $\{f_{ij}\}$ .
3. За допомогою **процедури-функції** знайти суму елементів побічної діагоналі масиву  $\{f_{ij}\}$ .
4. За допомогою **процедури-функції** знайти кількість ненульових елементів, розташованих у непарних стовпцях масиву  $\{f_{ij}\}$ .

**Задача 5**

Заданий двовимірний масив числових значень –  $\{w_{ij}\}$ ,  $i=1,2,3$ ;  $j=1,2,3,4$ .

1. У **головній програмі** здійснити введення елементів  $\{w_{ij}\}$ , виклик необхідних процедур і виведення результатів.
2. За допомогою **процедури-підпрограми** підрахувати кількість рядків, в яких нема від'ємних елементів, та кількість стовпців масиву  $\{w_{ij}\}$ , в яких нема додатних елементів.
3. За допомогою **процедури-функції** знайти добуток усіх елементів масиву  $\{w_{ij}\}$ , сума індексів яких дорівнює числу 5.
4. За допомогою **процедури-функції** підрахувати кількість елементів масиву  $\{w_{ij}\}$ , більших за середнє арифметичного елементів цього масиву.

**Задача 6**

Заданий двовимірний масив числових значень –  $\{t_{ij}\}$ ,  $i=1,2,3$ ;  $j=1,2,3$ .

1. У **головній програмі** здійснити введення елементів  $t_{ij}$ , виклик необхідних процедур і виведення результатів.
2. За допомогою **процедури-підпрограми** знайти індекси (номери стовпця і рядка) усіх елементів масиву  $\{t_{ij}\}$ , що більші за середнє арифметичне елементів непарних стовпців масиву  $\{t_{ij}\}$ .
3. За допомогою **процедури-функції** знайти мінімальне значення серед елементів масиву  $\{t_{ij}\}$ , розташованих на головній діагоналі.
4. За допомогою **процедури-функції** знайти суму значень від'ємних елементів масиву  $\{t_{ij}\}$ , сума індексів яких є парним числом.

**Задача 7**

Задані двовимірні масиви числових значень –  $\{a_{ij}\}$ ,  $i=1,2,\dots,m$ ;  
 $j=1,2,\dots,m$  та  $\{b_{ij}\}$ ,  $i=1,2,\dots,k$ ;  $j=1,2,\dots,k$ .

1. У **головній програмі** здійснити введення елементів  $a_{ij}$  та  $b_{ij}$ , виклик необхідних процедур і виведення результатів.
2. За допомогою **процедури-підпрограми** сформувати масив  $\{c_m\}$  з від'ємних елементів, розташованих на головній діагоналі масиву  $\{b_{ij}\}$ .
3. За допомогою **процедури-функції** знайти кількість від'ємних елементів масивів  $\{a_{ij}\}$  та  $\{b_{ij}\}$  окремо.
4. За допомогою **процедури-функції** знайти добуток елементів масиву  $\{a_{ij}\}$ , сума індексів яких дорівнює числу  $(m-1)$ .

**Задача 8**

Заданий двовимірний масив числових значень -  $\{y_{ij}\}$ ,  $i=1,2,3,4$ ;  
 $j=1,2,3,4$ .

1. У **головній програмі** здійснити введення елементів  $y_{ij}$ , виклик необхідних процедур і виведення результатів.
2. За допомогою **процедури-підпрограми** сформувати  $\{x_j\}$ ,  $j=1,2,3,4$  з мінімальних елементів кожного рядка масиву  $\{y_{ij}\}$ .
3. За допомогою **процедури-підпрограми** сформувати масив  $\{u_{ij}\}$ ,  $i=1,2,3,4$ ;  $j=1,2,3,4$  з елементів масиву  $\{y_{ij}\}$ , замінивши всі від'ємні елементи нулями.
4. За допомогою **процедури-функції** підрахувати суму елементів масиву  $\{y_{ij}\}$ , що задовольняють умову  $-12 < y_{ij} < 25$ .

**Задача 9**

Заданий двовимірний масив числових значень –  $\{d_{ij}\}$ ,  $i=1,2,3,4$ ;  
 $j=1,2,3,4$ .

1. У **головній програмі** здійснити введення елементів  $d_{ij}$ , виклик необхідних процедур і виведення результатів.
2. За допомогою **процедури-підпрограми** сформувати масив  $\{c_i\}$ ,  $i=1,2,3,4$ ; елементами якого будуть сума та добуток елементів кожного парного рядка масиву  $\{d_{ij}\}$ .
3. За допомогою **процедури - підпрограми** сформувати масив  $\{y_t\}$ ,  $t=1,2,3,4$ ; елементами якого будуть сума та добуток елементів кожного непарного стовпця масиву  $\{d_{ij}\}$ .
4. За допомогою **процедури - функції** знайти номер рядка, де міститься мінімальний елемент масиву  $\{d_{ij}\}$ .

**Задача 10**

Задані двовимірні масиви числових значень –  $\{a_{ij}\}$  та  $\{b_{ij}\}$ ,  $i=1,2,\dots,n$ ;  $j=1,2,\dots,m$ .

1. У **головній програмі** здійснити введення елементів  $\{a_{ij}\}$  та  $\{b_{ij}\}$ , виклик необхідних процедур і виведення результатів.
2. За допомогою **процедури - підпрограми** сформувати масив  $\{u_{ij}\}$ ,  $i=1,n$ ;  $j=1,m$ , де  $\{u_{ij}\}=\max(a_{ij}, b_{ij})$  ( $u_{11}=\max(a_{11}, b_{11})$   $u_{12}=\max(a_{12}, b_{12}),\dots$ ).
3. За допомогою **процедури-функції** підрахувати кількість елементів парних рядків масиву  $\{u_{ij}\}$ , що задовольняють умову  $f < u_{ij} < d$  ( $f, d$ - довільні числа,  $f < d$ ).
4. За допомогою **процедури-функції** знайти різницю між найбільшим елементом масиву  $\{a_{ij}\}$  і найменшим елементом масиву  $\{b_{ij}\}$ .

**Задача 11**

Заданий двовимірний масив числових значень –  $\{f_{ij}\}$ ,  $i=1,2,\dots,m$ ;  $j=1,2,\dots,m$ .

1. У **головній програмі** здійснити введення елементів  $f_{ij}$ , виклик необхідних процедур і виведення результатів.
2. За допомогою **процедури-підпрограми** знайти мінімальне значення серед елементів другого рядка масиву  $\{f_{ij}\}$  та максимальне значення серед елементів  $m$ -го стовпця масиву  $\{f_{ij}\}$ .
3. За допомогою **процедури-функції** сформувати масив  $\{a_k\}$   $k=1,2,\dots,m$  з елементів побічної діагоналі масиву  $\{f_{ij}\}$ .
4. За допомогою **процедури-функції** знайти перший ненульовий елемент масиву  $\{a_k\}$ .

**Задача 12**

Заданий двовимірний масив числових значень –  $\{w_{ij}\}$ ,  $i=1,2,3$ ;  $j=1,2,3,4$ .

1. У **головній програмі** здійснити введення елементів  $\{w_{ij}\}$ , виклик необхідних процедур і виведення результатів.
2. За допомогою **процедури-підпрограми** знайти мінімальний і максимальний елементи масиву  $\{w_{ij}\}$ .
3. За допомогою **процедури-функції** знайти добуток усіх елементів масиву  $\{f_{ij}\}$ , сума індексів яких дорівнює числу 3.
4. За допомогою **процедури-функції** обчислити середнє арифметичне елементів масиву  $\{f_{ij}\}$ .

**Задача 13**

Заданий двовимірний масив числових значень –  $\{t_{ij}\}$ ,  $i=1,2,3$ ;  $j=1,2,3$ .

1. У **головній програмі** здійснити введення елементів  $t_{ij}$ , виклик необхідних процедур і виведення результатів.
2. За допомогою **процедури-функції** знайти кількість від'ємних елементів, розташованих у непарних рядках  $\{t_{ij}\}$ .
3. За допомогою **процедури-підпрограми** сформувати масив  $\{b_j\}$   $j=1,2,3$  з максимальних елементів кожного стовпця масиву  $\{t_{ij}\}$ .
4. За допомогою **процедури-функції** знайти суму значень від'ємних елементів масиву  $\{t_{ij}\}$ , сума індексів яких є парним числом.

**Задача 14**

Задані двовимірні масиви числових значень –  $\{a_{ij}\}$ ,  $i=1,2,\dots,m$ ;  $j=1,2,\dots,m$  та  $\{b_{ij}\}$ ,  $i=1,2,\dots,m$ ;  $j=1,2,\dots,m$ .

1. У **головній програмі** здійснити введення елементів  $a_{ij}$  та  $b_{ij}$ , виклик необхідних процедур і виведення результатів.
2. За допомогою **процедури - підпрограми** сформувати масив  $\{z_{ij}\}$ ,  $i=1,m$ ;  $j=1,m$ , де  $\{z_{ij}=\min(a_{ij}, b_{ij})\}$  ( $z_{11}=\max(a_{11}, b_{11})$   $z_{12}=\max(a_{12}, b_{12}),\dots$ ).
3. За допомогою **процедури-підпрограми** знайти кількість елементів парних рядків масивів  $\{a_{ij}\}$  та  $\{b_{ij}\}$  окремо.
4. За допомогою **процедури-функції** знайти добуток елементів головної діагоналі масиву  $\{a_{ij}\}$ .

**Задача 15**

Задані двовимірні масиви числових значень –  $\{a_{ij}\}$  та  $\{b_{ij}\}$ ,  $i=1,2,\dots,n$ ;  $j=1,2,\dots,m$ .

1. У **головній програмі** здійснити введення елементів  $a_{ij}$  та  $b_{ij}$ , виклик необхідних процедур і виведення результатів.
2. За допомогою **процедури - підпрограми** знайти, в якому з масивів більше додатних елементів, та найменший елемент масиву  $b_{ij}$ .
3. За допомогою **процедури-функції** обчислити середнє арифметичне елементів масиву  $\{b_{ij}\}$ .
4. За допомогою **процедури-функції** знайти різницю між найбільшим і найменшим значеннями масиву  $\{a_{ij}\}$ .

Продовження додатка 2

*Варіанти додаткових індивідуальних завдань для заочної форми навчання*

Варіант завдання (2 останні цифри шифру залікової книжки)				Номер задачі				
01	31	61	91	1	2	3	4	1
02	32	62	92	2	3	4	1	2
03	33	63	93	3	4	1	2	3
04	34	64	94	4	1	2	3	4
05	35	65	95	1	1	1	1	1
06	36	66	96	1	3	4	1	5
07	37	67	97	3	2	1	2	6
08	38	68	98	4	3	2	3	7
09	39	69	99	1	3	3	4	8
10	40	70	00	2	2	2	2	2
11	41	71		3	1	1	2	1
12	42	72		4	1	1	3	2
13	43	73		1	2	4	4	3
14	44	74		2	3	3	1	4
15	45	75		3	3	3	3	3
16	46	76		4	2	2	3	5
17	47	77		3	2	3	4	6
18	48	78		2	3	4	1	7
19	49	79		3	4	1	2	8
20	50	80		4	4	4	4	4
21	51	81		1	2	1	4	4
22	52	82		3	2	1	2	3
23	53	83		2	4	2	1	2
24	54	84		4	3	4	3	1
25	55	85		3	3	2	2	6
26	56	86		1	3	2	4	5
27	57	87		2	4	4	1	7
28	58	88		4	3	3	3	6
29	59	89		3	4	1	2	8
30	60	90		2	4	3	1	2

*При розв'язанні задач описати сутність методів, що використовуються, навести схеми алгоритмів та описати призначення блоків у них, записати Basic-програми моделювання.*

**Варіанти індивідуальних завдань до задачі № 1**

Для моделювання статичної системи, поведінка якої описується нелінійним рівнянням:

1) ізолювати корені в інтервалі  $x_0, x_k$  з кроком  $h$  методом послідовного перебору;

2) уточнити знайдені корені рівняння з заданою точністю  $\xi$ :  
— варіанти, що закінчуються 0,2,4,6,8, — методом ітерацій;  
— варіанти, що закінчуються 1,3,5,7,9, — методом ділення інтервалу навпіл.

1. При розрахунку симетричних коливань рухомої одиниці застосовується частотне рівняння

$$p^4 - (v_1^2 + v_2^2)p^2 + v_1^2(v_2^2 - v_3^2) = 0,$$

де  $v_i$  — задані парціальні частоти системи.

Розв'язати рівняння відносно  $p$ .

2. Для обчислення значень частот власних гармонічних коливань кузова при відсутності нерівностей рельсів та виляння візка у колії використовується рівняння

$$p^4 - (a_1 + a_3)p^2 + a_1a_3 - a_2a_4 = 0,$$

де  $a_1, a_2, a_3, a_4$  — задані змінні.

Розв'язати рівняння відносно  $p$ .

3. Під час руху з постійною швидкістю візка вагона на пологих кривих ділянках положення полюса повороту (відстань  $\alpha$  між ним та центром візка) задається рівнянням  $2F(2I_b \cos \beta + S(\sin \beta + \sin \alpha)) - Hb = 0$ , де  $F$  — сила сухого тертя;  $H$  — поперечна сила;  $2I_b$  — база візка;  $2S$  — ширина колії;

$$\cos \beta = \frac{I_b - a}{\sqrt{S^2 + (I_b - a)^2}}, \quad \sin \beta = \frac{S}{\sqrt{S^2 + (I_b - a)^2}}, \quad \sin \alpha = \frac{S}{\sqrt{S^2 + (I_b + a)^2}}.$$

Розв'язати рівняння відносно  $a$ .

4. Для знаходження значень частот власних коливань обресорених частин рухомої одиниці використовується рівняння

$$\lambda^8 + k_1\lambda^6 + k_2\lambda^4 + k_3\lambda^2 + k_4 = 0,$$

де  $k_1, k_2, k_3, k_4$  — задані змінні. Розв'язати рівняння відносно  $\lambda$ .



**Варіанти індивідуальних завдань до задачі № 2**

Для моделювання динамічної системи, поведінка якої описується звичайними диференціальними рівняннями, знайти відповідне рішення на відрізку  $[t_0, t_k]$  з кроком  $h$ :

- варіанти, що закінчуються 0,2,4,6,8, — методом Ейлера;
- варіанти, що закінчуються 1,3,5,7,9, — методом Рунге-Кутта.

1. Імовірність безвідмовної роботи вузлів рухомої одиниці під час притирання, в залежності від призначення, якості, режимів експлуатації тощо описується диференціальним рівнянням

$$\frac{dP(t)}{dt} = -(\lambda + \lambda_1 t^n) dt,$$

де  $\lambda$  — інтенсивність відмов за період нормальної експлуатації;  $\lambda_1, n$  — параметри інтенсивності відмов.

Розв'язати рівняння відносно  $P$ ,  $P(a) = P_0$ .

2. Диференціальне рівняння руху маси локомотива

$$\frac{dV}{dt} + \frac{\rho\lambda}{M} + \frac{T_k}{M} = 0,$$

де  $M$  — маса локомотива;  $\lambda$  — розповсюдження сили натягу;  $T_k$  — сила тяги.

Розв'язати рівняння відносно  $V$ ,  $V(t_0) = V_0$ .

3. Імовірність відмови вузлів рухомої одиниці можна записати диференціальним рівнянням

$$\frac{dQ(t)}{dt} = f(t),$$

де  $f(t)$  — частота відмов (кількість відмов за одиницю часу, віднесена до початкової кількості вузлів).

Розв'язати рівняння відносно  $Q$ , якщо  $f(t) = \frac{t^3 + 1}{t^3}$ ,  $Q(t_0) = Q_0$ .

4. Значення внутрішньої температури  $T$  стінки в будь-якій точці можна знайти з розв'язання диференціального рівняння

$$\frac{dT}{dx} = -\frac{Q}{\lambda(1+BT)}.$$

Розв'язати рівняння відносно  $T$ ,  $T(t_0) = T_1$ ,  $B$ ,  $Q$ ,  $\lambda$  — задані змінні.

**Варіанти індивідуальних завдань до задачі № 3**

Для моделювання системи, поведінка якої описується визначеним інтегралом, знайти відповідне рішення:

- варіанти, що закінчуються 0,2,4,8, — методом прямокутників;
- варіанти, що закінчуються 1,5,7, — методом трапецій;
- варіанти, що закінчуються 3,6,9, — методом Симпсона.

1. Частота вільних коливань кузова у випадку несиметричної упругої характеристики ресор визначається за формулою

$$p^2 = \frac{5}{2m_k A_0^5} \int_{-A_0}^{A_0} F(z - \delta) z^3 dz,$$

де  $m_k$  — маса кузова;  $A_0$  — напіврозмах коливань;  $\delta$  — зміщення центра коливань від початку координат;  $F(z) = cz + F_0$  — характеристика відновлюючої сили. Обчислити визначений інтеграл та знайти частоту коливань. Крок інтегрування  $h$ .

2. При розрахунку нерівності рельсової колії застосовується спектральний метод, у якому для реалізації випадкової функції нерівностей необхідно отримати автокореляційну функцію, що має вигляд

$$k(\tau) = \lim_{T \rightarrow \infty} \frac{1}{T} I,$$

де  $I = \int_0^T \eta(t)\eta(t+\tau)dt$ ,  $\eta(t)$  — реалізація випадкової функції;  $\tau$  — зсув випадкового процесу за часом;  $T$  — тривалість реалізації випадкової функції. Обчислити визначений інтеграл  $I$ , якщо функція  $\eta(x) = x + 1$ . Крок інтегрування  $h$ .

3. Середня трудомісткість ремонту двигуна даного типу, визначається за формулою

$$W = \int_0^{W_{\max}} W p(w) dw,$$

де  $W$  — випадкове значення трудомісткості даного типу;  $P(w) = W + 1$  — щільність розподілення трудомісткості даного типу. Обчислити даний визначений інтеграл з кроком  $h$ .

4. Сумарна витрата пального через трубу радіуса  $R$  визначається виразом

$$G = 2K \int_0^R \sqrt{ar^2 + br + c} dr,$$

для випадку апроксимації графіка швидкості — квадратичною параболою. Обчислити визначений інтеграл. Крок інтегрування  $h$  та значення  $R, a, b, c, K$  вважати заданими.

**Варіанти індивідуальних завдань до задачі № 4**

Для моделювання системи, поведінка якої описується заданою функцією, розв'язати задачу оптимізації:

— варіанти, що закінчуються 0,2,4,6,8, — методом перебору;

— варіанти, що закінчуються 1,3,5,7,9, — методом золотого перетину.

1. Візок рівноприскорено рухається уздовж заданого шляху зі швидкістю

$$v(t) = \sqrt{\dot{x}^2(t) + \dot{y}^2(t)},$$

$$\begin{cases} x(t) = \frac{1}{2}(e^t + e^{-t}) \\ y(t) = \frac{1}{2}(e^t - e^{-t}) \end{cases} \text{ (дуга гіперболи), } t \text{ — часовий параметр.}$$

Визначити точку шляху на відрізку  $[a, b]$  з кроком  $h$ , в якій швидкість візка була максимальною,  $\varepsilon = 10^{-4}$ ,  $h = 0.1$ .

2. Візок рівноприскорено рухається уздовж заданого шляху з прискоренням

$$a(t) = \sqrt{\ddot{x}^2(t) + \ddot{y}^2(t)},$$

$$\begin{cases} x(t) = \cos t \\ y(t) = \sin t \end{cases} \text{ (дуга кола), } t \text{ — часовий параметр.}$$

Визначити точку шляху на відрізку  $[a, b]$  з кроком  $h$ , у якій прискорення візка було мінімальним,  $\varepsilon = 10^{-4}$ ,  $h = 0.1$ .

3. Кутове прискорення (відцентрове) обумовлює силу, що сприяє перекиданню візка (при досягненні критичних показників).

$$F(t) = m\sqrt{\ddot{x}^2(t) + \ddot{y}^2(t)}, \quad m \text{ — маса візка.}$$

$$\begin{cases} x(t) = t \\ y(t) = t^2 \end{cases} \text{ (дуга параболи), } t \text{ — часовий параметр}$$

Знайти параметр, при якому досягається максимум функції  $F$  на відрізку  $[a, b]$  з кроком  $h$ , тобто може виникнути перекидання,  $\varepsilon = 10^{-4}$ ,  $h = 0.1$ .

4. Залежність амплітуди електричного струму від частоти в трансформаторі (трансформатор розглядається як простий коливальний контур) задається співвідношенням  $s = \frac{r}{\sqrt{R^2 + \left(L\omega - \frac{1}{C\omega}\right)^2}}$ .

Відомі  $R$  — зовнішній опір,  $L$  — індуктивність,  $C$  — ємність. Для якої частоти  $\omega$  даний електричний контур досягає резонансного стану (знайти максимум амплітуди  $s$  для  $\omega$ , що змінюється в діапазоні  $[\omega_1, \omega_2]$  із кроком  $h$ ),  $\varepsilon = 10^{-4}$ ,  $h = 0.1$ .

**Варіанти індивідуальних завдань до задачі № 5**

Для забезпечення надійності електронної апаратури необхідно здійснити такі заходи:

- підвищити надійність комплектуючих елементів (вид 1);
- ввести інформаційну, алгоритмічну й апаратну надмірності (вид 2).

Необхідно визначити послідовність реалізації заходів, при якій задана надійність буде досягнута з мінімальними витратами. Відома вартість шляху між сусідніми точками, вона задана як елементи відповідних матриць.

Вартості горизонтальних кроків з вершин з відповідними індексами — елементи матриці  $A$ ; вартості вертикальних кроків з вершин з відповідними індексами — елементи матриці  $B$ .

Необхідно:

- накреслити мережу згідно з даним варіантом (табл. Д.2.1);
- виконати розрахунок визначення оптимального шляху;
- відмітити на мережі знайдений оптимальний шлях.

Таблиця Д.2.1

Варіант	Вартість шляху між сусідніми точками	
1	2	3
1	$A_{0,0}=3, A_{0,1}=4 A_{0,2}=9 A_{0,3}=8$ $A_{1,0}=4, A_{1,1}=9 A_{1,2}=4 A_{1,3}=7$ $A_{2,0}=8, A_{2,1}=5 A_{2,2}=10 A_{2,3}=4$ $A_{3,0}=4, A_{3,1}=7 A_{3,2}=10 A_{3,3}=5$	$B_{0,0}=4, B_{0,1}=6 B_{0,2}=9 B_{0,3}=5 B_{0,4}=4$ $B_{1,0}=8, B_{1,1}=6 B_{1,2}=5 B_{1,3}=9 B_{1,4}=10$ $B_{2,0}=5, B_{2,1}=3 B_{2,2}=2 B_{2,3}=10 B_{2,4}=6$
2	$A_{0,0}=2, A_{0,1}=7 A_{0,2}=8 A_{0,3}=7$ $A_{1,0}=3, A_{1,1}=8 A_{1,2}=3 A_{1,3}=6$ $A_{2,0}=7, A_{2,1}=4 A_{2,2}=9 A_{2,3}=3$ $A_{3,0}=3, A_{3,1}=6 A_{3,2}=9 A_{3,3}=4$	$B_{0,0}=5, B_{0,1}=7 B_{0,2}=10 B_{0,3}=6 B_{0,4}=5$ $B_{1,0}=9, B_{1,1}=7 B_{1,2}=6 B_{1,3}=10 B_{1,4}=11$ $B_{2,0}=6, B_{2,1}=4 B_{2,2}=3 B_{2,3}=11 B_{2,4}=7$

Продовження додатка 2

Продовження табл. Д.2.1

1	2	3
3	$A_{0,0}=4, A_{0,1}=9 A_{0,2}=10 A_{0,3}=9$ $A_{1,0}=5, A_{1,1}=10 A_{1,2}=5 A_{1,3}=8$ $A_{2,0}=9, A_{2,1}=6 A_{2,2}=11 A_{2,3}=5$ $A_{3,0}=5, A_{3,1}=7 A_{3,2}=11 A_{3,3}=6$	$B_{0,0}=6, B_{0,1}=8 B_{0,2}=11 B_{0,3}=7 B_{0,4}=6$ $B_{1,0}=10, B_{1,1}=8 B_{1,2}=7 B_{1,3}=11 B_{1,4}=12$ $B_{2,0}=7, B_{2,1}=5 B_{2,2}=4 B_{2,3}=12 B_{2,4}=8$
4	$A_{0,0}=5, A_{0,1}=10 A_{0,2}=11 A_{0,3}=10$ $A_{1,0}=6, A_{1,1}=11 A_{1,2}=6 A_{1,3}=9$ $A_{2,0}=10, A_{2,1}=7 A_{2,2}=12 A_{2,3}=6$ $A_{3,0}=6, A_{3,1}=9 A_{3,2}=12 A_{3,3}=7$	$B_{0,0}=6, B_{0,1}=8 B_{0,2}=11 B_{0,3}=7 B_{0,4}=6$ $B_{1,0}=10, B_{1,1}=8 B_{1,2}=7 B_{1,3}=11 B_{1,4}=12$ $B_{2,0}=7, B_{2,1}=5 B_{2,2}=4 B_{2,3}=12 B_{2,4}=8$
5	$A_{0,0}=4, A_{0,1}=9 A_{0,2}=10 A_{0,3}=9$ $A_{1,0}=5, A_{1,1}=10 A_{1,2}=5 A_{1,3}=8$ $A_{2,0}=9, A_{2,1}=6 A_{2,2}=11 A_{2,3}=5$ $A_{3,0}=5, A_{3,1}=7 A_{3,2}=11 A_{3,3}=6$	$B_{0,0}=7, B_{0,1}=9 B_{0,2}=12 B_{0,3}=8 B_{0,4}=7$ $B_{1,0}=11, B_{1,1}=9 B_{1,2}=8 B_{1,3}=12 B_{1,4}=13$ $B_{2,0}=8, B_{2,1}=6 B_{2,2}=5 B_{2,3}=13 B_{2,4}=9$
6	$A_{0,0}=5, A_{0,1}=10 A_{0,2}=11 A_{0,3}=10$ $A_{1,0}=6, A_{1,1}=11 A_{1,2}=6 A_{1,3}=9$ $A_{2,0}=10, A_{2,1}=7 A_{2,2}=12 A_{2,3}=6$ $A_{3,0}=6, A_{3,1}=9 A_{3,2}=12 A_{3,3}=7$	$B_{0,0}=7, B_{0,1}=9 B_{0,2}=12 B_{0,3}=8 B_{0,4}=7$ $B_{1,0}=11, B_{1,1}=9 B_{1,2}=8 B_{1,3}=12 B_{1,4}=13$ $B_{2,0}=8, B_{2,1}=6 B_{2,2}=5 B_{2,3}=13 B_{2,4}=9$
7	$A_{0,0}=6 A_{0,1}=11 A_{0,2}=12 A_{0,3}=11$ $A_{1,0}=7, A_{1,1}=12 A_{1,2}=7 A_{1,3}=10$ $A_{2,0}=11, A_{2,1}=8 A_{2,2}=13 A_{2,3}=7$ $A_{3,0}=7, A_{3,1}=10 A_{3,2}=13 A_{3,3}=8$	$B_{0,0}=4, B_{0,1}=6 B_{0,2}=9 B_{0,3}=5 B_{0,4}=4$ $B_{1,0}=8, B_{1,1}=6 B_{1,2}=5 B_{1,3}=9 B_{1,4}=10$ $B_{2,0}=5, B_{2,1}=3 B_{2,2}=2 B_{2,3}=10 B_{2,4}=6$
8	$A_{0,0}=2, A_{0,1}=7 A_{0,2}=8 A_{0,3}=7$ $A_{1,0}=3, A_{1,1}=8 A_{1,2}=3 A_{1,3}=6$ $A_{2,0}=7, A_{2,1}=4 A_{2,2}=9 A_{2,3}=3$ $A_{3,0}=3, A_{3,1}=6 A_{3,2}=9 A_{3,3}=4$	$B_{0,0}=7, B_{0,1}=9 B_{0,2}=12 B_{0,3}=8 B_{0,4}=7$ $B_{1,0}=11, B_{1,1}=9 B_{1,2}=8 B_{1,3}=12 B_{1,4}=13$ $B_{2,0}=8, B_{2,1}=6 B_{2,2}=5 B_{2,3}=13 B_{2,4}=9$

## Приклади виконання індивідуальних завдань

### Задача 1

При розрахунку симетричних коливань рухомої одиниці застосовується частотне рівняння відносно  $p$

$$p^4 - (v_1^2 + v_2^2)p^2 + v_1^2(v_2^2 - v_3^2) = 0,$$

де  $v_i$  — задані парціальні частоти системи.

1) Ізолювати корені в інтервалі  $x_0, x_k$  з кроком  $h$  методом послідовного перебору.

Опис алгоритму (рис. Д.2.7):

**1,10** — початок, кінець

**2** — введення початкових значень  $x_0, x_k, h$

**3** — введення значень  $v_1, v_2, v_3$

**4** — надання початкового значення аргументу  $a = x_0$

**5** — збільшення аргументу на крок  $b = a + h$

**6** — перевірка умови  $f(a) f(b) \leq 0$

**7** — введення інтервалу, де існує корінь

**8** — збільшення аргументу на крок  $a = a + h$

**9** — перевірка умови  $a \leq x_k$

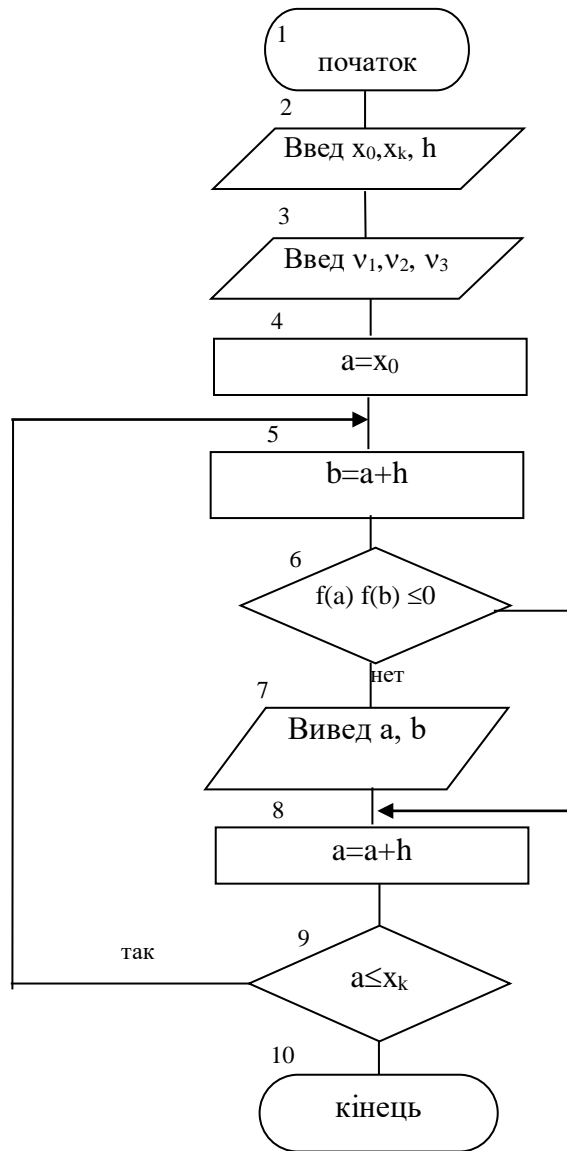


Рис. Д.2.7. Метод послідовного перебору

При дослідженні детермінованих моделей у багатьох випадках виникає необхідність розв'язання нелінійних рівнянь з одним невідомим вигляду

$$f(x)=0.$$

Нелінійні рівняння з одним невідомим поділяються на алгебраїчні і трансцендентні. Рівняння вигляду  $f(x)=0$  називається алгебраїчним, якщо функція є алгебраїчною функцією.

Якщо функція  $f(x)$  не є алгебраїчною, то рівняння називається трансцендентним.

Число  $c \in [a, b]$  у якому функція  $f(x)$  дорівнює нулю, називається коренем рівняння.

У деяких випадках розв'язання трансцендентних рівнянь можна звести до розв'язання алгебраїчних рівнянь, але оскільки більшість нелінійних рівнянь не розв'язується шляхом аналітичних перетворень (тобто прямими, точними методами), на практиці їх розв'язують чисельними методами. Розв'язати таке рівняння означає установити, чи має воно корені, скільки коренів, і знайти значення коренів із заданою точністю.

Задача чисельного знаходження дійсних і комплексних коренів нелінійного рівняння звичайно складається з двох етапів: відділення коренів та уточнення коренів.

### **Ізоляція або відділення коренів**

Відділення кореня (ізоляція) — визначення якнайменших інтервалів, на яких міститься один корінь.

Якщо безперервна та монотонна функція на кінцях інтервалу  $[a, b]$  має значення різних знаків, то на цьому інтервалі міститься хоча б один корінь. Розіб'ємо інтервал існування функції  $[x_n, x_k]$  на рівні відрізки довжиною  $\Delta x$  та будемо обчислювати значення функції на кінцях цих відрізків ( $f(a)$  та  $f(b)$  відповідно). Умовою наявності кореня буде  $f(a)*f(b) \leq 0$ .



*Текст програми:*

```

REM ІЗОЛЯЦІЯ КОРЕНІВ НЕЛІНІЙНОГО РІВНЯННЯ ПРИ
МОДЕЛЮВАННІ СТАТИЧНОЇ СИСТЕМИ
CLS
INPUT "ВВЕДІТЬ ЗНАЧЕННЯ ІНТЕРВАЛУ X0,XK "; X0,XK
INPUT "ВВЕДІТЬ ЗНАЧЕННЯ КРОКУ Н "; Н
INPUT "ВВЕДІТЬ ЗНАЧЕННЯ V1,V2, V3"; V1,V2, V3
DEF FNF (X)=X^4-(V1^2+V2^2)*X^2+V1^2*(V2^2-V3^2)
A= X0
M1: B = A + H
IF FNF(A) * FNF(B) <= 0 THEN PRINT " КОРІНЬ НА
ВІДРІЗКУ: "; A; " - "; B
A = A + H
IF A < XK THEN M1
END

```

2) *Уточнити знайдені корені рівняння з заданою точністю  $\xi$  методом ділення інтервалу навпіл.*

### ***Уточнення коренів методом ділення інтервалу навпіл***

Для уточнення кореня, що належить відрізку  $[a,b]$ , ділимо відрізок навпіл, тобто вибираємо початкове наближення, що дорівнює  $t=(a+b)/2$ . Якщо  $f(t)=0$ , то  $t$  є коренем рівняння. Якщо  $f(t) \neq 0$ , то вибираємо той з відрізків  $[a,t]$  або  $[t,b]$ , на кінцях якого функція  $f(x)$  має протилежні знаки (рис. Д.2.8).

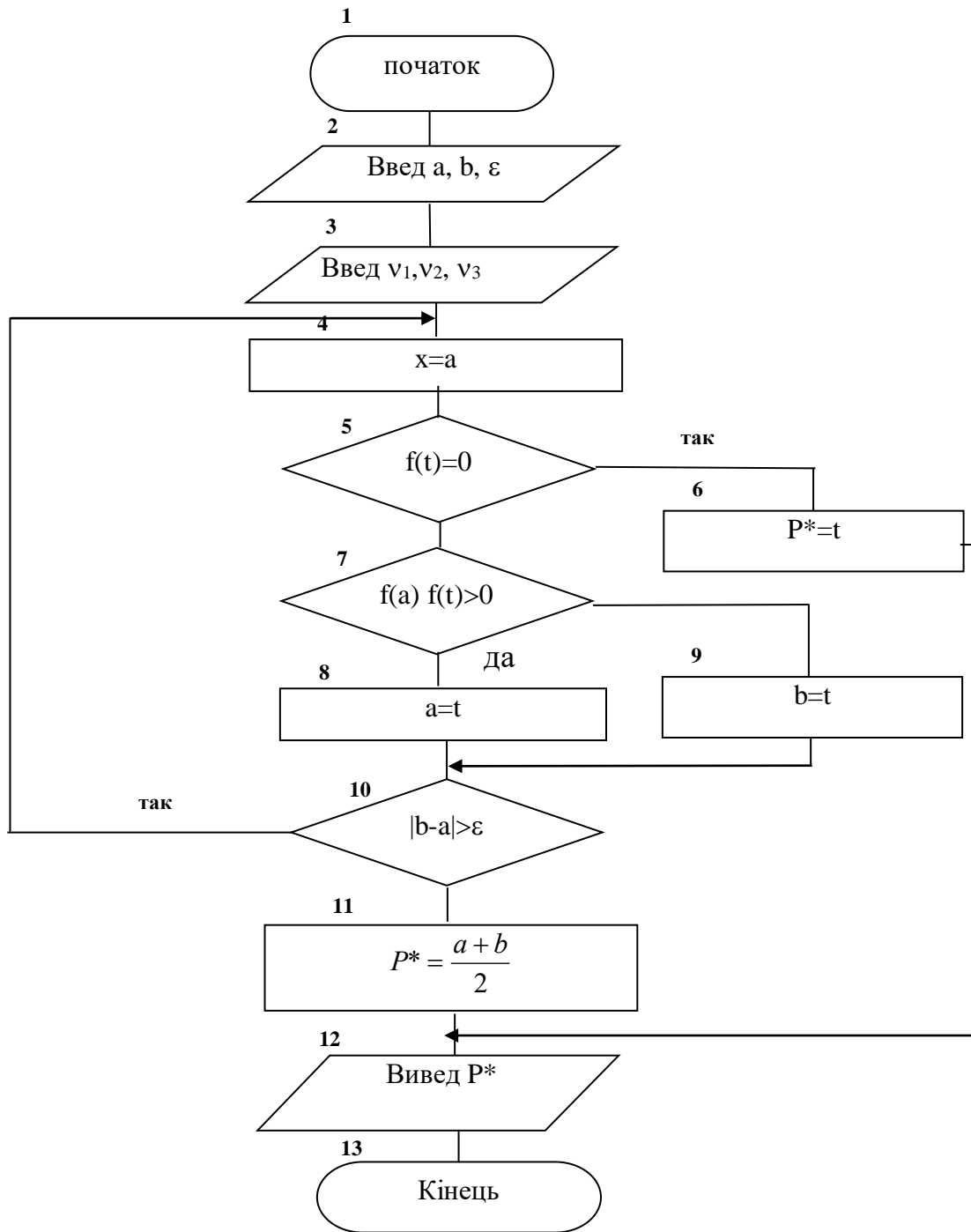


Рис. Д.2.8. Метод послідовного перебору

## Продовження додатка 2

Отриманий відрізок знову ділимо навпіл, аналізуємо виконання умови  $|b-a| > E$ , де  $E$  — точність знаходження кореня, і якщо вона досягнута, закінчуємо алгоритм. У протилежному випадку повторюємо наведену вище послідовність визначення нового відрізка, що містить корінь рівняння.

*Опис алгоритму:*

- 1,13 – початок, кінець
- 2,3 – введення початкових значень  $a, b, E, v_1, v_2, v_3$
- 4 – розрахунок середини інтервалу  $t = (a+b)/2$
- 5 – перевірка умови  $f(t) = 0$
- 6 – розрахунок кореня в середині інтервалу  $P^* = t$
- 7 – перевірка умови  $f(a) * f(t) > 0$
- 8 –  $a = t$
- 9 –  $b = t$
- 10 – перевірка умови  $|b-a| > e$
- 11 – розрахунок кореня в середині інтервалу  $P^* = (a+b)/2$
- 12 – виведення кореня  $P^*$

*Текст програми:*

```
REM УТОЧНЕННЯ КОРЕНІВ МЕТОДОМ ДІЛЕННЯ
ІНТЕРВАЛУ НАВПІЛ
INPUT "ПОЧАТКОВИЙ ІНТЕРВАЛ А,В ==>"; А, В
INPUT "ТОЧНІСТЬ ОБЧИСЛЕННЯ Е ==>"; Е
INPUT "ВВЕДІТЬ ЗНАЧЕННЯ V1,V2, V3"; V1,V2, V3
DEF FNF (X)=X^4-(V1^2+V2^2)*X^2+V1^2*(V2^2-V3^2)
CLS
M2: T = (A + B) / 2
IF FNF(T) = 0 THEN X=T: GOTO M1
IF FNF(A) * FNF(T) < 0 THEN B = T ELSE A = T
IF ABS(B - A) > E THEN GOTO M2
X=(A + B) / 2
M1: PRINT "КОРІНЬ ="; X
```

**Задача 2**

Імовірність відмови вузлів рухомої одиниці можна записати диференціальним рівнянням

$$\frac{dQ(t)}{dt} = f(t),$$

де  $f(t)$  — частота відмов (кількість відмов за одиницю часу віднесена до початкової кількості вузлів). Розв'язати рівняння методом Ейлера відносно  $Q$  на відрізку  $[t_0, t_k]$  з кроком  $h$ ,

якщо  $f(t) = \frac{t^3 + 1}{t^3}$ ,  $Q(t_0) = Q_0$ .

Нехай поведінка моделі описується диференціальним рівнянням

$$y' = f(x, y).$$

Розв'язати таке рівняння означає знайти таку функцію  $y = y(x)$ , котра задовольняє початкову умову  $y(x_0) = y_0$  та визначається рівнянням

$$y = \int f(x, y(x)) dx + c,$$

де  $c$  — довільна константа.

Характерним для усіх чисельних методів є те, що при розв'язанні диференціальних рівнянь потрібно знайти не первісну, а лише значення самої функції  $y_0, y_1, \dots, y_k$  на певному інтервалі  $[x_0, x_k]$ . Кожне наступне значення функції  $y$  обчислюється через попереднє за допомогою рекурентних співвідношень (рис. Д.2.9).

**Метод Ейлера**

У його основі лежить розкладення функції в ряд Тейлора біля точки  $x_0$

$$y(x_0 + h) = y(x_0) + hy'(x_0) + \frac{1}{2!} h^2 y''(x_0) + \frac{1}{3!} h^3 y'''(x_0) + \dots$$

Оскільки значення  $h$  мале, у розрахунковій формулі залишають перші два елементи ряду

$$y(x_0 + h) = y(x_0) + hy'(x_0).$$

Цю формулу, враховуючи вхідне рівняння, можна записати у вигляді

$$y(x_{n+1}) = y(x_n) + hf(x_n, y_n), \quad n = 0, 1, 2, \dots$$

Похибка методу в цьому випадку має величину порядку  $h^2$ .

Опис алгоритму:

- 1,11 – початок, кінець
- 2,3,4 – введення значень  $t_0, t_k, Q_0, h, k$
- 5 – завдання початкових значень  $x=t_0, y=Q_0$
- 6 – завдання початкового значення  $i=1$
- 7 – розраховування  $y = y + h * f(x,y)$  – наступне  $Q$
- 8 – зміна  $x=x+h$  та  $i=i+1$  із заданим кроком
- 9 – перевірка умови  $i \leq k$
- 10 – виведення  $x, y$
- 11 – перевірка умови  $t \leq t_k$

Текст програми:

```

REM ЧИСЕЛЬНЕ РОЗВ'ЯЗАННЯ ДИФЕРЕНЦІЙНОГО
РІВНЯННЯ МЕТОДОМ ЕЙЛЕРА
CLS
INPUT "ПОЧАТОК ІНТЕРВАЛУ T0 ="; T0
INPUT " ПОЧАТКОВЕ ЗНАЧЕННЯ ТК ="; TK
INPUT " H,Q0,K ="; H,Q0,K
DEF FNF (X,Y) = (X^3+1)/X^3
X=T0 : Y=Q0
4 I=1
5 Y=Y+H* FNF (X,Y)
X=X+H
I=I+1
IF I<=K THEN 5
PRINT " X="; X, " Y="; Y
IF X<= TK THEN 4
END

```

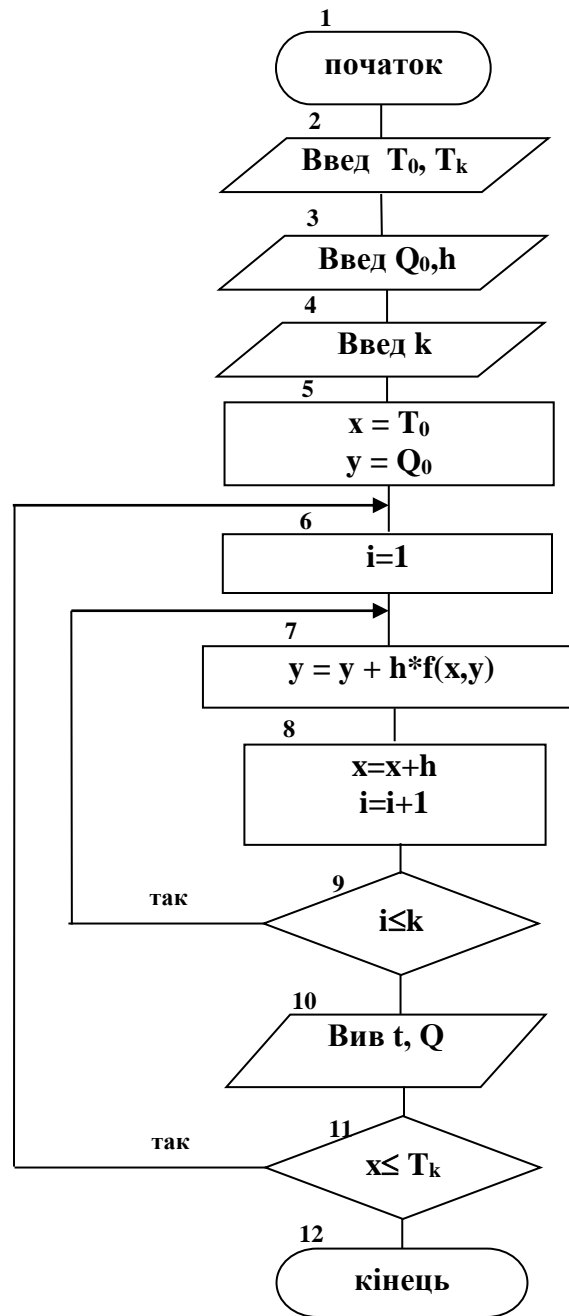


Рис. Д.2.9

**Задача 3**

При розрахунку нерівності рельсової колії застосовується спектральний метод, у якому для реалізації випадкової функції нерівностей необхідно отримати автокореляційну функцію, що має вигляд

$$k(\tau) = \lim_{T \rightarrow \infty} \frac{1}{T} I;$$

де  $I = \int_0^T \eta(t)\eta(t+\tau)dt$ ;  $\eta(t)$  — реалізація випадкової функції,  $\tau$  — зсув випадкового процесу за часом;  $T$  — тривалість реалізації випадкової функції. Обчислити визначений інтеграл  $I$  методом прямокутників, якщо функція  $\eta(x) = x + 1$ . Крок інтегрування  $h$ .

В інженерних розрахунках та при моделюванні часто виникає необхідність обчислення значень визначеного інтегралу вигляду

$$D = \int_a^b f(x)dx .$$

Формули для приблизного обчислення отримують заміною підінтегральної функції  $f(x)$  інтерполяційним поліномом. У цьому випадку визначений інтеграл може бути записаний у вигляді

$$D = \int_a^b f(x)dx = \sum_{k=1}^m A_k f(x_k) + R,$$

де  $x_k$  — вузли інтерполяції;  $A_k$  — коефіцієнти, що залежать від суті формули та вибору вузлів;  $R$  — похибка обчислювальної процедури.

У залежності від того, як інтерполюється  $f(x)$ , отримують різні формули для обчислення визначених інтегралів. Якщо використовують поліноми нульового порядку, маємо формулу прямокутників, першого порядку — формулу трапецій, другого порядку — формулу Симпсона.

Після заміни функції поліномом значення визначеного інтеграла обчислюється як сума площ елементарних фігур, що обмежені віссю  $x$  та кривою  $y=f(x)$ .

При використанні методу прямокутників приблизне значення інтеграла визначається за формулою

$$D = \int_a^b f(x)dx \approx \frac{b-a}{n} (y_1 + y_2 + \dots + y_n) = \frac{b-a}{n} \sum_{i=1}^n y_i,$$

де  $y_i$  — значення  $f(x)$  на початку кожного  $i$ -го інтервалу;  $n$  — кількість відрізків, на які поділений діапазон інтегрування;  $a$  — нижня межа інтегрування;  $b$  — верхня межа інтегрування.

У цьому методі крива підінтегральної функції замінюється ламаною лінією, відрізки якої паралельні осі абсцис з наступним визначенням суми площин елементарних прямокутників (рис. Д.2.10).

*Опис алгоритму:*

- 1,11 — початок, кінець
- 2,3 — введення значень  $T, h, \tau$
- 4,5 — надання сумі та аргументу початкових значень  $s=0$  та  $x=0$
- 6 — обчислення значення  $f(x)$
- 7 — накопичення суми та збільшення аргументу на крок
- 8 — перевірка умови  $X < T$
- 9 — обчислення значення інтеграла
- 10 — виведення значення інтеграла

*Текст програми:*

```
REM ЧИСЕЛЬНЕ ОБЧИСЛЕННЯ ВИЗНАЧЕНОГО ІНТЕГРАЛА
REM МЕТОДОМ ПРЯМОКУТНИКІВ
DEF FNF (X) = (X+1)*(X+1+TA)
CLS
INPUT " H ="; H
INPUT " T ="; T
INPUT "TA=";TA
S=0
X=0
M1: S= S+FNF(X)
X=X+H
IF X<T THEN M1
I=H*S
PRINT "I="; I
END
```



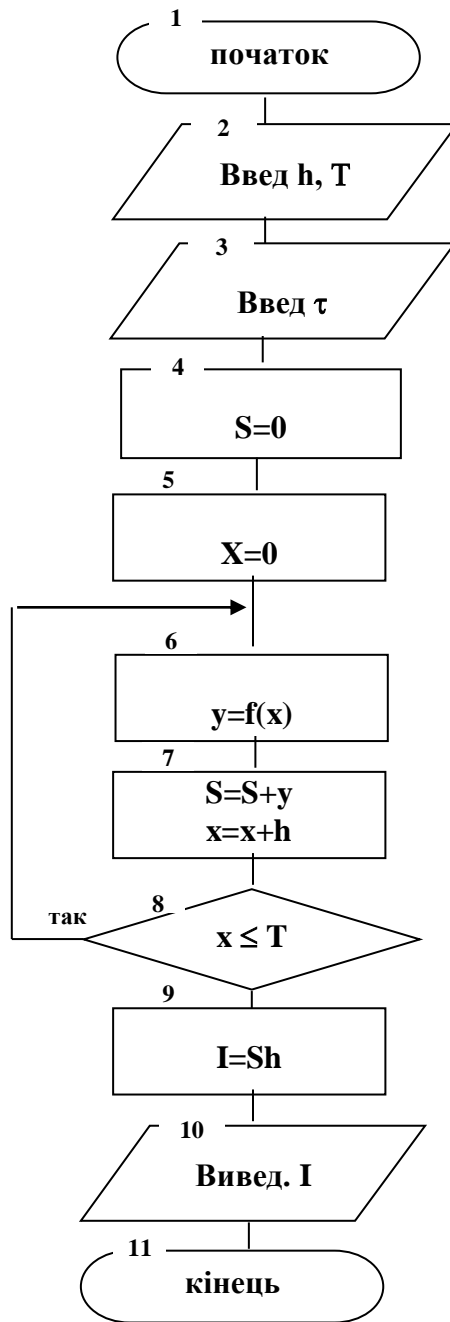


Рис. Д.2.10

**Задача 4**

*Залежність амплітуди електричного струму від частоти в трансформаторі (трансформатор розглядається як простий коливальний контур) задається співвідношенням*

$$s = \frac{r}{\sqrt{R^2 + \left(L\omega - \frac{1}{C\omega}\right)^2}}.$$

*Відомі  $R$  — зовнішній опір,  $L$  — індуктивність,  $C$  — ємність. Для якої частоти  $\omega$  даний електричний контур досягає резонансного стану (знайти максимум амплітуди  $s$  для  $\omega$ , що змінюється в діапазоні  $[\omega_1, \omega_2]$  із кроком  $h$ ),  $\varepsilon = 10^{-4}$ ,  $h = 0.1$*

*Для моделювання системи, поведінка якої описується заданою функцією, розв'язати оптимізаційну задачу методом перебору (рис. Д.2.11).*

При будь-якому способі завдання функції для обчислення екстремуму необхідно мати визначену сукупність значень  $\mathbf{x}$  та  $\mathbf{Y}=\mathbf{F}(\mathbf{x})$  на заданому інтервалі дослідження.

Незалежно від типу розв'язуваної задачі для пошуку екстремуму можна користуватися тим самим алгоритмом, тобто задачу мінімізації можна легко перетворити на задачу максимізації, змінивши знак функції на протилежний.

Метод полягає в знаходженні значень цільової функції  $\mathbf{F}(\mathbf{x}^*)$  у точках, якими ми розбиваємо інтервал  $[\mathbf{x}_0, \mathbf{x}_k]$  на рівні відрізки, довжиною  $\Delta\mathbf{X}$ , порівнянні кожного обчисленого значення зі значенням, найбільшим на даному етапі, і запам'ятовуванням мінімального значення  $\mathbf{MIN}=\mathbf{F}(\mathbf{x}^*)$  і аргументу  $\mathbf{x}^*$ , що забезпечує цей мінімум.

Перед початком циклу з визначення  $\mathbf{MIN}$  значення функції  $\mathbf{Y}=\mathbf{F}(\mathbf{x})$  обчислюємо перше значення функції, що приймається за  $\mathbf{MIN}$ , тобто  $\mathbf{YMIN}=\mathbf{Y}_1$ .

Для обчислення наступного значення функції змінюємо значення параметра циклу  $i$ , якщо нове значення параметра припустиме, обчислюємо відповідне значення функції. Отримане значення функції  $\mathbf{Y}_2$  порівнюємо з  $\mathbf{YMIN}$ .

Якщо  $\mathbf{Y}_2 < \mathbf{YMIN}$ , то  $\mathbf{YMIN}$  приймає значення  $\mathbf{Y}_2$  ( $\mathbf{YMIN}=\mathbf{Y}_2$ ), інакше  $\mathbf{YMIN}$  зберігає своє значення, і після зміни параметра циклу процес повторюється.

Вихід з циклу здійснюється по досягненні параметром верхньої границі інтервалу. Слід зазначити, що при розв'язанні таких задач йдеться не про  $\mathbf{MIN}$  або  $\mathbf{MAX}$  функції, а про  $\mathbf{MIN}$  або  $\mathbf{MAX}$  серед обчислених значень цієї функції. Це пояснюється тим, що комп'ютер

## Продовження додатка 2

обчислює дискретні значення функції при відповідних дискретних значеннях параметра, а реальний **MIN** або **MAX** може бути між ними. Підвищити точність визначення **MIN** або **MAX** можна за рахунок зменшення кроку зміни параметра циклу.

*Опис алгоритму:*

- 1,11** – початок, кінець
- 2** – введення значень  $R, L, C, W1, W2$
- 3** – надання початкового значення  $w=w1$
- 4** – обчислення значення  $f(w)$  ( $s$ )
- 5** – надання початкових значень (значень на початку інтервалу) –  $smax$  та  $wmax$
- 6** – організація циклу перебору за  $w$
- 7** – обчислення значення  $f(w)$  ( $s$ )
- 8** – перевірка умови  $S > Smax$
- 9** – надання поточних значень функції та аргументу –  $smax, wmax$
- 10** – виведення значень функції та аргументу

*Текст програми:*

```
REM ОПТИМІЗАЦІЯ МЕТОДОМ ПЕРЕБОРУ
DEF FNF (X) = R/SQR(R^2+(L*X-1/(C*X))^2)
CLS
INPUT " R="; R
INPUT " L="; L
INPUT " C="; C
INPUT " W1="; W1
INPUT " W2="; W2
W=W1
S= FNF (X)
SMAX=S
WMAX=W
FOR W=W1 TO W2 STEP .1
  S= FNF (W)
  IF S>SMAX THEN SMAX=S : WMAX=W
NEXT W
PRINT "SMAX="; SMAX, "WMAX="; WMAX
END
```

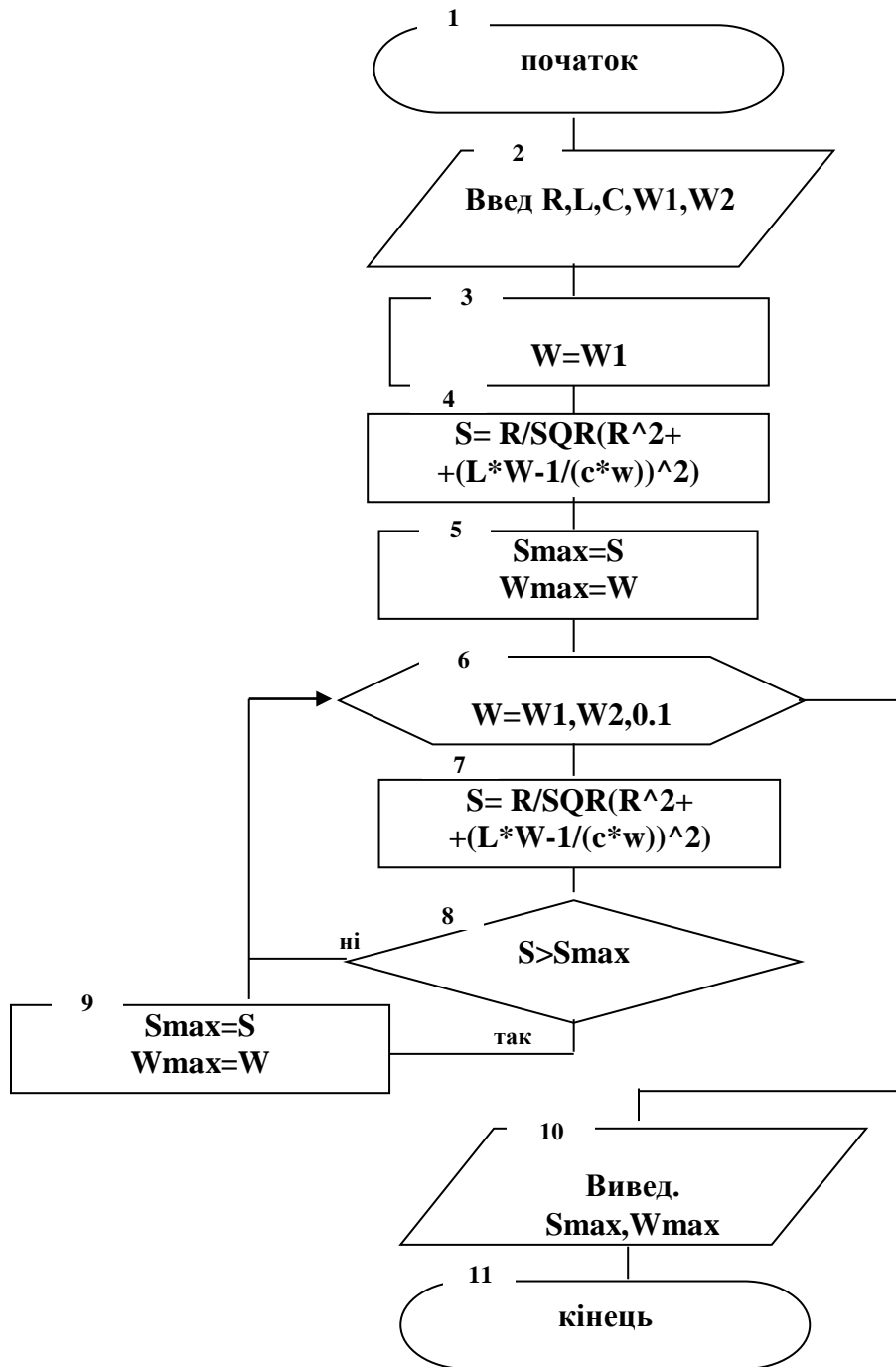


Рис. Д.2.11

**Задача 5**

*Визначення шляху мінімальної вартості методом динамічного програмування.*

Для забезпечення надійності електронної апаратури необхідно здійснити такі заходи:

- підвищити надійність комплектуючих елементів (вид 1);
- ввести інформаційну, алгоритмічну й апаратну надмірності (вид 2).

Необхідно визначити послідовність реалізації заходів, при якій задана надійність буде досягнута з мінімальними витратами.

Оскільки задача двопараметрична, то як модель використовуємо граф, а процедуру оптимізації представляємо як визначення шляху найменшої вартості з початкової вершини в кінцеву. Цей шлях проходить через ряд проміжних точок.

Задані 20 точок (вершин).

Маємо прямокутний граф, галузі якого відповідають проведеним заходам: вертикальні — заходам виду 1; горизонтальні — заходам виду 2.

Оскільки точки розташовані у вигляді матриці, то зручно позначити їхніми номерами рядки і стовпці, у яких вони знаходяться.

$A_{0,0}=4$ , $A_{0,1}=9$	$A_{0,2}=10$	$A_{0,3}=9$	$B_{0,0}=7$ ,	$B_{0,1}=9$	$B_{0,2}=12$	$B_{0,3}=8$	$B_{0,4}=7$
$A_{1,0}=5$ , $A_{1,1}=10$	$A_{1,2}=5$	$A_{1,3}=8$	$B_{1,0}=11$ ,	$B_{1,1}=9$	$B_{1,2}=8$	$B_{1,3}=12$	$B_{1,4}=13$
$A_{2,0}=9$ , $A_{2,1}=6$	$A_{2,2}=11$	$A_{2,3}=5$	$B_{2,0}=8$ ,	$B_{2,1}=6$	$B_{2,2}=5$	$B_{2,3}=13$	$B_{2,4}=9$
$A_{3,0}=5$ , $A_{3,1}=7$	$A_{3,2}=11$	$A_{3,3}=6$					

Відома вартість шляху між сусідніми точками, вона показана у вигляді чисел на лініях, що з'єднують точки.

Вартості горизонтальних та вертикальних кроків з вершин з відповідними індексами:

Ліва нижня точка є початковою, а права верхня — кінцевою. Шлях складається з кроків — відрізків ліній, що з'єднують суміжні точки.

Необхідно:

- накреслити мережу згідно з заданим варіантом;
- виконати розрахунок визначення оптимального шляху;
- відмітити на мережі знайдений оптимальний шлях;
- скласти схему алгоритму розв'язання задачі;
- розробити програму розв'язання задачі на ЕОМ.

Продовження додатка 2

Вихідні 20 точок можна розташувати в такий спосіб — рис. Д.2.12. У такий спосіб весь шлях буде складатися з семи кроків: три вертикальних і чотири горизонтальних. Необхідно знайти таку послідовність кроків, вартість якої буде мінімальною.

На першому етапі оптимізацію кроків починаємо з кінця. Рухаючись від кінцевої точки, записуємо біля вершини графа мінімальні вартості переміщення з даного стану в кінцеву вершину. Рухатися по шляху можна тільки вліво і вниз.

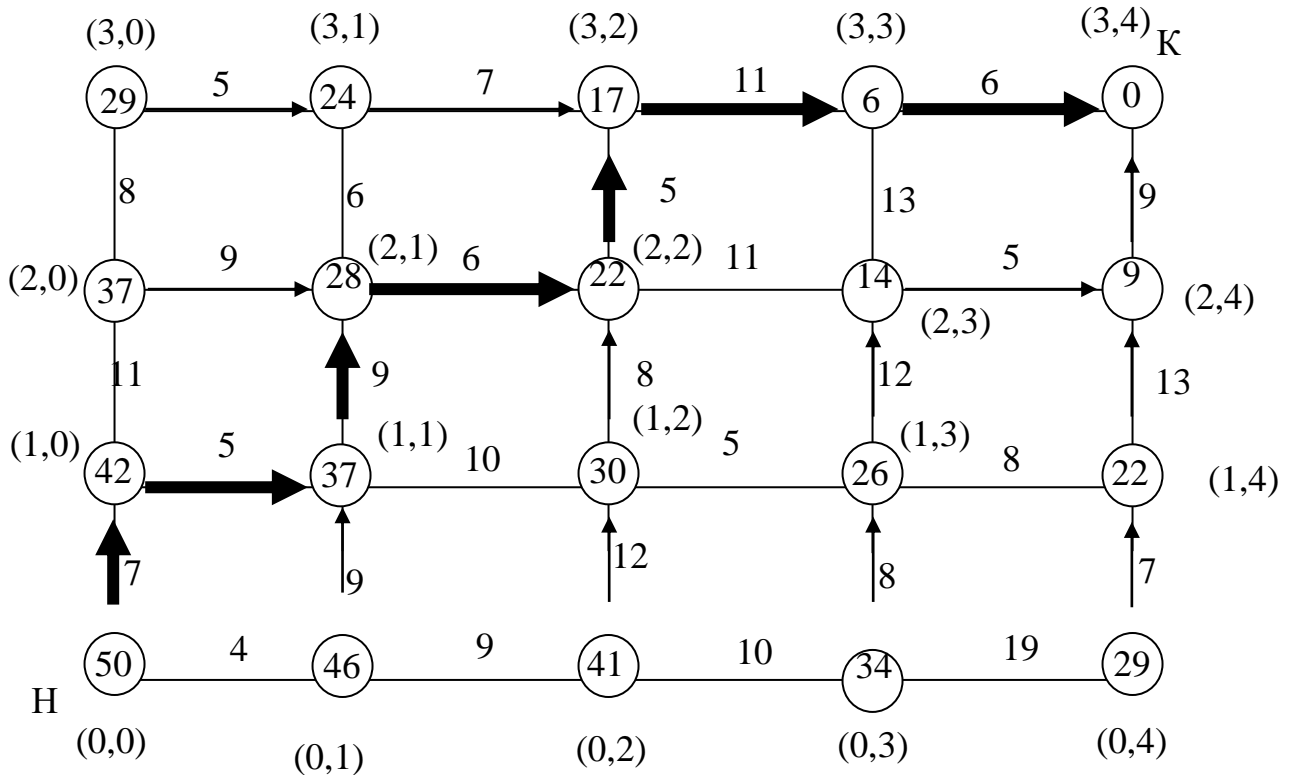


Рис. Д.2.12

У кінцевій точці (3,4) записуємо нуль, тому що вартість шляху з цієї точки в саму себе дорівнює нулю. Далі переходимо до точки (3,3). З цієї точки в кінцеву є єдиний шлях — крок вправо, його вартість дорівнює 6, цей крок і буде оптимальним. Записуємо 6 у точці (3,3) і позначаємо оптимальний крок стрілкою. Наступна точка (3,2), з неї шлях у кінцеву точку також єдиний — крок вправо до точки (3,3) плюс крок від точки (3,3) до точки (3,4). Вартість оптимального шляху з точки (3,3) у кінцеву точку відома і дорівнює 6, отже, вартість оптимального шляху від точки (3,2) до кінцевої

точки дорівнює вартості шляху від точки (3,3) до точки (3,4) плюс вартість кроку з точки (3,2) у точку (3,3), тобто  $6+11=17$ . Записуємо в точці (3,2) 17 і позначаємо стрілкою оптимальний крок. Аналогічно визначаються оптимальні кроки і вартості шляху для всіх точок, що знаходяться у верхньому рядку. Для точок, розташованих у крайньому правому стовпці, дії аналогічні, тому що для них існує єдиний шлях у кінцеву точку, що складається з вертикальних кроків нагору. Для інших точок шлях у кінцеву точку не є єдиним, тому що з них можна робити два кроки — нагору або вправо. Очевидно, що оптимальним буде крок, при якому вартість шляху до кінцевої точки буде меншою. Наприклад, із точки (2,3) можна рухатися через точку (2,4) і через точку (3,3). У першому випадку вартість шляху складає  $5+9=14$ , а в другому —  $13+6=19$ , значить оптимальним є крок вправо. Позначаємо цей крок стрілкою, а в точці (2,3) записуємо 14. Аналогічно визначаються оптимальні кроки і вартості шляху для всіх інших точок. На рис. Д.2.13 показаний результат оптимізації всіх кроків.

На другому етапі оптимізується план у цілому — з оптимальних кроків складаємо оптимальний шлях, рухаючись по стрілках з початкової точки в кінцеву вправо і нагору.

Для запису вищевикладених дій у вигляді алгоритму введемо позначення:

**a(i, j)** — вартість горизонтального кроку;

**b(i, j)** — вартість вертикального кроку;

**c(i, j)** — вартість оптимального шляху з точки (i, j) до кінцевої точки.

Ці величини зручно представити у вигляді двовимірних масивів: **a** — розміром 4 на 4, **b** — розміром 3 на 5, **c** — розміром 4 на 5.

Вихідними даними є масиви **a** і **b**. Нумерацію рядків і стовпців у цих масивах починаємо з нуля, рухаючись від початкової точки до кінцевої.

Продовження додатка 2

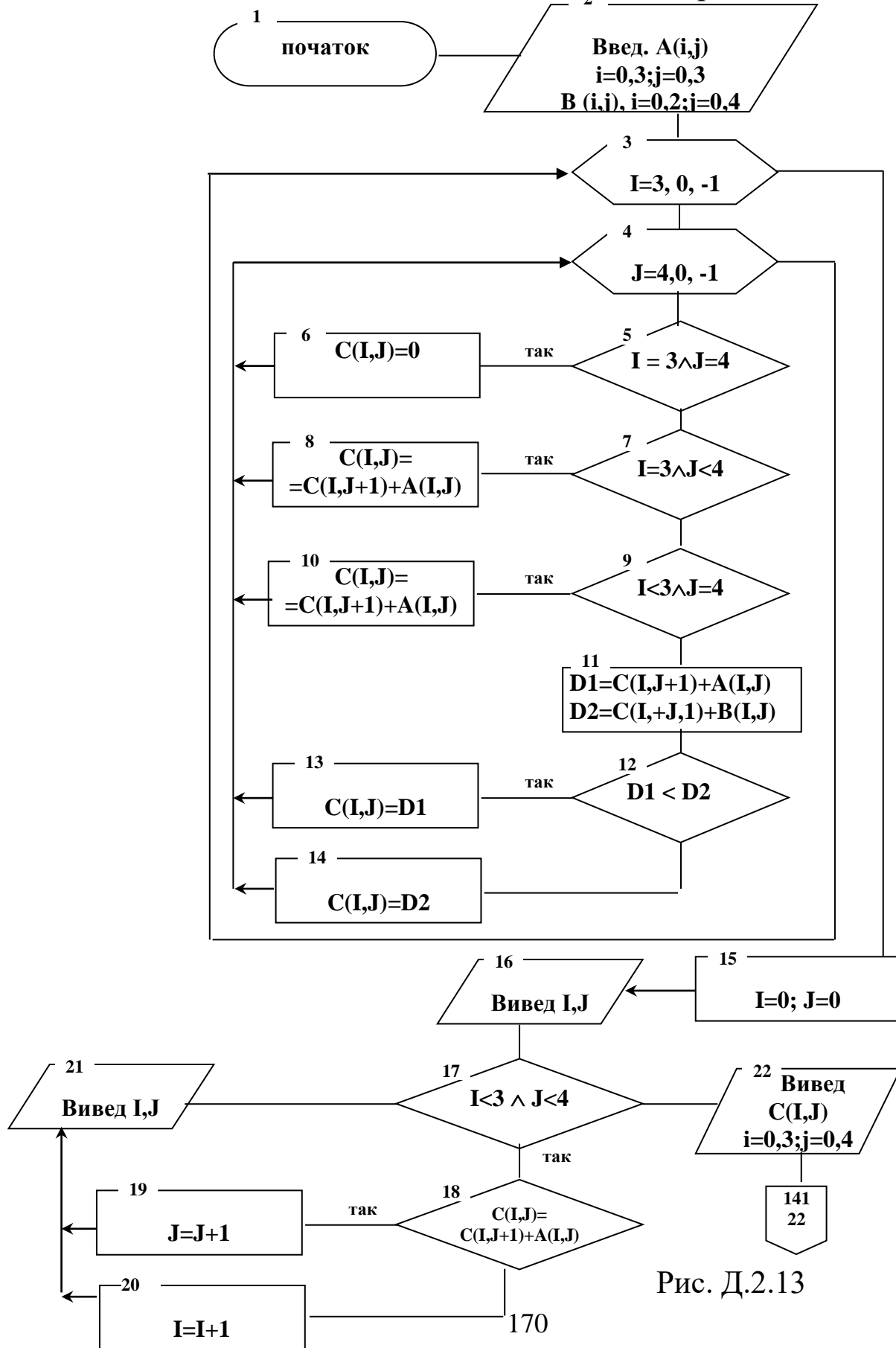


Рис. Д.2.13



Опис алгоритму:

- 1,29 – початок, кінець
- 2 – введення матриць заходів
- 3-14 – заповнення матриць оптимальних вартостей
- 15 – завдання початкового положення шляху
- 17-20 – визначення шляху найменшої вартості для елементів усередині матриці оптимальних вартостей
- 22 – виведення матриці вартості
- 23-24 – визначення шляху найменшої вартості для елементів верхнього рядка
- 26-27 – визначення шляху найменшої вартості для елементів правого стовпця
- 16, 21, 25, 28 – виведення шляху найменшої вартості (координат відповідних вершин)

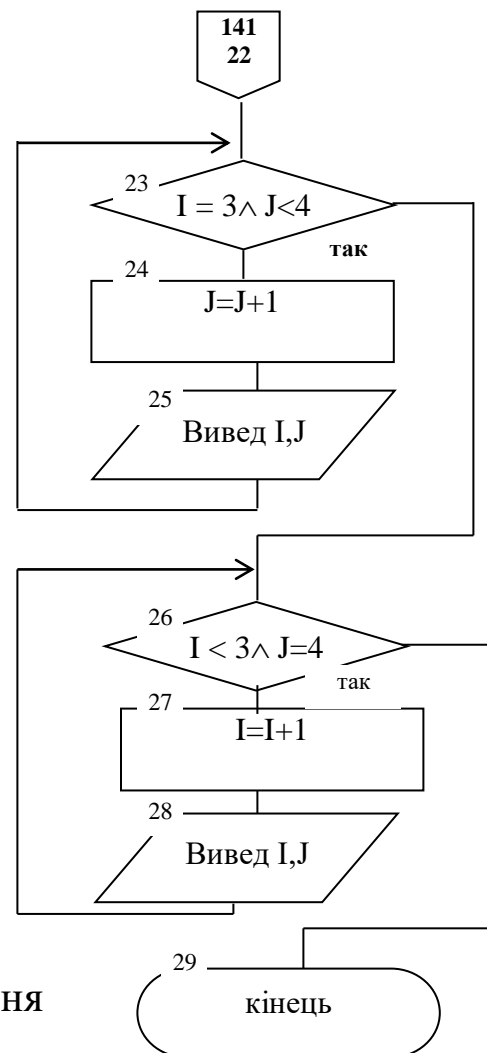


Рис. Д.2.13 Закінчення

*Текст програми:*

```
CLS
DIM C(3, 4) AS INTEGER
DIM B(2, 4) AS INTEGER
DIM A(3, 3) AS INTEGER
‘ВВЕДЕННЯ А
FOR I = 0 TO 3
    FOR J = 0 TO 3
        READ A(I, J)
    NEXT J
NEXT I
DATA 4,9,10,9
DATA 5,10,5,8
DATA 9,6,11,5
DATA 5,7,11,6
‘ВИБЕДЕННЯ А
PRINT "МАТРИЦЯ А"
FOR I = 0 TO 3
    FOR J = 0 TO 3
        PRINT USING "###"; A(I, J);
    NEXT J
    PRINT
NEXT I
‘ВВЕДЕННЯ В
FOR I = 0 TO 2
    FOR J = 0 TO 4
        READ B(I, J)
    NEXT J
NEXT I
```

```
DATA 7,9,12,8,7
DATA 11,9,8,12,13
DATA 8,6,5,13,9
‘ВІВЕДЕННЯ В
PRINT " МАТРИЦЯ В"
FOR I = 0 TO 2
    FOR J = 0 TO 4
        PRINT USING "###"; B(I, J);
    NEXT J
    PRINT
NEXT I
PRINT
PRINT
PRINT "    ВИХІДНИЙ ГРАФ"
FOR I = 3 TO 0 STEP -1
    H = CSRLIN + 1
    FOR J = 0 TO 4
        LOCATE (H), (1 + 15 * J): PRINT "("; I; ","; J; ")";
        IF J <> 4 THEN PRINT USING " ### "; A(I, J);
        IF I <> 0 THEN
            LOCATE (H + 1), (3 + 15 * J)
            PRINT USING " ### "; B(I - 1, J);
        END IF
    NEXT J
NEXT I
PRINT
SLEEP
```

```

FOR I = 3 TO 0 STEP -1
  FOR J = 4 TO 0 STEP -1
    IF I = 3 AND J = 4 THEN
      C(I, J) = 0
    ELSEIF I = 3 AND J < 4 THEN
      C(I, J) = C(I, J + 1) + A(I, J)
    ELSEIF I < 3 AND J = 4 THEN
      C(I, J) = C(I + 1, J) + B(I, J)
    ELSE
      D1 = C(I, J + 1) + A(I, J)
      D2 = C(I + 1, J) + B(I, J)
      IF D1 < D2 THEN C(I, J) = D1 ELSE C(I, J) = D2
    END IF
  NEXT J
NEXT I
CLS
PRINT " ТАБЛИЦЯ ВАРТОСТІ"
FOR I = 3 TO 0 STEP -1
  H = CSRLIN + 1
  FOR J = 0 TO 4
    LOCATE (H), (1 + 10 * J): PRINT " (; I; ", "; J; )"
    LOCATE (H + 1), (3 + 10 * J): PRINT USING " ### "; C(I, J);
  NEXT J
NEXT I
PRINT
PRINT " ШЛЯХ ОПТИМАЛЬНОЇ ВАРТОСТІ "
I = 0
J = 0
PRINT "(; I; ", "; J; )";
DO WHILE I < 3 AND J < 4
  IF C(I, J) = C(I, J + 1) + A(I, J) THEN J = J + 1 ELSE I = I + 1
  PRINT " (; I; ", "; J; )";
LOOP
IF I = 3 AND J < 4 THEN J = J + 1: PRINT " (; I; ", "; J; )";
IF I < 3 AND J = 4 THEN I = I + 1: PRINT " (; I; ", "; J; )";

```

МАТЕМАТИЧНІ МЕТОДИ ТА МОДЕЛІ  
В РОЗРАХУНКАХ НА ЕОМ

*Навчальний посібник*

Відповідальний за випуск Меркулов В.С.

Редактор Ібрагімова Н.В.

---

Підписано до друку 24.12.07 р.

Формат паперу 60x84 1/16 . Папір писальний.

Умовн.-друк.арк. 10,75. Обл.-вид.арк. 11,0.

Замовлення № Тираж 1500. Ціна

---

Видавництво УкрДАЗТу, свідоцтво ДК № 2874 від 12.06.2007 р.  
Друкарня УкрДАЗТу,  
61050, Харків - 50, пл. Фейєрбаха, 7

<sup>2</sup>  
Введ. X<sub>n</sub>,  
X<sub>k</sub>, ΔX