

ПОЛТАВСЬКИЙ УНІВЕРСИТЕТ ЕКОНОМІКИ І ТОРГІВЛІ

Навчально-науковий інститут денної освіти
Форма навчання денна
Кафедра комп'ютерних наук та інформаційних технологій

Допускається до захисту
Завідувач кафедри
_____ Олена ОЛЬХОВСЬКА
(підпис)
«___» _____ 2024 р.

КВАЛІФІКАЦІЙНА РОБОТА
на тему
МЕТОД ГІЛОК ТА МЕЖ ДЛЯ РОЗВ'ЯЗУВАННЯ
КОМБІНАТОРНИХ ТРАНСПОРТНИХ ЗАДАЧ НА
ПЕРЕСТАВЛЕННЯХ, ЙОГО ПРОГРАМНА РЕАЛІЗАЦІЯ

зі спеціальності 122 Комп'ютерні науки
освітня програма «Комп'ютерні науки»
ступеня магістра

Виконавець роботи Шаманський Максим Юрійович
_____ «___» _____ 2024 р.
(підпис)

Науковий керівник к. ф.-м. н., доцент,
Парфьонова Тетяна Олександрівна
_____ «___» _____ 2024 р.
(підпис)

Рецензент _____

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ, ТЕРМІНІВ	3
1. ПОСТАНОВКА ЗАВДАННЯ	6
2. ІНФОРМАЦІЙНИЙ ОГЛЯД.....	7
2.1. Аналіз існуючих методів розв’язання комбінаторних задач оптимізації ...	7
2.2. Аналіз програмних засобів для розв’язання комбінаторних задач	10
2.3. Огляд сфер застосування методі гілок та меж.....	12
3. ТЕОРЕТИЧНА ЧАСТИНА	14
3.1. Постановка комбінаторної транспортної задачі (КТЗ) та її математична модель	14
3.2. Суть методу гілок та меж для розв’язування КТЗ	16
3.3. Алгоритм реалізації прикладу розв’язування комбінаторної транспортної задачі методом гілок та меж	17
4. ПРАКТИЧНА ЧАСТИНА	47
4.1. Блок-схема алгоритму роботи програми.....	47
4.2. Опис програмної реалізації.....	51
4.3. Інструкція користувачу	60
4.4. Актуальність вибору мови програмування.....	60
ВИСНОВКИ.....	62
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	63
ДОДАТОК А. КОД ПРОГРАМИ.....	65

**ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ,
СКОРОЧЕНЬ, ТЕРМІНІВ**

Умовні позначення, символи, скорочення, терміни	Пояснення умовних позначень, скорочень, символів
В&В	Branch and Bound – метод гілок та меж
ДП	Динамічне програмування
КТЗ	Комбінаторні транспортні задачі

ВСТУП

В умовах постійної конкуренції головними завданнями сучасних логістичних систем є оптимізація витрат на транспортування та підвищення ефективності логістичних операцій. Реалізація цих завдань вимагає розроблення ефективних алгоритмів, які дозволяють знайти оптимальні маршрути, мінімізуючи витрати на паливо, час перевезення тощо. Подібні задачі можна розглядати як комбінаторні транспортні задачі на переставленнях. Такі задачі мають широке застосування не тільки в транспортній логістиці, але й в інших галузях, таких як планування виробництва, розподіл енергоресурсів, телекомунікації тощо. Дослідження в цій області має значний потенціал для покращення ефективності роботи в різних сферах.

Існує ряд методів розв'язання комбінаторних задач [2-9]. Одним із них є метод гілок та меж, який реалізується у даній роботі. Дослідження ефективних методів розв'язання комбінаторних транспортних задач на переставленнях є актуальним завданням, що має значну теоретичну та практичну цінність для різних галузей економіки та науки.

Мета роботи – алгоритмізація та програмна реалізація методу гілок та меж для розв'язування комбінаторних транспортних задач на переставленнях.

Головне завдання – створити алгоритм розв'язування комбінаторної транспортної задачі на конкретному прикладі, розробити програму, що його реалізує. Суттєвою умовою є можливість використання отриманого програмного продукту в навчальному процесі під час вивчення курсу «Елементи комбінаторної оптимізації».

Методи дослідження – математичне моделювання транспортних задач комбінаторного типу, метод гілок та меж.

Для програмної реалізації було використано мову програмування Kotlin, фреймворку Compose Desktop та бібліотеки Voyager.

Об'єкт – комбінаторні задачі, методи їх розв'язування та програмна реалізація.

Предмет – комбінаторні транспортні задачі на переставленнях, метод гілок та меж для їх розв'язування, програмна реалізація.

Головними завданнями роботи є вивчення відповідної літератури та аналіз методів для розв'язування подібних задач; дотримання методичних рекомендацій з написання кваліфікаційної роботи; розробка та програмна реалізація алгоритму роботи програми; перевірка коректності та усунення виявлених помилок і недоліків.

Програмна реалізація може бути використана в навчальному процесі під час вивчення курсу «Елементи комбінаторної оптимізації».

Структурні елементи кваліфікаційної роботи включають вступ, чотири розділи та висновки. Перший містить постановку завдання. У другому розділі, інформаційному огляді, здійснено огляд подібних задач та методів їх розв'язування. Третій розділ – теоретична частина. Складається із постановки умови задачі, що реалізується, її математичної моделі, опису основної суті методу гілок та меж для КТЗ, покрокового опису алгоритму реалізації. У четвертому розділі (практичній частині) – блок-схеми, опис програми та актуальність вибору програмних інструментів для досягнення основної мети роботи.

Розроблено функціонуючу програму, яка реалізує приклад розв'язування комбінаторної транспортної задачі на переставленнях методом гілок та меж, яку можна використовувати у навчальному процесі. Особисто розроблено алгоритм розв'язування КТЗ методом гілок та меж на заданому прикладі, здійснено програмну реалізацію даного алгоритму.

Обсяг пояснювальної записки: 146 стор., в т.ч. основна частина - 64 стор., джерела - 12 назв.

1. ПОСТАНОВКА ЗАВДАННЯ

Головним завданням роботи є створення програми, що реалізує метод гілок та меж для розв'язування комбінаторних транспортних задач. В зв'язку з цим виникає ряд підзадач. А саме, необхідно задати умову комбінаторної задачі, розв'язати її методом гілок та меж, обрати мову програмування для реалізації, побудувати алгоритм роботи програми і безпосередньо виконати програмну реалізацію. Для реалізації обрано наступну умову комбінаторної транспортної задачі.

Скласти математичну модель задачі. Умова задана транспортною таблицею з тарифами перевезень, запасами та потребами.

	B_1	B_2	B_3	a_i
A_1	7	1	5	7
A_2	2	4	3	5
A_3	8	6	9	6
b_j	6	6	6	

Обсяги перевезень задані як елементи мультимножини

$$G = \{1,1,1,2,2,2,3,3,3\}.$$

Передбачити в програмі постійну наявність умови задачі на екрані. Крім того, переходити від одного кроку до іншого (наприклад, кнопкою «ДАЛІ»), але в подальшому має бути доступ до пройдених кроків.

Розв'язок даної задачі поділяється на три рівні. Реалізація першого рівня має демонструвати детальне розв'язання задачі згідно алгоритму методу з поясненнями, в інших двох рівнях необхідно передбачити участь користувача у процесі розв'язування. При цьому програма має постійно керувати та коригувати всі дії з метою засвоєння даного методу користувачем.

2. ІНФОРМАЦІЙНИЙ ОГЛЯД

2.1. Аналіз існуючих методів розв'язання комбінаторних задач оптимізації

Існує велика кількість методів, кожен з яких має свої переваги та недоліки, а вибір методу залежить від специфіки задачі. Однак, можна виділити кілька основних груп методів.

1. *Точні методи* гарантують знаходження оптимального розв'язку, але можуть бути складними з точки зору обчислення для великих задач. До точних можна віднести метод повного перебору, динамічне програмування, метод гілок та меж, метод простих методів, метод відсікання площин (метод Гоморі).

Метод повного перебору. Суть цього методу полягає в перевірці всіх можливих варіантів розв'язку задачі. Алгоритм перебирає всі комбінації вхідних даних, оцінюючи кожен з них за заданим критерієм. Варіант, що дає найкращий результат за певним критерієм, вважається розв'язком. Ефективний тільки для дуже малих задач, оскільки його складність експоненційно зростає з розміром задачі.

Динамічне програмування (ДП). Згідно цього методу розв'язання оптимізаційних задач відбувається шляхом розбиття їх на менші підзадачі, розв'язання цих підзадач рекурсивне. Головна ідея – розв'язувати кожен підзадачу лише один раз і зберігати її розв'язок для подальшого використання. Якщо знайдено оптимальні розв'язки для підзадач, то з них можна побудувати оптимальний розв'язок для вихідної задачі. Під час розв'язання вихідної задачі деякі підзадачі зустрічаються неодноразово. ДП зберігає результати розв'язання цих підзадач, щоб уникнути повторних обчислень. Це значно покращує ефективність в порівнянні з методом повного перебору.

Метод гілок та меж. Згідно даного методу перебирається простір розв'язків з умовою врахування оцінок, що відсікають гілки, які не містять

оптимального розв'язку. Ефективний для багатьох задач, але його ефективність залежить від якості обчислення оцінок.

Його суть полягає в систематичному дослідженні простору пошуку шляхом розбиття його на менші підпростори (гілки) та відсікання тих підпросторів (межі), які не можуть містити оптимального розв'язку.

Простір можливих розв'язків представляється у вигляді дерева. Кожен вузол дерева відповідає певній підмножині можливих розв'язків. Кожна гілка надає можливість вибору, який веде до нової підмножини розв'язків. Відбувається процес розбиття вузла дерева на декілька "дочірніх" вузлів. Це відповідає розділенню простору пошуку на менші підпростори.

Для кожного вузла дерева обчислюється оцінка цільової функції, яка дає приблизне уявлення про те, яким може бути найкращий розв'язок у піддереві, що починається з цього вузла. Якщо оцінка для вузла дерева гарантує, що жодний з розв'язків у піддереві не може бути кращим за вже знайдений найкращий розв'язок, то піддерево відсікається – дослідження цього піддерева припиняється. Процес повторюється рекурсивно для решти вузлів, поки не буде знайдено оптимальний розв'язок або не будуть досліджені всі вузли.

Можна виділити основні переваги цього методу, а саме: ефективний для задач з великим простором пошуку, дозволяє значно зменшити кількість перевірок, гарантує знаходження оптимального розв'язку.

Одним із головних недоліків можна зазначити те, що метод може бути складним у реалізації для складних задач.

Ефективність залежить від стратегії розгалуження та оцінки. Неефективні стратегії можуть призвести до дослідження великої кількості вузлів.

Метод простих методів. Застосовується для задач лінійного програмування з цілочисловими обмеженнями, послідовно покращуючи розв'язок шляхом додавання обмежень. Ефективний для деяких типів задач.

Метод відсікання площин (метод Гоморі). Це ітеративний алгоритм для розв'язання задач цілочисельного програмування. Починається розв'язування

задачі лінійного програмування без урахування обмежень цілочисельності, а далі поступово додаються обмеження (площини, що відсікаються), що виключають нецілочлові розв'язки (не виключаючи оптимального цілочислового розв'язку), до тих пір, доки не буде знайдено оптимальне цілочислове рішення. Головна ідея – систематичне звуження області допустимих розв'язків. Цей метод має два варіанти, перший з яких призначений для розв'язку повністю цілочислових задач (перший алгоритм Гоморі), а другий – частково цілочислових задач (другий алгоритм Гоморі). Відрізняються вони способом формування відсікання.

2. Евристичні та метаверистичні методи не гарантують знаходження оптимального розв'язку, але забезпечують знаходження досить гарних розв'язків за прийнятний час, особливо для великих задач. Серед них: жадібний алгоритм, локальний пошук, метаверистичні методи, алгоритми на основі імітації відпалу.

Жадібний алгоритм. Поступово будує розв'язок, на кожному кроці роблячи локально оптимальний вибір. Простий у реалізації, але не завжди знаходить оптимальний розв'язок.

Локальний пошук. Починає з початкового розв'язку та ітеративно покращує його шляхом локальних змін. Пошук спрямований на оптимізацію цільової функції, а не на пошук оптимального шляху. Приклади: hill climbing, simulated annealing, tabu search.

Метаверистичні методи. Об'єднують принципи локального та глобального пошуку в загальну евристичну схему розв'язання задач оптимізації. Основою багатьох з них є локальний пошук: спочатку знаходиться початкове рішення, а потім воно вдосконалюється за допомогою інших методів. Використовуються для пошуку хороших розв'язків в складних ландшафтах. Приклади: генетичні алгоритми, мурашині алгоритми, ройовий інтелект.

Алгоритм на основі імітації відпалу. Це алгоритм для знаходження наближеного глобального оптимуму в задачах дискретної та комбінаторної оптимізації, особливо корисний, коли швидкість знаходження рішення

важливіша за його точність. Алгоритм базується на аналогії з фізичним процесом відпалу.

Табу пошук. Уникає циклічного повторення вже відвіданих розв'язків, використовуючи список заборонених (табу) рухів.

3. Апроксимаційні алгоритми. Гарантують знаходження розв'язку, якість якого може бути оцінена відносно оптимального розв'язку.

Вибір найкращого методу залежить від:

- *Розміру задачі.* Для малих задач можуть підходити точні методи, а для великих – евристичні або метаевристичні.

- *Структури задачі.* Деякі методи ефективні тільки для задач з певною структурою.

- *Вимог до якості розв'язку.* Якщо потрібен оптимальний розв'язок, використовуються точні методи. Якщо достатньо гарного розв'язку, використовуються евристичні або метаевристичні методи.

- *Обчислювальних ресурсів.* Точні методи можуть вимагати значних обчислювальних ресурсів.

2.2. Аналіз програмних засобів для розв'язання комбінаторних задач

Аналіз програмних засобів для розв'язання комбінаторних задач залежить від конкретного типу задач та вимог до розв'язку. Існують різні програмні засоби, що охоплюють широкий спектр методів та можливостей. Ось деякі аспекти, які слід розглянути при аналізі таких засобів:

Типи комбінаторних задач:

Задача комівояжера. Суть задачі полягає в наступному: є комівояжер, який повинен відвідати певне число міст (вузлів), причому кожне місто рівно один раз, і повернутися до вихідного міста. Мета – знайти найкоротший маршрут, який проходить через усі міста.

Задача призначення. Задачі призначення розв'язуються як за допомогою точних алгоритмів (наприклад, угорський алгоритм), так і евристичними методами.

Задача рюкзака. Існують спеціалізовані засоби для розв'язання задач рюкзака з різними типами обмежень та цільових функцій.

Задача розкладу. Для задач розкладу використовуються різні методи, включаючи методи планування проектів, розкладу ресурсів тощо. Програмні засоби часто інтегрують різні алгоритми для розв'язання конкретних сценаріїв.

Загальні комбінаторні задачі. Деякі програмні пакети мають більш загальний підхід, дозволяючи формулювати задачі у вигляді математичних моделей та застосовувати різні алгоритми для їх розв'язання.

Типи алгоритмів: точні алгоритми (гарантують знаходження оптимального розв'язку, але можуть бути дуже повільними для великих задач), евристичні алгоритми (не гарантують оптимального розв'язку, але забезпечують знаходження досить гарного розв'язку за прийнятний час), метаевристичні алгоритми (об'єднують різні евристичні підходи для покращення якості та швидкості пошуку).

Функціональні можливості: моделювання задач (зручність формулювання задачі за допомогою графічного інтерфейсу або мови моделювання), вибір алгоритмів (наявність різноманітних алгоритмів для розв'язання різних типів задач), візуалізація результатів (можливості візуального представлення результатів розв'язання задачі), аналіз чутливості (дозволяє оцінити вплив змін вхідних даних на розв'язок), інтеграція з іншими системами (можливість обміну даними з іншими програмними системами), масштабованість (здатність розв'язувати задачі великого розміру).

Розглянемо деякі програмні засоби.

Спеціалізовані пакети. Існують спеціалізовані пакети для розв'язання конкретних типів комбінаторних задач (наприклад, для TSP). Приклади: Concorde, LKH.

Пакетні системи математичного програмування. Такі як CPLEX, Gurobi, SCIP, дозволяють формулювати та розв'язувати різні задачі математичного програмування, включаючи комбінаторні задачі.

Мови програмування з бібліотеками. Python з бібліотеками SciPy, OR-Tools, NetworkX, або інші мови програмування з відповідними бібліотеками, дозволяють розробляти власні алгоритми та розв'язувачі.

Задля оптимального вибору програмних засобів важливо враховувати критерії оцінки: швидкість роботи (час, необхідний для знаходження розв'язку), якість розв'язку (наскільки близький знайдений розв'язок до оптимального), простота використання (наскільки легко використовувати програмний засіб), достовірність (наскільки надійний програмний засіб), вартість (вартість ліцензії або використання програмного засобу).

2.3. Огляд сфер застосування методі гілок та меж

Метод гілок та меж (Branch and Bound, B&B) – це потужний алгоритмічний підхід до розв'язання задач цілочисельного програмування та задач комбінаторної оптимізації. Цей метод є досить універсальним, що дозволяє застосовувати його в різних сферах. Оглянемо деякі з них.

1. Задачі транспортної логістики.

Оптимізація маршрутів. Планування маршрутів транспорту для доставки товарів з мінімальними витратами часу та ресурсів. Наприклад, доставка товарів з декількох пунктів до декількох пунктів призначення (транспортна задача), або обслуговування клієнтів у певному порядку (задача комівояжера).

Планування розкладу. Розробка оптимальних розкладів перевезень для транспортних засобів, враховуючи обмеження часу, доступності транспортних засобів та обмеження пропускної здатності.

Оптимізація розміщення складів. Визначення оптимального розташування складів для мінімізації витрат на транспорт та зберігання.

2. Задачі виробничого планування.

Планування виробництва. Визначення оптимального плану виробництва для задоволення попиту та мінімізації витрат, враховуючи обмеження ресурсів та часу.

Розрізання матеріалів. Оптимізація розкрою матеріалів (наприклад, рулонів тканини, металевих листів) з метою мінімізації відходів.

3. Задачі планування графіків.

Планування проектів. Оптимізація плану проекту для досягнення заданих цілей у мінімальний термін, враховуючи залежності між завданнями та обмеження ресурсів.

Розклад персоналу. Призначення працівників на різні завдання з урахуванням їхніх навичок, доступності та обмежень на робочий час.

4. Задачі в інших галузях.

Інвестиційне планування. Вибір оптимального портфеля інвестицій з урахуванням ризику та очікуваної прибутковості.

Комп'ютерна графіка. Знаходження оптимальних шляхів для обчислень в алгоритмах комп'ютерної графіки та моделювання.

Біологія та медицина. Розв'язання задач оптимізації в біоінформатиці та моделюванні біологічних процесів.

Телекомунікації. Оптимізація мереж та маршрутизації даних.

Незважаючи на широкі можливості, метод гілок та меж має певні обмеження, а саме: для великих задач розв'язання може вимагати значних обчислювальних ресурсів та часу. Ефективність алгоритму сильно залежить від вибору стратегій. Неправильний вибір може призвести до значного збільшення часу розв'язання.

В цілому, метод гілок та меж є цінним інструментом для розв'язання широкого кола задач оптимізації, але його застосування вимагає ретельного аналізу та вибору відповідних стратегій для досягнення ефективності.

3. ТЕОРЕТИЧНА ЧАСТИНА

3.1. Постановка комбінаторної транспортної задачі (КТЗ) та її математична модель

Постановка КТЗ. Нехай виробляється деякий однорідний продукт, який необхідно розподілити між n споживачами $B_j, j = 1, \dots, n$.

Задача полягає в тому, щоб визначити обсяги перевезень від i -го виробника до j -го споживача для всіх $i \in J_m, j \in J_n$, які забезпечують мінімальні сумарні витрати на перевезення продукту. Разом з цим нехай перевезення вантажу можна здійснити певними місткостями об'ємами $g_1, g_2, \dots, g_k$ в кількості $k, k = m \cdot n$.

Введемо наступні позначення:

A_i – можливі пункти виробництва в кількості $m, i = 1, \dots, m$.

a_i – максимально можливі обсяги виробництва продукції в i -му пункті, $i \in J_m$.

b_j – мінімально можливі обсяги споживання в j -му пункті призначення, $j \in J_n$.

c_{ij} – транспортні витрати на перевезення одиниці продукції від i -го виробника до j -го споживача $i \in J_m, j \in J_n$.

x_{ij} – обсяги перевезень від i -го виробника до j -го споживача, $i \in J_m, j \in J_n$.

Математична модель КТЗ.

Цільова функція

$$\sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} \rightarrow \min \quad (3.1)$$

Обмеження

$$\sum_{j=1}^n x_{ij} = a_i, \quad i = \overline{1, m} \quad (3.2)$$

$$\sum_{i=1}^m x_{ij} = b_j, \quad j = \overline{1, n} \quad (3.3)$$

$$\sum_{i=1}^m a_i = \sum_{j=1}^n b_j, \quad (3.4)$$

$$x = (x_{11}, x_{12}, \dots, x_{1n}, \dots, x_{m1}, \dots, x_{mn}) \in E_k(G), \quad (3.5)$$

де $E_k(G)$ – множина переставлень місткостей, якими перевозиться продукт. Іншими словами $E_k(G)$ – множина переставлень з елементів мультимножини $G = \{g_1, g_2, \dots, g_k\}$.

Теорема 3.1. Для того, щоб задача (3.1)-(3.5) мала розв'язок необхідно, щоб для елементів мультимножини G виконувалась умова

$$\sum_{r=1}^k g_r = \sum_{i=1}^m a_i = \sum_{j=1}^n b_j, \quad (3.6)$$

Теорема 3.2. Для того, щоб задача (3.1)-(3.5) мала розв'язок необхідно, щоб для елементів мультимножини G виконувались умови

$$\sum_{j=1}^n g_{\alpha_j} = a_i, i \in J_m, \quad (3.7)$$

$$\sum_{i=1}^m g_{\beta_i} = b_j, j \in J_n. \quad (3.8)$$

Тут $g_{\alpha_j}, g_{\beta_i} \in G$, $\alpha_j, \beta_i, (\alpha_1, \dots, \alpha_n), (\beta_1, \dots, \beta_m) \in J_k \forall i \in J_m, \forall j \in J_n$,
 $\alpha_i \neq \alpha_j, \beta_i \neq \beta_j \forall i \neq j$.

Приклад 3.1. Розглянемо приклад КТЗ, для якої не виконується умова теореми 3.2, тобто ця задача не має допустимих розв'язків.

Умова задана таблицею 3.1.

Таблиця 3.1 – Умова КТЗ до прикладу 3.1.

	B_1	B_2	a_i
A_1	1	5	15
A_2	4	2	5
A_3	3	6	10
b_j	15	15	

$G = \{0, 0, 1, 9, 10, 10\}$.

Умова теореми 3.1 виконується, тобто

$$\sum_{r=1}^k g_r = \sum_{i=1}^m a_i = \sum_{j=1}^n b_j = 30,$$

де

$$\sum_{r=1}^k g_r = 0 + 0 + 1 + 9 + 10 + 10 = 30,$$

$$\sum_{i=1}^m a_i = 15 + 5 + 10 = 30, \sum_{j=1}^n b_j = 15 + 15 = 30.$$

Але задача не має допустимих розв'язків, адже не виконуються умови теореми 3.2. Зокрема, умова $x_{21} + x_{22} = 5$ не може бути виконана, так як серед елементів мультимножини G немає таких двох, які б в сумі дали 5.

3.2. Суть методу гілок та меж для розв'язування КТЗ

Розглянемо суть методу гілок та меж у застосуванні до розв'язування комбінаторної транспортної задачі на переставленнях (3.1)-(3.5). Шляхом перебору можливих варіантів, враховуючи оцінки, цей метод дозволяє знайти план перевезень з мінімальними витратами.

На першому етапі перевіряємо умову балансу запасів і потреб, а також виконання умов теорем 3.1 та 3.2. Якщо вони виконуються, то переходимо на наступний крок. На цьому етапі необхідно відсортувати елементи c_{ij} множини вартостей по не спаданню та елементи g_t множини об'ємів місткостей по не зростанню.

Далі обираємо серед елементів c_{ij} множини вартостей мінімальне значення і обираємо відповідне йому x_{ij} . На першому рівні присвоюємо значення обраній змінній x_{ij} , що дорівнює максимальному значенню із мультимножини G . Якщо знайти основу мультимножини G (кортеж її різних елементів), то кількість елементів основи визначає кількість рівнів у дереві розв'язку.

Врахуємо деякі позначення. G_B – множина вибраних елементів з G . $\tilde{G}$ – різниця мультимножини G та G_B , тобто $\tilde{G} = G - G_B$.

Нехай c_l^* – коефіцієнт біля визначених змінних в цільовій функції, $\tilde{c}_r$ – коефіцієнт біля невизначених змінних в ній. Зафіксовані змінні в G позначимо $z_l = x_{ij} = g_{i_l}$, $l = 1, \dots, t$ та упорядкуємо $z_1 \geq z_2 \geq \dots \geq z_t$. Тоді маємо $g_{i_1} \geq g_{i_2} \geq \dots \geq g_{i_t}$. Незафіксовані змінні в G позначимо $y_r = x_{ij}$, $y_1 \geq y_2 \geq \dots \geq y_\tau$, де $\tau = k - t$.

Упорядкуємо елементи $\tilde{G} = \{\widetilde{g}_{i_1}, \widetilde{g}_{i_2}, \dots, \widetilde{g}_{i_\tau}\}$.

$$\widetilde{g}_{i_1} \geq \widetilde{g}_{i_2} \geq \dots \geq \widetilde{g}_{i_\tau}.$$

А також $c_l^* \leq c_{l+1}^* \forall l = 1, \dots, t-1$, $\tilde{c}_r \leq \tilde{c}_{r+1} \forall r = 1, \dots, \tau-1$.

На кожній ітерації обчислюємо оцінки

$$v = \sum_{l=1}^t c_l^* g_{i_l} \text{ та } \xi = \sum_{l=1}^t c_l^* g_{i_l} + \sum_{r=1}^{\tau} \tilde{c}_r \widetilde{g}_{i_r}.$$

Нехай на деякому етапі галуження знайшли значення цільової функції F_0 . Тоді на наступних кроках слідкуємо, щоб виконувалась умова: $\xi \leq F_0$, інакше – відсікаємо розв'язок. Якщо знаходимо нове значення цільової функції $F_1 \leq F_0$, то вибираємо значення F_1 . Виконавши повний перебір, необхідні відсікання, отримаємо мінімальне значення цільової функції та відповідний план перевезення.

3.3. Алгоритм реалізації прикладу розв'язування комбінаторної транспортної задачі методом гілок та меж

На екрані умова КТЗ (див.розділ 1), яка постійно доступна.

Крок 1. На екрані з'являється наступна інформація:

Матриця вартостей перевезення продукції від i -го виробника до j -го

споживача має вигляд: $C = \begin{pmatrix} 7 & 1 & 5 \\ 2 & 4 & 3 \\ 8 & 6 & 9 \end{pmatrix}.$

Для того, щоб дана задача мала розв'язок необхідно, щоб елементи g_t

($t = \overline{1, k}, k = m \cdot n$) в G задовольняли умову $\sum_{t=1}^k g_t = \sum_{i=1}^m a_i = \sum_{j=1}^n b_j$.

Для даної задачі ця умова виконується, а саме:

$$\sum_{t=1}^9 g_t = 1 + 1 + 1 + 2 + 2 + 2 + 3 + 3 + 3 = 18.$$

$$\sum_{i=1}^3 a_i = 7 + 5 + 6 = 18. \quad \sum_{j=1}^3 b_j = 6 + 6 + 6 = 18.$$

Крок 2. На екрані: Математична модель задачі:

Цільова функція:

$$7x_{11} + x_{12} + 5x_{13} + 2x_{21} + 4x_{22} + 3x_{23} + 8x_{31} + 6x_{32} + 9x_{33} \rightarrow \min$$

УМОВИ

$$x_{11} + x_{12} + x_{13} = 7, x_{21} + x_{22} + x_{23} = 5,$$

$$x_{31} + x_{32} + x_{33} = 6, x_{11} + x_{21} + x_{31} = 6,$$

$$x_{12} + x_{22} + x_{32} = 6, x_{13} + x_{23} + x_{33} = 6.$$

$$x = (x_{11}, x_{12}, x_{13}, x_{21}, x_{22}, x_{23}, x_{31}, x_{32}, x_{33}) \in E_{9,9}(G).$$

Крок 3. На екрані:

Відсортуємо вартості перевезення продукції по не спаданню. Отримаємо

$$c_{12} = 1 < c_{21} = 2 < c_{23} = 3 < c_{22} = 4 < c_{13} = 5 < c_{32} = 6 < c_{11} = 7 < c_{31} = 8 < c_{33} = 9. \text{ Тобто}$$

$$c_{ij}: 1 \leq 2 \leq 3 \leq 4 \leq 5 \leq 6 \leq 7 \leq 8 \leq 9.$$

Відсортуємо елементи множини G по не зростанню. Отримаємо

$$g_t: 3 \geq 3 \geq 3 \geq 2 \geq 2 \geq 2 \geq 1 \geq 1 \geq 1.$$

Крок 4. На екрані: Основа мультимножини G матиме вигляд:

$$S(G) = (1, 2, 3).$$

Кількість елементів у $S(G)$ відповідає кількості рівнів, що будуть реалізовані у заданій задачі. $|S(G)| = 3$. Тобто буде 3 рівні.

Галуження виконуємо по принципу: найменшому коефіцієнту цільової функції ставиться у відповідність найбільший елемент із G .

Найменший коефіцієнт цільової функції (вартість перевезення) $c_{12} = 1$. Тому починаємо на кожному рівні присвоювати значення із $S(G)$ змінній x_{12} від максимального до мінімального. У прикладі матимемо:

$$\text{Рівень 1: } x_{12} = 3. \quad \text{Рівень 2: } x_{12} = 2. \quad \text{Рівень 3: } x_{12} = 1.$$

Крок 5. На екрані: *Рівень 1*. Найменший коефіцієнт цільової функції (вартість перевезення) $c_{12} = 1$. Тому присвоювати значення будемо змінній x_{12} . Найбільший елемент із G це 3. Отже, на початку першого рівня матимемо $x_{12} = 3$ (табл.3.1).

Крок 6. На екрані: $G_B = \{3\}$ – множина використаних елементів з G .
 $\tilde{G} = \{1,1,1,2,2,2,3,3\}$, де $\tilde{G} = G - G_B$.

За умовою $x_{11} + x_{12} + x_{13} = 7$, $x_{21} + x_{22} + x_{23} = 5$, $x_{31} + x_{32} + x_{33} = 6$,

$$x_{11} + x_{21} + x_{31} = 6, x_{12} + x_{22} + x_{32} = 6, x_{13} + x_{23} + x_{33} = 6.$$

Тоді після підстановки x_{12} маємо (табл.1):

$$x_{11} + x_{13} = 7 - 3 = 4, x_{21} + x_{22} + x_{23} = 5, x_{31} + x_{32} + x_{33} = 6,$$

$$x_{11} + x_{21} + x_{31} = 6, x_{22} + x_{32} = 6 - 3 = 3, x_{13} + x_{23} + x_{33} = 6.$$

Таблиця 3.1 – Присвоєння x_{12} на рівні 1.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		3		7	4
A_2				5	5
A_3				6	6
b_j	6	6	6		
Залишок потреб після заповнення клітин	6	3	6		

Крок 7. На екрані: Обчислюємо оцінки:

для присвоєного значення x_{12} : $\nu = c_{12} \cdot x_{12} = 1 \cdot 3 = 3$.

$\xi = \nu + c^*$, яка обчислюється відповідно до відсортованих c_{ij} та g_t

c_{ij} : обране 1, інші відсортовані $2 \leq 3 \leq 4 \leq 5 \leq 6 \leq 7 \leq 8 \leq 9$.

g_t : обране 3, інші відсортовані $3 \geq 3 \geq 2 \geq 2 \geq 2 \geq 1 \geq 1 \geq 1$.

Тобто $\xi = 3 + 2 \cdot 3 + 3 \cdot 3 + 4 \cdot 2 + 5 \cdot 2 + 6 \cdot 2 + 7 \cdot 1 + 8 \cdot 1 + 9 \cdot 1 = 72$.

Крок 8. На екрані: *Рівень 1.1*. Наступне найменше значення коефіцієнту цільової функції $c_{21} = 2$. Тому присвоювати значення будемо змінній x_{21} .

Найбільший елемент із $\tilde{G}$ після попереднього вибору – це 3. Отже, далі матимемо $x_{21} = 3$.

$G_B = \{3,3\}$ – множина використаних елементів з G .

$\tilde{G} = \{1,1,1,2,2,2,3\}$, де $\tilde{G} = G - G_B$.

$$x_{11} + x_{13} = 4, x_{22} + x_{23} = 5 - 3 = 2, x_{31} + x_{32} + x_{33} = 6,$$

$$x_{11} + x_{31} = 6 - 3 = 3, x_{22} + x_{32} = 3, x_{13} + x_{23} + x_{33} = 6.$$

На цьому етапі таблиця 3.1 змінюється на таблицю 3.2.

Таблиця 3.2 – Присвоєння x_{21} на рівні 1.1.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		3		7	4
A_2	3			5	2
A_3				6	6
b_j	6	6	6		
Залишок потреб після заповнення клітин	3	3	6		

Крок 9. На екрані: Обчислюємо оцінки:

для присвоєних значень x_{12} та x_{21} $v = c_{12} \cdot x_{12} + c_{21} \cdot x_{21} = 1 \cdot 3 + 2 \cdot 3 = 9$.

$$\xi = v + c^*,$$

яка обчислюється відповідно до відсортованих c_{ij} та g_t

c_{ij} : обрані 1,2; інші відсортовані $3 \leq 4 \leq 5 \leq 6 \leq 7 \leq 8 \leq 9$.

g_t : обрані 3,3; інші відсортовані $3 \geq 2 \geq 2 \geq 2 \geq 1 \geq 1 \geq 1$.

Тобто $\xi = 9 + 3 \cdot 3 + 4 \cdot 2 + 5 \cdot 2 + 6 \cdot 2 + 7 \cdot 1 + 8 \cdot 1 + 9 \cdot 1 = 72$.

Крок 10. На екрані: **Рівень 1.1.1.** Наступне найменше значення коефіцієнту цільової функції $c_{23} = 3$. Тому присвоювати значення будемо змінній x_{23} . Найбільший елемент із $\tilde{G}$ після попереднього вибору – це 3. Отже, далі матимемо $x_{23} = 3$. $G_B = \{3,3,3\}$ – множина використаних елементів з G .

$\tilde{G} = \{1,1,1,2,2,2\}$, де $\tilde{G} = G - G_B$. $x_{11} + x_{13} = 4$, $x_{22} + x_{23} = 2$,

$x_{22} = 2 - 3 = -1 < 0$, тобто для $x_{21} = 3$ та $x_{23} = 3$ не виконується умова

$x_{21} + x_{22} + x_{23} = 5$. отже, відсікаємо $x_{23} = 3$.

Крок 11. На екрані: *Рівень 1.1.2*. Так як відсікаємо $x_{23} = 3$, то розглядаємо наступний найбільший елемент із $\tilde{G}$ – це 2. Отже, далі матимемо $x_{23} = 2$.

$G_B = \{2,3,3\}$ – множина використаних елементів з G .

$\tilde{G} = \{1,1,1,2,2,3\}$, де $\tilde{G} = G - G_B$. $x_{11} + x_{13} = 4$, $x_{22} + x_{23} = 2$, $x_{22} = 2 - 2 = 0$.

А так як $0 \in \tilde{G} = \{1,1,1,2,2,3\}$, то відсікаємо $x_{23} = 2$.

Крок 12. На екрані: *Рівень 1.1.3*. Так як відсікаємо $x_{23} = 2$, то розглядаємо наступний найбільший елемент із $\tilde{G}$ – це 1. Отже, далі матимемо $x_{23} = 1$.

$G_B = \{1,3,3\}$ – множина використаних елементів з G .

$\tilde{G} = \{1,1,2,2,2,3\}$, де $\tilde{G} = G - G_B$.

На цьому етапі таблиця 3.2 змінюється на таблицю 3.3.

$$x_{11} + x_{13} = 4, x_{22} = 2 - 1 = 1, x_{31} + x_{32} + x_{33} = 6,$$

$$x_{11} + x_{31} = 3, x_{22} + x_{32} = 3, x_{32} = 3 - 1 = 2, x_{13} + x_{33} = 6 - 1 = 5.$$

Отримаємо $x_{23} = 1$, $x_{22} = 1$ та $x_{32} = 2$. (табл.3.4).

Крок 13. На екрані: Обчислюємо оцінки:

для присвоєних значень x_{12} , x_{21} , x_{23} , x_{22} та x_{32}

$$\begin{aligned} v &= c_{12} \cdot x_{12} + c_{21} \cdot x_{21} + c_{23} \cdot x_{23} + c_{22} \cdot x_{22} + c_{32} \cdot x_{32} = \\ &= 1 \cdot 3 + 2 \cdot 3 + 3 \cdot 1 + 4 \cdot 1 + 6 \cdot 2 = 28. \end{aligned}$$

Таблиця 3.3 – Присвоєння x_{23} на рівні 1.1.3.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		3		7	4
A_2	3		1	5	1
A_3				6	6
b_j	6	6	6		
Залишок потреб після заповнення клітин	3	3	5		

$$\xi = v + c^*,$$

яка обчислюється відповідно до відсортованих та обраних c_{ij} та g_t

c_{ij} : обрані 1,2,3,4,6; інші відсортовані: $5 \leq 7 \leq 8 \leq 9$.

g_t : обрані 3,3,1,1,2; інші відсортовані $3 \geq 2 \geq 2 \geq 1$.

Таблиця 3.4 – Присвоєння змінних на рівні 1.1.3.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		3		7	4
A_2	3	1	1	5	0
A_3		2		6	4
b_j	6	6	6		
Залишок потреб після заповнення клітин	3	0	5		

Тобто $\xi = 28 + 5 \cdot 3 + 7 \cdot 2 + 8 \cdot 2 + 9 \cdot 1 = 82$.

Крок 14. На екрані: *Рівень 1.1.3.1*. З попереднього кроку маємо

$$G_B = \{1,1,2,3,3\}, \tilde{G} = \{1,2,2,3\}.$$

Наступне найменше значення коефіцієнту цільової функції $c_{13} = 5$. Тому присвоювати значення будемо змінній x_{13} . Найбільший елемент із $\tilde{G}$ після попереднього вибору – це 3. Отже, далі матимемо $x_{13} = 3$.

$G_B = \{1,1,2,3,3,3\}$ – множина використаних елементів з G .

$\tilde{G} = \{1,2,2\}$, де $\tilde{G} = G - G_B$.

$x_{11} + x_{13} = 4$, $x_{11} = 4 - 3 = 1 \in \{1,2,2\}$, тоді $\tilde{G} = \{2,2\}$, $x_{13} + x_{33} = 5$,

$x_{33} = 5 - 3 = 2 \in \{2,2\}$, тоді $\tilde{G} = \{2\}$, $x_{11} + x_{31} = 3$, $x_{31} = 3 - 1 = 2 \in \{2\}$.

Отже, $x_{13} = 3$, $x_{11} = 1$, $x_{33} = 2$, $x_{31} = 2$ (табл.3.5).

Крок 15. На екрані:

Так як всі клітинки заповнені, то обчислюємо значення цільової функції:

$$F_0 = c_{12} \cdot x_{12} + c_{21} \cdot x_{21} + c_{23} \cdot x_{23} + c_{22} \cdot x_{22} + c_{32} \cdot x_{32} + c_{13} \cdot x_{13} + c_{11} \cdot x_{11} + c_{33} \cdot x_{33} + c_{31} \cdot x_{31} = 28 + 5 \cdot 3 + 7 \cdot 1 + 9 \cdot 2 + 8 \cdot 2 = 84.$$

Таблиця 3.5 – Присвоєння змінних на рівні 1.1.3.1.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1	1	3	3	7	0
A_2	3	1	1	5	0
A_3	2	2	2	6	0
b_j	6	6	6		
Залишок потреб після заповнення клітин	0	0	0		

Крок 16. На екрані: *Рівень 1.1.3.2.* Маємо $G_B = \{1,1,2,3,3\}$. $\tilde{G} = \{1,2,2,3\}$.

Тепер знову будемо присвоювати значення змінній x_{13} . Наступний найбільший елемент із $\tilde{G}$ після 3 – це 2. Отже, далі матимемо $x_{13} = 2$.

$G_B = \{1,1,2,2,3,3\}$ – множина використаних елементів з G .

$\tilde{G} = \{1,2,3\}$, де $\tilde{G} = G - G_B$.

$x_{11} + x_{13} = 4$, $x_{11} = 4 - 2 = 2 \in \{1,2,3\}$, тоді $\tilde{G} = \{1,3\}$, $x_{13} + x_{33} = 5$,

$x_{33} = 5 - 2 = 3 \in \{1,3\}$, тоді $\tilde{G} = \{1\}$, $x_{11} + x_{31} = 3$, $x_{31} = 3 - 2 = 1 \in \{1\}$.

Отже, $x_{13} = 2$, $x_{11} = 2$, $x_{33} = 3$, $x_{31} = 1$ (табл.3.6).

Таблиця 3.6 – Присвоєння змінних на рівні 1.1.3.2

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1	2	3	2	7	0
A_2	3	1	1	5	0
A_3	1	2	3	6	0
b_j	6	6	6		
Залишок потреб після заповнення клітин	0	0	0		

Крок 17. На екрані:

Так як всі клітинки заповнені, то обчислюємо значення цільової функції:

$$F_1 = c_{12} \cdot x_{12} + c_{21} \cdot x_{21} + c_{23} \cdot x_{23} + c_{22} \cdot x_{22} + c_{32} \cdot x_{32} + c_{13} \cdot x_{13} + c_{11} \cdot x_{11} + c_{33} \cdot x_{33} + c_{31} \cdot x_{31} = 87.$$

$F_1 = 87 > F_0 = 84$. Тому цей розв'язок відсікаємо.

Крок 18. На екрані: *Рівень 1.1.3.3.* Маємо $G_B = \{1,1,2,3,3\}$. $\tilde{G} = \{1,2,2,3\}$.

Тепер знову будемо присвоювати значення змінній x_{13} . Наступний найбільший елемент із $\tilde{G}$ після 2 – це 1. Отже, далі матимемо $x_{13} = 1$.

$G_B = \{1,1,1,2,3,3\}$ – множина використаних елементів з G .

$\tilde{G} = \{2,2,3\}$, де $\tilde{G} = G - G_B$. $x_{11} + x_{13} = 4$, $x_{11} = 4 - 1 = 3 \in \{2,2,3\}$, тоді

$\tilde{G} = \{2,2\}$, $x_{13} + x_{33} = 5$, $x_{33} = 5 - 1 = 4 \notin \{2,2\}$. Тоді відсікається $x_{13} = 1$.

Крок 19. На екрані: *Рівень 1.2.* Присвоєне значення змінній на рівні 1 – $x_{12} = 3$. Наступне найменше значення коефіцієнту цільової функції $c_{21} = 2$. Тому присвоювати значення будемо змінній x_{21} . Найбільший елемент із $\tilde{G}$ після 3 – це 2. Отже, далі матимемо $x_{21} = 2$.

$G_B = \{2,3\}$ – множина використаних елементів з G .

$\tilde{G} = \{1,1,1,2,2,3,3\}$, де $\tilde{G} = G - G_B$.

$$x_{11} + x_{13} = 4, x_{22} + x_{23} = 5 - 2 = 3, x_{31} + x_{32} + x_{33} = 6,$$

$$x_{11} + x_{31} = 6 - 2 = 4, x_{22} + x_{32} = 3, x_{13} + x_{23} + x_{33} = 6.$$

На цьому етапі таблиця 3.1 змінюється на таблицю 3.7.

Таблиця 3.7 – Присвоєння x_{21} на рівні 1.2.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		3		7	4
A_2	2			5	3
A_3				6	6
b_j	6	6	6		
Залишок потреб після заповнення клітин	4	3	6		

Крок 20. На екрані: Обчислюємо оцінки:

для присвоєних значень x_{12} та x_{21} $v = c_{12} \cdot x_{12} + c_{21} \cdot x_{21} = 1 \cdot 3 + 2 \cdot 2 = 7$.

$$\xi = v + c^*,$$

яка обчислюється відповідно до відсортованих c_{ij} та g_t

c_{ij} : обрані 1,2; інші відсортовані $3 \leq 4 \leq 5 \leq 6 \leq 7 \leq 8 \leq 9$.

g_t : обрані 3,2; інші відсортовані $3 \geq 3 \geq 2 \geq 2 \geq 1 \geq 1 \geq 1$.

Тобто $\xi = 7 + 3 \cdot 3 + 4 \cdot 3 + 5 \cdot 2 + 6 \cdot 2 + 7 \cdot 1 + 8 \cdot 1 + 9 \cdot 1 = 74$.

Крок 21. На екрані: *Рівень 1.2.1*. Наступне найменше значення коефіцієнту цільової функції $c_{23} = 3$. Тому присвоювати значення будемо змінній x_{23} . Найбільший елемент із $\tilde{G}$ після попереднього вибору – це 3. Отже, далі матимемо $x_{23} = 3$.

$G_B = \{2,3,3\}$ – множина використаних елементів з G .

$\tilde{G} = \{1,1,1,2,2,3\}$, де $\tilde{G} = G - G_B$. $x_{11} + x_{13} = 4$, $x_{22} + x_{23} = 3$,

$x_{22} = 3 - 3 = 0$, але $0 \notin \tilde{G} = \{1,1,1,2,2,3\}$. Отже, відсікаємо $x_{23} = 3$.

Крок 22. На екрані: *Рівень 1.2.2*. Наступне найменше значення коефіцієнту цільової функції $c_{23} = 3$. Тому присвоювати значення будемо змінній x_{23} . Найбільший елемент із $\tilde{G}$ після попереднього вибору – це 2. Отже, далі матимемо $x_{23} = 2$ (табл.3.8). $G_B = \{2,2,3\}$ – множина використаних елементів з G .

$\tilde{G} = \{1,1,1,2,3,3\}$, де $\tilde{G} = G - G_B$.

$x_{11} + x_{13} = 4$, $x_{22} + x_{23} = 3$, $x_{22} = 3 - 2 = 1$, $x_{31} + x_{32} + x_{33} = 6$,

$x_{11} + x_{31} = 4$, $x_{22} + x_{32} = 3$, $x_{32} = 3 - 1 = 2$, $x_{13} + x_{23} + x_{33} = 6$,

$x_{13} + x_{33} = 6 - 2 = 4$. Отримаємо $x_{23} = 2$, $x_{22} = 1$ та $x_{32} = 2$. (табл.3.9).

Крок 23. На екрані: Обчислюємо оцінки:

для присвоєних значень x_{12} , x_{21} , x_{23} , x_{22} та x_{32}

$$\begin{aligned} v &= c_{12} \cdot x_{12} + c_{21} \cdot x_{21} + c_{23} \cdot x_{23} + c_{22} \cdot x_{22} + c_{32} \cdot x_{32} = \\ &= 1 \cdot 3 + 2 \cdot 2 + 3 \cdot 2 + 4 \cdot 1 + 6 \cdot 2 = 29. \end{aligned}$$

Таблиця 3.8 – Присвоєння x_{23} на рівні 1.2.2.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		3		7	4
A_2	2		2	5	1
A_3				6	6
b_j	6	6	6		
Залишок потреб після заповнення клітин	4	3	4		

Таблиця 3.9 – Присвоєння змінних на рівні 1.2.2.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		3		7	4
A_2	2	1	2	5	0
A_3		2		6	4
b_j	6	6	6		
Залишок потреб після заповнення клітин	4	0	4		

$$\xi = v + c^*,$$

яка обчислюється відповідно до відсортованих та обраних c_{ij} та g_t

c_{ij} : обрані 1,2,3,4,6; інші відсортовані: $5 \leq 7 \leq 8 \leq 9$.

g_t : обрані 3, 2, 2, 1, 2; інші відсортовані $3 \geq 3 \geq 1 \geq 1$.

Тобто $\xi = 29 + 5 \cdot 3 + 7 \cdot 3 + 8 \cdot 1 + 9 \cdot 1 = 82$.

Крок 24. На екрані: *Рівень 1.2.2.1*. З попереднього кроку маємо

$$G_B = \{1,2,2,2,3\}. \tilde{G} = \{1,1,3,3\}.$$

Наступне найменше значення коефіцієнту цільової функції $c_{13} = 5$. Тому присвоювати значення будемо змінній x_{13} . Найбільший елемент із $\tilde{G}$ після попереднього вибору – це 3. Отже, далі матимемо $x_{13} = 3$.

$G_B = \{1,2,2,2,3,3\}$ – множина використаних елементів з G .

$\tilde{G} = \{1,1,3\}$, де $\tilde{G} = G - G_B$.

$x_{11} + x_{13} = 4, x_{11} = 4 - 3 = 1 \in \{1,1,3\}$, тоді $\tilde{G} = \{1,3\}$, $x_{13} + x_{33} = 4$,

$x_{33} = 4 - 3 = 1 \in \{1,3\}$, тоді $\tilde{G} = \{3\}$, $x_{11} + x_{31} = 4, x_{31} = 4 - 1 = 3 \in \{3\}$

Отже, $x_{13} = 3, x_{11} = 1, x_{33} = 1, x_{31} = 3$ (табл.3.10).

Крок 25. На екрані: Так як всі клітинки заповнені, то обчислюємо значення цільової функції:

$$F_2 = c_{12} \cdot x_{12} + c_{21} \cdot x_{21} + c_{23} \cdot x_{23} + c_{22} \cdot x_{22} + c_{32} \cdot x_{32} + c_{13} \cdot x_{13} + c_{11} \cdot x_{11} + c_{33} \cdot x_{33} + c_{31} \cdot x_{31} = 29 + 5 \cdot 3 + 7 \cdot 1 + 9 \cdot 1 + 8 \cdot 3 = 84 = F_0$$

Крок 26. На екрані: **Рівень 1.2.2.2.** З попереднього кроку маємо

$$G_B = \{1,2,2,2,3\}. \tilde{G} = \{1,1,3,3\}.$$

Таблиця 3.10 – Присвоєння змінних на рівні 1.2.2.1.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1	1	3	3	7	0
A_2	2	1	2	5	0
A_3	3	2	1	6	0
b_j	6	6	6		
Залишок потреб після заповнення клітин	0	0	0		

Наступне найменше значення коефіцієнту цільової функції $c_{13} = 5$. Тому присвоювати значення будемо змінній x_{13} . Найбільший елемент із $\tilde{G}$ після попереднього вибору – це 1. Отже, далі матимемо $x_{13} = 1$.

$G_B = \{1,1,2,2,2,3\}$ – множина використаних елементів з G .

$\tilde{G} = \{1,3,3\}$, де $\tilde{G} = G - G_B$.

$x_{11} + x_{13} = 4, x_{11} = 4 - 1 = 3 \in \{1,3,3\}$, тоді $\tilde{G} = \{1,3\}$,

$x_{13} + x_{33} = 4, x_{33} = 4 - 1 = 3 \in \{1,3\}$, тоді $\tilde{G} = \{1\}$,

$x_{11} + x_{31} = 4, x_{31} = 4 - 3 = 1 \in \{1\}$

Отже, $x_{13} = 1, x_{11} = 3, x_{33} = 3, x_{31} = 1$. (табл.3.11).

Крок 27. На екрані:

Так як всі клітинки заповнені, то обчислюємо значення цільової функції:

$$F_3 = c_{12} \cdot x_{12} + c_{21} \cdot x_{21} + c_{23} \cdot x_{23} + c_{22} \cdot x_{22} + c_{32} \cdot x_{32} + c_{13} \cdot x_{13} + c_{11} \cdot x_{11} + c_{33} \cdot x_{33} + c_{31} \cdot x_{31} = 29 + 5 \cdot 1 + 7 \cdot 3 + 9 \cdot 3 + 8 \cdot 1 = 90 > F_0.$$

Тому цей розв'язок відсікаємо.

Таблиця 3.11 – Присвоєння змінних на рівні 1.2.2.2.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1	3	3	1	7	0
A_2	2	1	2	5	0
A_3	1	2	3	6	0
b_j	6	6	6		
Залишок потреб після заповнення клітин	0	0	0		

Крок 28. На екрані: *Рівень 1.2.3.* Наступне найменше значення коефіцієнту цільової функції $c_{23} = 3$. Тому присвоювати значення будемо змінній x_{23} . Найбільший елемент із $\tilde{G}$ після попереднього вибору – це 1. Отже, далі матимемо $x_{23} = 1$ (табл.3.12).

$G_B = \{1,2,3\}$ – множина використаних елементів з G .

$\tilde{G} = \{1,1,2,2,3,3\}$, де $\tilde{G} = G - G_B$.

$$x_{11} + x_{13} = 4, x_{22} + x_{23} = 3, x_{22} = 3 - 1 = 2, x_{31} + x_{32} + x_{33} = 6,$$

$$x_{11} + x_{31} = 4, x_{22} + x_{32} = 3, x_{32} = 3 - 2 = 1, x_{13} + x_{23} + x_{33} = 6,$$

$$x_{13} + x_{33} = 6 - 1 = 5. \text{ Отримаємо } x_{23} = 1, x_{22} = 2 \text{ та } x_{32} = 1. \text{ (табл.3.13).}$$

Крок 29. На екрані: Обчислюємо оцінки: для присвоєних значень x_{12} , x_{21} , x_{23} , x_{22} та x_{32}

$$\begin{aligned} v &= c_{12} \cdot x_{12} + c_{21} \cdot x_{21} + c_{23} \cdot x_{23} + c_{22} \cdot x_{22} + c_{32} \cdot x_{32} = \\ &= 1 \cdot 3 + 2 \cdot 2 + 3 \cdot 1 + 4 \cdot 2 + 6 \cdot 1 = 24. \end{aligned}$$

Таблиця 3.12 – Присвоєння x_{23} на рівні 1.2.3.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		3		7	4
A_2	2		1	5	2
A_3				6	6
b_j	6	6	6		
Залишок потреб після заповнення клітин	4	3	5		

Таблиця 3.13 – Присвоєння змінних на рівні 1.2.3.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		3		7	4
A_2	2	2	1	5	0
A_3		1		6	5
b_j	6	6	6		
Залишок потреб після заповнення клітин	4	0	5		

$$\xi = v + c^*,$$

яка обчислюється відповідно до відсортованих та обраних c_{ij} та g_t

c_{ij} : обрані 1,2,3,4,6; інші відсортовані: $5 \leq 7 \leq 8 \leq 9$.

g_t : обрані 3,2,1,2,1; інші відсортовані $3 \geq 3 \geq 2 \geq 1$.

Тобто $\xi = 24 + 5 \cdot 3 + 7 \cdot 3 + 8 \cdot 2 + 9 \cdot 1 = 85 > F_0$.

Тому цей розв'язок відсікаємо.

Крок 30. На екрані:

Рівень 1.3. Присвоєне значення змінній на рівні 1 – $x_{12} = 3$. Наступне найменше значення коефіцієнту цільової функції $c_{21} = 2$. Тому присвоювати

значення будемо змінній x_{21} . Найбільший елемент із $\tilde{G}$ після попереднього вибору – це 1. Отже, далі матимемо $x_{21} = 1$.

$G_B = \{1,3\}$ – множина використаних елементів з G .

$\tilde{G} = \{1,1,2,2,2,3,3\}$, де $\tilde{G} = G - G_B$.

$$x_{11} + x_{13} = 4, x_{22} + x_{23} = 5 - 1 = 4, x_{31} + x_{32} + x_{33} = 6,$$

$$x_{11} + x_{31} = 6 - 1 = 5, x_{22} + x_{32} = 3, x_{13} + x_{23} + x_{33} = 6.$$

Результати реалізації рівня 1.3 представлено в таблиці 3.14.

Таблиця 3.14 – Присвоєння x_{21} на рівні 1.3.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		3		7	4
A_2	1			5	4
A_3				6	6
b_j	6	6	6		
Залишок потреб після заповнення клітин	5	3	6		

Крок 31. На екрані: Обчислюємо оцінки:

для присвоєних значень x_{12} та x_{21} $v = c_{12} \cdot x_{12} + c_{21} \cdot x_{21} = 1 \cdot 3 + 2 \cdot 1 = 5$.

$$\xi = v + c^*,$$

яка обчислюється відповідно до відсортованих c_{ij} та g_t

c_{ij} : обрані 1,2; інші відсортовані $3 \leq 4 \leq 5 \leq 6 \leq 7 \leq 8 \leq 9$.

g_t : обрані 3,1; інші відсортовані $3 \geq 3 \geq 2 \geq 2 \geq 2 \geq 1 \geq 1$.

Тобто $\xi = 5 + 3 \cdot 3 + 4 \cdot 3 + 5 \cdot 2 + 6 \cdot 2 + 7 \cdot 2 + 8 \cdot 1 + 9 \cdot 1 = 79$.

Крок 32. На екрані:

Рівень 1.3.1. Наступне найменше значення коефіцієнту цільової функції $c_{23} = 3$. Тому присвоювати значення будемо змінній x_{23} . Найбільший елемент із $\tilde{G}$ після попереднього вибору – це 3. Отже, далі матимемо $x_{23} = 3$.

$G_B = \{1,3,3\}$ – множина використаних елементів з G .

$\tilde{G} = \{1,1,2,2,2,3\}$, де $\tilde{G} = G - G_B$.

$$x_{11} + x_{13} = 4, x_{22} + x_{23} = 4, x_{22} = 4 - 3 = 1, x_{31} + x_{32} + x_{33} = 6,$$

$$x_{11} + x_{31} = 5, x_{22} + x_{32} = 3, x_{32} = 3 - 1 = 2, x_{13} + x_{23} + x_{33} = 6,$$

$$x_{13} + x_{33} = 6 - 3 = 3. \text{ Отже, } x_{23} = 3, x_{22} = 1, x_{32} = 2.$$

Крок 33. На екрані:

На цьому етапі таблиця 3.14 змінюється на таблицю 3.15.

Таблиця 3.15 – Присвоєння x_{21} на рівні 1.3.1.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		3		7	4
A_2	1	1	3	5	0
A_3		2		6	4
b_j	6	6	6		
Залишок потреб після заповнення клітин	5	0	3		

Обчислюємо оцінки: для присвоєних значень $x_{12}, x_{21}, x_{23}, x_{22}, x_{32}$

$$\begin{aligned} v &= c_{12} \cdot x_{12} + c_{21} \cdot x_{21} + c_{23} \cdot x_{23} + c_{22} \cdot x_{22} + c_{32} \cdot x_{32} \\ &= 1 \cdot 3 + 2 \cdot 1 + 3 \cdot 3 + 4 \cdot 1 + 6 \cdot 2 = 30. \\ \xi &= v + c^*, \end{aligned}$$

яка обчислюється відповідно до відсортованих c_{ij} та g_t

c_{ij} : обрані 1, 2, 3, 4, 6; інші відсортовані $5 \leq 7 \leq 8 \leq 9$.

g_t : обрані 3, 1, 3, 1, 2; інші відсортовані $3 \geq 2 \geq 2 \geq 1$.

$$\text{Тобто } \xi = 30 + 5 \cdot 3 + 7 \cdot 2 + 8 \cdot 2 + 9 \cdot 1 = 84 = F_0.$$

Далі значення цільової функції буде збільшуватись, тому можна цей розв'язок відсікати.

Крок 34. На екрані: **Рівень 1.3.2.** Наступне найменше значення коефіцієнту цільової функції $c_{23} = 3$. Тому присвоювати значення будемо змінній x_{23} . Найбільший елемент із $\tilde{G}$ після 3 – це 2. Отже, далі матимемо $x_{23} = 2$.

$G_B = \{1, 2, 3\}$ – множина використаних елементів з G (табл.3.16).

$\tilde{G} = \{1,1,2,2,3,3\}$, де $\tilde{G} = G - G_B$.

$$x_{11} + x_{13} = 4, x_{22} + x_{23} = 4, x_{22} = 4 - 2 = 2, x_{31} + x_{32} + x_{33} = 6,$$

$$x_{11} + x_{31} = 5, x_{22} + x_{32} = 3, x_{32} = 3 - 2 = 1, x_{13} + x_{23} + x_{33} = 6,$$

$$x_{13} + x_{33} = 6 - 2 = 4.$$

Отже, $x_{23} = 2, x_{22} = 2, x_{32} = 1$ (табл.3.17).

Таблиця 3.16 – Присвоєння x_{23} на рівні 1.3.2.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		3		7	4
A_2	1		2	5	2
A_3				6	6
b_j	6	6	6		
Залишок потреб після заповнення клітин	5	3	4		

Таблиця 3.17 – Присвоєння змінних на рівні 1.3.2.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		3		7	4
A_2	1	2	2	5	0
A_3		1		6	5
b_j	6	6	6		
Залишок потреб після заповнення клітин	5	0	4		

Крок 35. На екрані: Обчислюємо оцінки:

для присвоєних значень $x_{12}, x_{21}, x_{23}, x_{22}, x_{32}$

$$\begin{aligned} v &= c_{12} \cdot x_{12} + c_{21} \cdot x_{21} + c_{23} \cdot x_{23} + c_{22} \cdot x_{22} + c_{32} \cdot x_{32} \\ &= 1 \cdot 3 + 2 \cdot 1 + 3 \cdot 2 + 4 \cdot 2 + 6 \cdot 1 = 25. \\ \xi &= v + c^*, \end{aligned}$$

яка обчислюється відповідно до відсортованих c_{ij} та g_t

c_{ij} : обрані 1,2,3,4,6; інші відсортовані $5 \leq 7 \leq 8 \leq 9$.

g_t : обрані 3,1,2,2,1; інші відсортовані $3 \geq 3 \geq 2 \geq 1$.

Тобто $\xi = 25 + 5 \cdot 3 + 7 \cdot 3 + 8 \cdot 2 + 9 \cdot 1 = 86 > F_0$.

Цей розв'язок відсікається.

Крок 36. На екрані:

Рівень 1.3.3. Наступне найменше значення коефіцієнту цільової функції $c_{23} = 3$. Тому присвоювати значення будемо змінній x_{23} . Найбільший елемент із $\tilde{G}$ після 2 – це 1. Отже, далі матимемо $x_{23} = 1$.

$G_B = \{1,1,3\}$ – множина використаних елементів з G .

$\tilde{G} = \{1,2,2,2,3,3\}$, де $\tilde{G} = G - G_B$.

$x_{11} + x_{13} = 4, x_{22} + x_{23} = 4, x_{22} = 4 - 1 = 3, x_{31} + x_{32} + x_{33} = 6,$

$x_{11} + x_{31} = 5, x_{22} + x_{32} = 3, x_{32} = 3 - 3 = 0 \notin \tilde{G}$.

Тому розв'язок відсікається.

Крок 37. На екрані: **Рівень 2** виконайте самостійно, заповнюючи послідовно клітини згідно алгоритму (табл.3.18).

Таблиця 3.18 – Результат введення на рівні 2.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		2		7	5
A_2				5	5
A_3				6	6
b_j	6	6	6		
Залишок потреб після заповнення клітин	6	4	6		

Оцінки: $\nu =$, $\xi =$.

Інформація для повідомлення в разі помилки:

Рівень 2. Найменший коефіцієнт цільової функції (вартість перевезення) $c_{12} = 1$. Тому присвоювати значення будемо змінній x_{12} . Наступний

найбільший елемент із G після 3 - це 2. Отже, на початку другого рівня матимемо $x_{12} = 2$.

$G_B = \{2\}$ – множина використаних елементів з G .

$\tilde{G} = \{1,1,1,2,2,3,3,3\}$, де $\tilde{G} = G - G_B$.

Оцінки: для присвоєного значення x_{12} : $\nu = c_{12} \cdot x_{12} = 1 \cdot 2 = 2$.

$\xi = \nu + c^*$, яка обчислюється відповідно до відсортованих c_{ij} та g_t

c_{ij} : **обране 1**, інші відсортовані $2 \leq 3 \leq 4 \leq 5 \leq 6 \leq 7 \leq 8 \leq 9$.

g_t : **обране 2**, інші відсортовані $3 \geq 3 \geq 3 \geq 2 \geq 2 \geq 1 \geq 1 \geq 1$.

Тобто $\xi = 2 + 2 \cdot 3 + 3 \cdot 3 + 4 \cdot 3 + 5 \cdot 2 + 6 \cdot 2 + 7 \cdot 1 + 8 \cdot 1 + 9 \cdot 1 = 75$.

$\nu = 2, \xi = 75$.

Крок 38. На екрані: **Рівень 2.1**. Присвоєно значення змінної $x_{12} = 2$.

Введіть значення у комірку таблиці згідно алгоритму та обчисліть отримані при цьому оцінки (табл.3.19).

Таблиця 3.19 – Результат введення на рівні 2.1.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		2		7	5
A_2	3			5	2
A_3				6	6
b_j	6	6	6		
Залишок потреб після заповнення клітин	3	4	6		

Оцінки: $\nu =$, $\xi =$.

Примітка: у табл.2 і в наступних, червоним позначено те, що відображено, а синім – те, що має ввести користувач, обравши самостійно комірку для введення.

Інформація для повідомлення в разі помилки:

Рівень 2.1. Наступне найменше значення коефіцієнту цільової функції $c_{21} = 2$. Тому присвоювати значення будемо змінній x_{21} . Найбільший елемент із $\tilde{G}$ після попереднього вибору – це 3. Отже, далі матимемо $x_{21} = 3$.

$G_B = \{2,3\}$ – множина використаних елементів з G .

$\tilde{G} = \{1,1,1,2,2,3,3\}$, де $\tilde{G} = G - G_B$.

Оцінки:

для присвоєного значення x_{21} : $\nu = c_{12} \cdot x_{12} + c_{21} \cdot x_{21} = 1 \cdot 2 + 2 \cdot 3 = 8$.

$\xi = \nu + c^*$, яка обчислюється відповідно до відсортованих c_{ij} та g_t

c_{ij} : обрані 1,2, інші відсортовані $3 \leq 4 \leq 5 \leq 6 \leq 7 \leq 8 \leq 9$.

g_t : обрані 2,3 інші відсортовані $3 \geq 3 \geq 2 \geq 2 \geq 1 \geq 1 \geq 1$.

Тобто $\xi = 8 + 3 \cdot 3 + 4 \cdot 3 + 5 \cdot 2 + 6 \cdot 2 + 7 \cdot 1 + 8 \cdot 1 + 9 \cdot 1 = 75$.

$\nu = 8, \xi = 75$.

Крок 39. На екрані: **Рівень 2.1.1.** Введіть значення у комірку таблиці згідно алгоритму (табл.3.20).

Таблиця 3.20 – Результат введення на рівні 2.1.1.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		2		7	5
A_2	3		3	5	-1
A_3				6	6
b_j	6	6	6		
Залишок потреб після заповнення клітин	3	4	3		

Інформація для повідомлення в разі помилки:

Рівень 2.1.1. Наступне найменше значення коефіцієнту цільової функції $c_{23} = 3$. Тому присвоювати значення будемо змінній x_{23} . Найбільший елемент із $\tilde{G}$ після попереднього вибору – це 3. Отже, далі матимемо $x_{23} = 3$.

$G_B = \{2,3,3\}$ – множина використаних елементів з G .

$\tilde{G} = \{1,1,1,2,2,3\}$, де $\tilde{G} = G - G_B$.

$x_{22} + x_{23} = 2$, $x_{22} = 2 - 3 = -1 < 0$, тобто для $x_{21} = 3$ та $x_{23} = 3$ не виконується умова $x_{21} + x_{22} + x_{23} = 5$, отже, відсікаємо $x_{23} = 3$.

Крок 40. На екрані: **Рівень 2.1.2.**

Введіть значення у комірку таблиці згідно алгоритму (табл.3.21).

Таблиця 3.21 – Результат введення на рівні 2.1.2.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		2		7	5
A_2	3		2	5	0
A_3				6	6
b_j	6	6	6		
Залишок потреб після заповнення клітин	3	4	4		

Інформація для повідомлення в разі помилки:

Рівень 2.1.2. Наступне найменше значення коефіцієнту цільової функції $c_{23} = 3$. Тому присвоювати значення будемо змінній x_{23} . Наступний найбільший елемент із $\tilde{G}$ після 3 – це 2. Отже, далі матимемо $x_{23} = 2$.

$G_B = \{2,2,3\}$ – множина використаних елементів з G .

$\tilde{G} = \{1,1,1,2,3,3\}$, де $\tilde{G} = G - G_B$.

$x_{22} + x_{23} = 2$, $x_{22} = 2 - 2 = 0$,

але цього елемента немає в $\tilde{G}$, отже, відсікаємо $x_{23} = 2$.

Крок 41. На екрані: **Рівень 2.1.3.** Введіть значення у комірки таблиці згідно алгоритму (табл.3.22).

Таблиця 3.22 – Результат введення на рівні 2.1.3.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		2		7	5
A_2	3		1	5	1
A_3				6	6
b_j	6	6	6		
Залишок потреб після заповнення клітин	3	4	5		

Інформація для повідомлення в разі помилки:

Рівень 2.1.3. Наступне найменше значення коефіцієнту цільової функції $c_{23} = 3$. Тому присвоювати значення будемо змінній x_{23} . Наступний найбільший елемент із $\tilde{G}$ після 2 – це 1. Отже, далі матимемо $x_{23} = 1$.

$G_B = \{1,2,3\}$ – множина використаних елементів з G .

$\tilde{G} = \{1,1,2,2,3,3\}$, де $\tilde{G} = G - G_B$.

$x_{22} + x_{23} = 2, x_{22} = 2 - 1 = 1 \in \{1,1,2,2,3,3\}$, тоді $\tilde{G} = \{1,2,2,3,3\}$,

$x_{22} + x_{32} = 4, x_{32} = 4 - 1 = 3 \in \{1,2,2,3,3\}$, тоді $\tilde{G} = \{1,2,2,3\}$,

Отже, $x_{23} = 1, x_{22} = 1, x_{32} = 3$ (табл.3.23).

Таблиця 3.23 – Результат після введення на рівні 2.1.3.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		2		7	5
A_2	3	1	1	5	0
A_3		3		6	3
b_j	6	6	6		
Залишок потреб після заповнення клітин	3	0	3		

Крок 42. На екрані: Обчисліть оцінки: $\nu =$, $\xi =$.

Інформація для повідомлення в разі помилки:

Оцінки: для присвоєного значення x_{21} :

$$\begin{aligned} \nu &= c_{12} \cdot x_{12} + c_{21} \cdot x_{21} + c_{23} \cdot x_{23} + c_{22} \cdot x_{22} + c_{32} \cdot x_{32} = \\ &= 1 \cdot 2 + 2 \cdot 3 + 3 \cdot 1 + 4 \cdot 1 + 6 \cdot 3 = 33. \end{aligned}$$

$\xi = \nu + c^*$, яка обчислюється відповідно до відсортованих c_{ij} та g_t

c_{ij} : обрані 1,2,3, 4,6, інші відсортовані $5 \leq 7 \leq 8 \leq 9$.

g_t : обрані 2, 3, 1, 1, 3, інші відсортовані $3 \geq 2 \geq 2 \geq 1$.

Тобто $\xi = 33 + 5 \cdot 3 + 7 \cdot 2 + 8 \cdot 2 + 9 \cdot 1 = 87 > F_0 = 84$.

Тому $x_{23} = 1, x_{22} = 1, x_{32} = 3$ відсікаємо.

Крок 43. На екрані: **Рівень 2.2.** Присвоєно значення змінної $x_{12} = 2$.

Введіть значення у комірку таблиці згідно алгоритму та обчисліть отримані при цьому оцінки (табл.3.24).

Таблиця 3.24 – Результат введення на рівні 2.2.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		2		7	5
A_2	2			5	3
A_3				6	6
b_j	6	6	6		
Залишок потреб після заповнення клітин	4	4	6		

Оцінки: $\nu =$, $\xi =$.

Інформація для повідомлення в разі помилки:

Наступне найменше значення коефіцієнту цільової функції $c_{21} = 2$. Тому присвоювати значення будемо змінній x_{21} . Найбільший елемент із $\tilde{G}$ після 3 – це 2. Отже, далі матимемо $x_{21} = 2$.

$G_B = \{2,2\}$ – множина використаних елементів з G .

$\tilde{G} = \{1,1,1,2,3,3,3\}$, де $\tilde{G} = G - G_B$.

Оцінки: для присвоєного значення x_{21} :

$$\nu = c_{12} \cdot x_{12} + c_{21} \cdot x_{21} = 1 \cdot 2 + 2 \cdot 2 = 6.$$

$\xi = \nu + c^*$, яка обчислюється відповідно до відсортованих c_{ij} та g_t

c_{ij} : обрані 1,2, інші відсортовані $3 \leq 4 \leq 5 \leq 6 \leq 7 \leq 8 \leq 9$.

g_t : обрані 2,2, інші відсортовані $3 \geq 3 \geq 3 \geq 2 \geq 1 \geq 1 \geq 1$.

Тобто $\xi = 6 + 3 \cdot 3 + 4 \cdot 3 + 5 \cdot 3 + 6 \cdot 2 + 7 \cdot 1 + 8 \cdot 1 + 9 \cdot 1 = 78$.

$$\nu = 6, \xi = 78.$$

Крок 44. На екрані: **Рівень 2.2.1.**

Введіть значення у комірки таблиці згідно алгоритму (табл.3.25).

Таблиця 3.25 – Результат введення на рівні 2.2.1.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		2		7	5
A_2	2		3	5	0
A_3				6	6
b_j	6	6	6		
Залишок потреб після заповнення клітин	4	4	3		

Інформація для повідомлення в разі помилки:

Наступне найменше значення коефіцієнту цільової функції $c_{23} = 3$. Тому присвоювати значення будемо змінній x_{23} . Наступний найбільший елемент із $\tilde{G}$ – це 3. Отже, далі матимемо $x_{23} = 3$.

$G_B = \{2,2,3\}$ – множина використаних елементів з G .

$\tilde{G} = \{1,1,1,2,3,3\}$, де $\tilde{G} = G - G_B$. $x_{22} + x_{23} = 3, x_{22} = 3 - 3 = 0 \tilde{G}$.

Відсікаємо $x_{23} = 3$.

Крок 45. На екрані: **Рівень 2.2.2.** Введіть значення у комірки таблиці згідно алгоритму (табл.3.26).

Таблиця 3.26 – Результат після введення на рівні 2.2.2.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		2		7	5
A_2	2		2	5	1
A_3				6	6
b_j	6	6	6		
Залишок потреб після заповнення клітин	4	4	4		

Інформація для повідомлення в разі помилки:

Наступне найменше значення коефіцієнту цільової функції $c_{23} = 3$. Тому присвоювати значення будемо змінній x_{23} . Наступний найбільший елемент із $\tilde{G}$ після 3 – це 2. Отже, далі матимемо $x_{23} = 2$.

$G_B = \{2,2,2\}$ – множина використаних елементів з G .

$\tilde{G} = \{1,1,2,2,3,3\}$, де $\tilde{G} = G - G_B$.

$x_{22} + x_{23} = 3, x_{22} = 3 - 2 = 1 \in \{1,1,2,2,3,3\}$, тоді $\tilde{G} = \{1,2,2,3,3\}$,

$x_{22} + x_{32} = 4, x_{32} = 4 - 1 = 3 \in \{1,2,2,3,3\}$, тоді $\tilde{G} = \{1,2,2,3\}$,

Отже, $x_{23} = 2, x_{22} = 1, x_{32} = 3$ (табл.3.27).

Крок 46. На екрані: Обчисліть оцінки: $\nu =$, $\xi =$.

Інформація для повідомлення в разі помилки:

Оцінки: для присвоєного значення x_{21} :

$$\begin{aligned} \nu &= c_{12} \cdot x_{12} + c_{21} \cdot x_{21} + c_{23} \cdot x_{23} + c_{22} \cdot x_{22} + c_{32} \cdot x_{32} = \\ &= 1 \cdot 2 + 2 \cdot 2 + 3 \cdot 2 + 4 \cdot 1 + 6 \cdot 3 = 34. \end{aligned}$$

$\xi = \nu + c^*$, яка обчислюється відповідно до відсортованих c_{ij} та g_t

c_{ij} : обрані 1,2,3, 4,6, інші відсортовані $5 \leq 7 \leq 8 \leq 9$.

g_t : обрані 2, 2, 2, 1, 3, інші відсортовані $3 \geq 3 \geq 1 \geq 1$.

Таблиця 3.26 – Результат після введення всіх значень на рівні 2.2.2.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		2		7	5
A_2	2	1	2	5	0
A_3		3		6	3
b_j	6	6	6		
Залишок потреб після заповнення клітин	4	0	4		

Тобто $\xi = 34 + 5 \cdot 3 + 7 \cdot 3 + 8 \cdot 1 + 9 \cdot 1 = 87 > F_0 = 84$.

Тому $x_{23} = 2, x_{22} = 1, x_{32} = 3$ відсікаємо.

Крок 47. На екрані: **Рівень 2.2.3.**

Введіть значення у комірки таблиці згідно алгоритму (табл.3.28).

Інформація для повідомлення в разі помилки:

Наступне найменше значення коефіцієнту цільової функції $c_{23} = 3$. Тому присвоювати значення будемо змінній x_{23} . Наступний найбільший елемент із $\tilde{G}$ після 2 – це 1. Отже, далі матимемо $x_{23} = 1$.

Таблиця 3.28 – Результат після введення на рівні 2.2.3.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		2		7	5
A_2	2	2	1	5	0
A_3		2		6	4
b_j	6	6	6		
Залишок потреб після заповнення клітин	4	0	5		

$G_B = \{1,2,2\}$ – множина використаних елементів з G .

$\tilde{G} = \{1,1,2,3,3,3\}$, де $\tilde{G} = G - G_B$.

$x_{22} + x_{23} = 3, x_{22} = 3 - 1 = 2 \in \{1,1,2,3,3,3\}$, тоді $\tilde{G} = \{1,1,3,3,3\}$,

$x_{22} + x_{32} = 4, x_{32} = 4 - 2 = 2 \in \{1,1,3,3,3\}$. Тому цей розв’язок відсікаємо.

Крок 48. На екрані: **Рівень 2.3**. Присвоєно значення змінної $x_{12} = 2$.

Введіть значення у комірку таблиці згідно алгоритму та обчисліть отримані при цьому оцінки (табл.3.29).

Таблиця 3.29 – Результат введення на рівні 2.3.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		2		7	5
A_2	1			5	4
A_3				6	6
b_j	6	6	6		
Залишок потреб після заповнення клітин	5	4	6		

Оцінки: $\nu =$, $\xi =$.

Інформація для повідомлення в разі помилки:

Наступне найменше значення коефіцієнту цільової функції $c_{21} = 2$. Тому присвоювати значення будемо змінній x_{21} . Найбільший елемент із $\tilde{G}$ після 2 – це 1. Отже, далі матимемо $x_{21} = 1$.

$G_B = \{1,2\}$ – множина використаних елементів з G .

$\tilde{G} = \{1,1,2,2,3,3,3\}$, де $\tilde{G} = G - G_B$.

Оцінки: для присвоєного значення x_{21} :

$$v = c_{12} \cdot x_{12} + c_{21} \cdot x_{21} = 1 \cdot 2 + 2 \cdot 1 = 4.$$

$\xi = v + c^*$, яка обчислюється відповідно до відсортованих c_{ij} та g_t

c_{ij} : обрані 1,2, інші відсортовані $3 \leq 4 \leq 5 \leq 6 \leq 7 \leq 8 \leq 9$.

g_t : обрані 2, 1, інші відсортовані $3 \geq 3 \geq 3 \geq 2 \geq 2 \geq 1 \geq 1$.

Тобто $\xi = 4 + 3 \cdot 3 + 4 \cdot 3 + 5 \cdot 3 + 6 \cdot 2 + 7 \cdot 2 + 8 \cdot 1 + 9 \cdot 1 = 83$.

$$v = 4, \xi = 83.$$

Крок 49. На екрані: *Рівень 2.3.1.*

Введіть значення у комірки таблиці згідно алгоритму (табл.3.30).

Таблиця 3.30 – Результат введення на рівні 2.3.1.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		2		7	5
A_2	1		3	5	1
A_3				6	6
b_j	6	6	6		
Залишок потреб після заповнення клітин	5	4	3		

Інформація для повідомлення в разі помилки:

Наступне найменше значення коефіцієнту цільової функції $c_{23} = 3$. Тому присвоювати значення будемо змінній x_{23} . Наступний найбільший елемент із $\tilde{G}$ – це 3. Отже, далі матимемо $x_{23} = 3$.

$G_B = \{1,2,3\}$ – множина використаних елементів з G .

$\tilde{G} = \{1,1,2,2,3,3\}$, де $\tilde{G} = G - G_B$.

$$x_{22} + x_{23} = 4, x_{22} = 4 - 3 = 1 \in \{1,1,2,2,3,3\}, \text{ тоді } \tilde{G} = \{1,2,2,3,3\},$$

$$x_{22} + x_{32} = 4, x_{32} = 4 - 1 = 3 \in \{1,2,2,3,3\}, \text{ тоді } \tilde{G} = \{1,2,2,3\}$$

$$\text{Отже, } x_{23} = 3. x_{22} = 1. x_{32} = 3.$$

Крок 50. На екрані: *Рівень 2.3.1.*

Введіть значення у комірки таблиці згідно алгоритму (табл.3.31).

Таблиця 3.31 – Результат після введення на рівні 2.3.1.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		2		7	5
A_2	1	1	3	5	0
A_3		3		6	3
b_j	6	6	6		
Залишок потреб після заповнення клітин	5	0	3		

Крок 51. На екрані: Обчисліть оцінки: $\nu =$, $\xi =$.

Інформація для повідомлення в разі помилки:

Оцінки: для присвоєного значення x_{21} :

$$\begin{aligned} \nu &= c_{12} \cdot x_{12} + c_{21} \cdot x_{21} + c_{23} \cdot x_{23} + c_{22} \cdot x_{22} + c_{32} \cdot x_{32} = \\ &= 1 \cdot 2 + 2 \cdot 1 + 3 \cdot 3 + 4 \cdot 1 + 6 \cdot 3 = 35. \end{aligned}$$

$\xi = \nu + c^*$, яка обчислюється відповідно до відсортованих c_{ij} та g_t

c_{ij} : обрані 1,2,3,4,6, інші відсортовані $5 \leq 7 \leq 8 \leq 9$.

g_t : обрані 2,1,3,1,3, інші відсортовані $3 \geq 2 \geq 2 \geq 1$.

Тобто $\xi = 35 + 5 \cdot 3 + 7 \cdot 2 + 8 \cdot 2 + 9 \cdot 1 = 89 > F_0 = 84$.

Тому $x_{23} = 3$, $x_{22} = 1$, $x_{32} = 3$ відсікаємо.

Крок 52. На екрані: **Рівень 2.3.2.**

Введіть значення у комірки таблиці згідно алгоритму (табл.3.32).

Таблиця 3.32 – Результат після введення на рівні 2.3.2.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		2		7	5
A_2	1		2	5	2
A_3				6	6
b_j	6	6	6		
Залишок потреб після заповнення клітин	5	4	4		

Інформація для повідомлення в разі помилки:

Наступне найменше значення коефіцієнту цільової функції $c_{23} = 3$. Тому присвоювати значення будемо змінній x_{23} . Наступний найбільший елемент із $\tilde{G}$ після 3 – це 2. Отже, далі матимемо $x_{23} = 2$.

$G_B = \{1, 2, 2\}$ – множина використаних елементів з G .

$\tilde{G} = \{1, 1, 2, 3, 3, 3\}$, де $\tilde{G} = G - G_B$.

$x_{22} + x_{23} = 4, x_{22} = 4 - 2 = 2 \in \{1, 1, 2, 3, 3, 3\}$, тоді $\tilde{G} = \{1, 1, 3, 3, 3\}$,

$x_{22} + x_{32} = 4, x_{32} = 4 - 2 = 2 \notin \{1, 1, 3, 3, 3\}$ (табл.3.33)

Крок 53. На екрані: **Рівень 2.3.3.**

Введіть значення у комірки таблиці згідно алгоритму (табл.3.34).

Таблиця 3.33 – Результат після введення на рівні 2.3.2.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		2		7	5
A_2	1	2	2	5	0
A_3		2		6	4
b_j	6	6	6		
Залишок потреб після заповнення клітин	5	0	4		

Інформація для повідомлення в разі помилки:

Наступне найменше значення коефіцієнту цільової функції $c_{23} = 3$. Тому присвоювати значення будемо змінній x_{23} . Наступний найбільший елемент із $\tilde{G}$ після 2 – це 1. Отже, далі матимемо $x_{23} = 1$.

$G_B = \{1, 1, 2\}$ – множина використаних елементів з G .

$\tilde{G} = \{1, 2, 2, 3, 3, 3\}$, де $\tilde{G} = G - G_B$.

$x_{22} + x_{23} = 4, x_{22} = 4 - 1 = 3 \in \{1, 2, 2, 3, 3, 3\}$, тоді $\tilde{G} = \{1, 2, 2, 3, 3\}$,

$x_{22} + x_{32} = 4, x_{32} = 4 - 3 = 1 \in \{1, 2, 2, 3, 3\}$.

Тоді $x_{23} = 1, x_{22} = 3, x_{32} = 1$ (табл.3.35).

Крок 54. На екрані: Обчисліть оцінки: $\nu =$, $\xi =$.

Таблиця 3.34 – Результат після введення на рівні 2.3.3.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		2		7	5
A_2	1		1	5	3
A_3				6	6
b_j	6	6	6		
Залишок потреб після заповнення клітин	5	4	5		

Таблиця 3.35 – Результат після введення на рівні 2.3.3.

	B_1	B_2	B_3	a_i	Залишок запасів після заповнення клітин
A_1		2		7	5
A_2	1	3	1	5	3
A_3		1		6	6
b_j	6	6	6		
Залишок потреб після заповнення клітин	5	4	5		

Інформація для повідомлення в разі помилки:Оцінки: для присвоєного значення x_{21} :

$$\nu = c_{12} \cdot x_{12} + c_{21} \cdot x_{21} + c_{23} \cdot x_{23} + c_{22} \cdot x_{22} + c_{32} \cdot x_{32} =$$

$$= 1 \cdot 2 + 2 \cdot 1 + 3 \cdot 1 + 4 \cdot 3 + 6 \cdot 1 = 25.$$

 $\xi = \nu + c^*$, яка обчислюється відповідно до відсортованих c_{ij} та g_t c_{ij} : обрані 1,2,3,4,6, інші відсортовані $5 \leq 7 \leq 8 \leq 9$. g_t : обрані 2,1,1,3,1, інші відсортовані $3 \geq 3 \geq 2 \geq 2$.Тобто $\xi = 25 + 5 \cdot 3 + 7 \cdot 3 + 8 \cdot 2 + 9 \cdot 2 = 95 > F_0 = 84$.Тому $x_{23} = 1$, $x_{22} = 3$, $x_{32} = 1$ відсікаємо.

Крок 55. На екрані: Виконайте самостійно рівень 3, після чого запишіть відповідь до заданої задачі: $F_0 =$. Якщо введено правильно значення $F_0 = 84$, то на екрані з'являється: Цей результат дають два плани перевезень:

$$X_1 = (1 \ 3 \ 3 \ 3 \ 1 \ 1 \ 2 \ 2 \ 2) \text{ та } X_2 = (1 \ 3 \ 3 \ 2 \ 1 \ 2 \ 3 \ 2 \ 1).$$

Інформація для повідомлення в разі помилки:

Рівень 3: $x_{12} = 1, \nu = 1, \xi = 81$. Рівень 3.1: $x_{21} = 3, \nu = 7, \xi = 81$.

Рівень 3.1.1: $x_{23} = 3$ відсікається. Рівень 3.1.2: $x_{23} = 2$ відсікається. Рівень 3.1.3: $x_{23} = 1$ відсікається.

Рівень 3.2: $x_{21} = 2, \nu = 5, \xi = 84 = F_0$. Відсікаємо.

Рівень 3.3: $x_{21} = 1, \nu = 3, \xi = 90 = F_0$. Відсікаємо.

Отже, $F_0 = 84$. Цей результат дають два плани перевезень:

$$X_1 = (1 \ 3 \ 3 \ 3 \ 1 \ 1 \ 2 \ 2 \ 2) \text{ та } X_2 = (1 \ 3 \ 3 \ 2 \ 1 \ 2 \ 3 \ 2 \ 1).$$

4. ПРАКТИЧНА ЧАСТИНА

4.1. Блок-схема алгоритму роботи програми

Блок-схема є корисним інструментом для візуалізації алгоритму роботи програми, що робить його більш зрозумілим та легшим для сприйняття.

Далі розглянемо блок-схеми алгоритму розв'язування комбінаторної транспортної задачі та алгоритмів програмної реалізації методу гілок та меж для КТЗ (рис.4.1 - 4.4).

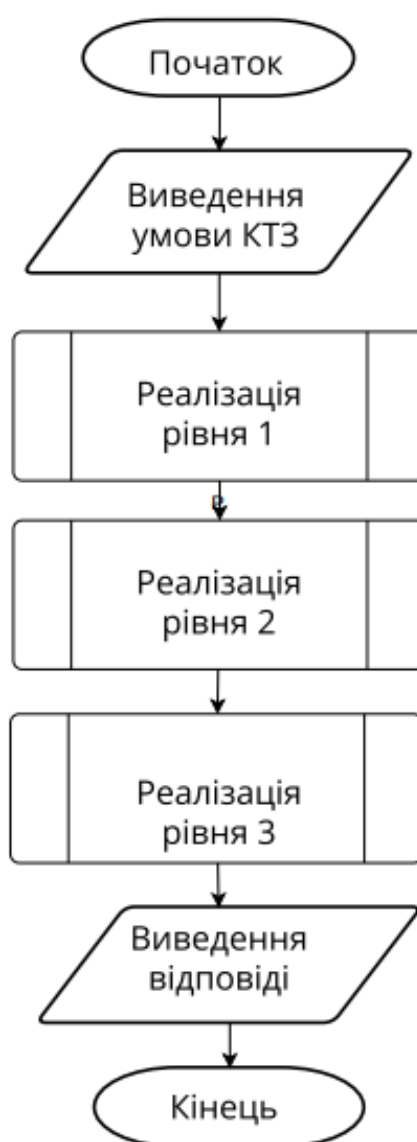


Рисунок 4.1 – Блок-схема алгоритму роботи програми

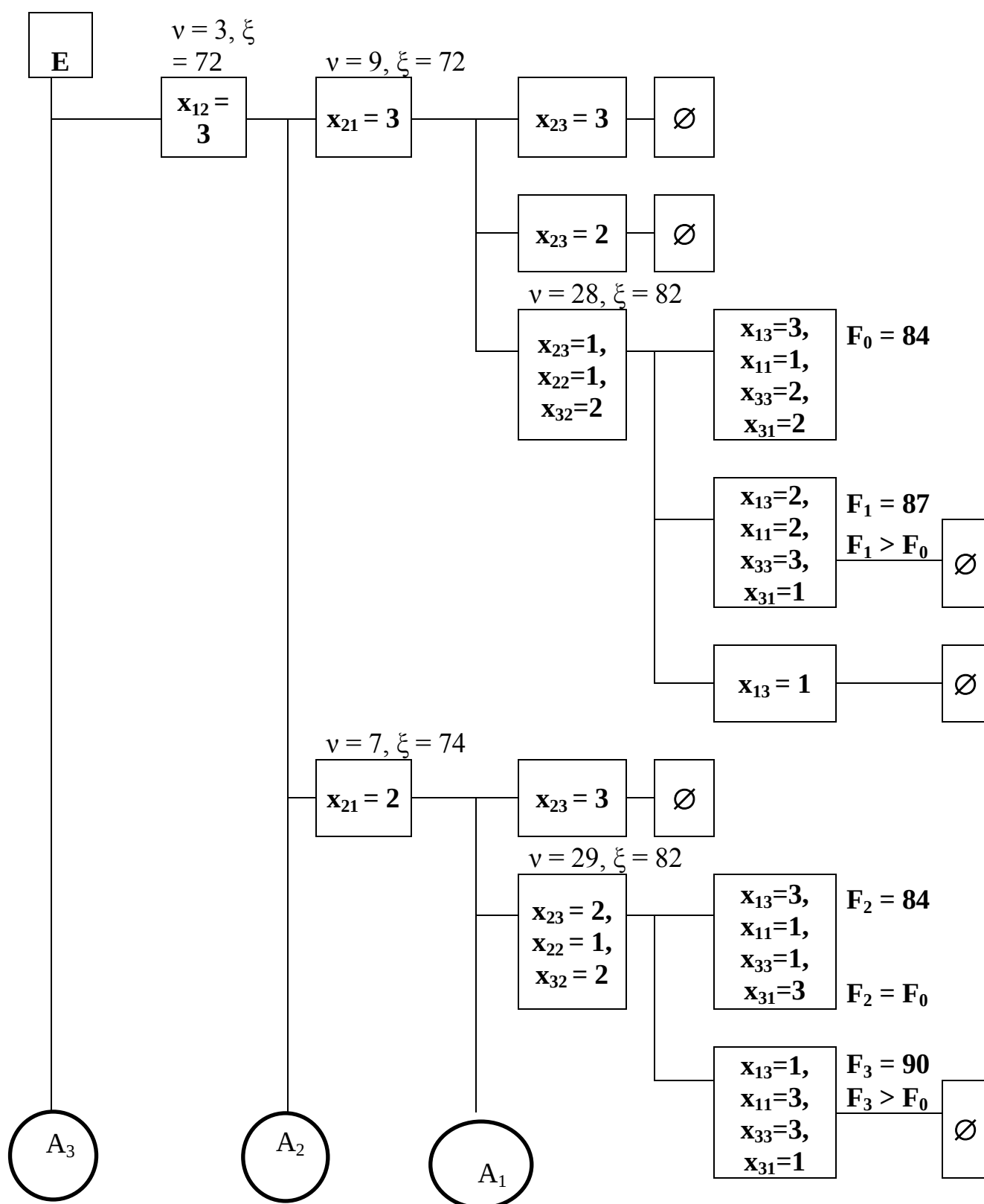


Рисунок 4.2 – Блок-схема розв'язування КТЗ (рівень 1)

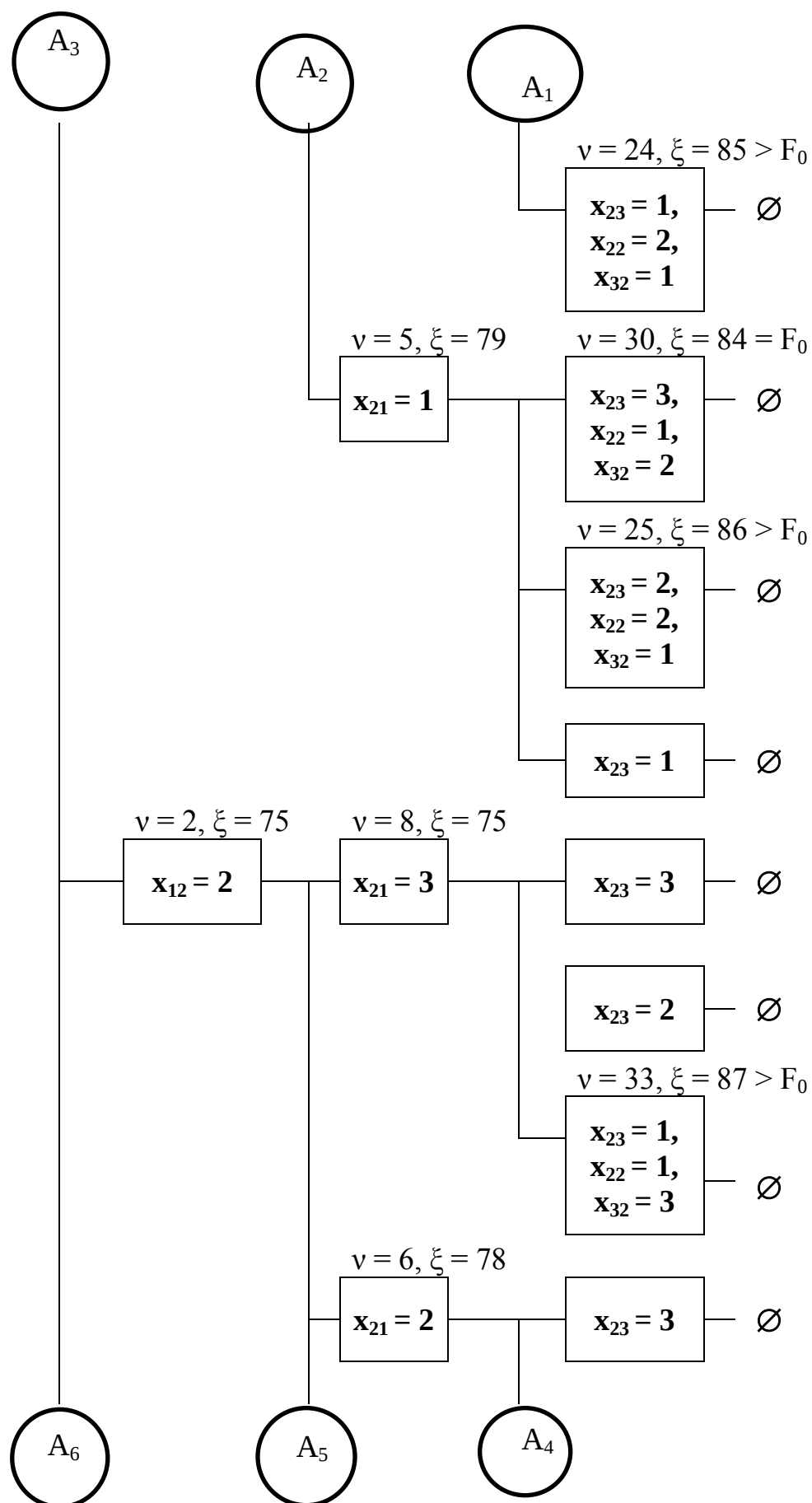


Рисунок 4.3 – Продовження 1 блок-схеми розв'язування КТЗ (рівень 2)

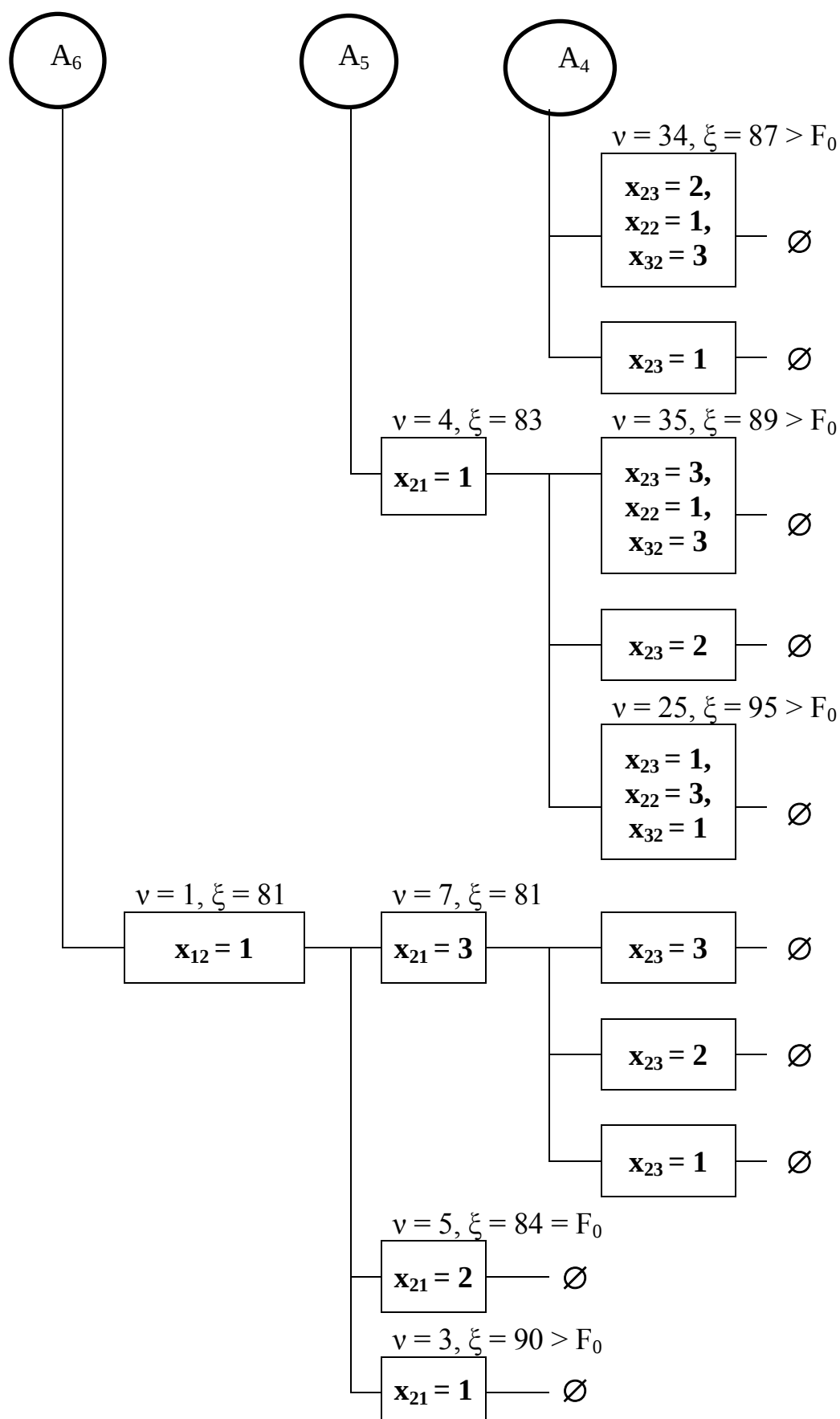


Рисунок 4.4 – Продовження 2 блок-схеми розв'язування КТЗ (рівень 3)

4.2. Опис програмної реалізації

Програма створена з використанням мови програмування Kotlin, фреймворку Compose Desktop та бібліотеки Voyager.

При запуску програми ми маємо простий інтерфейс (рис.4.5).

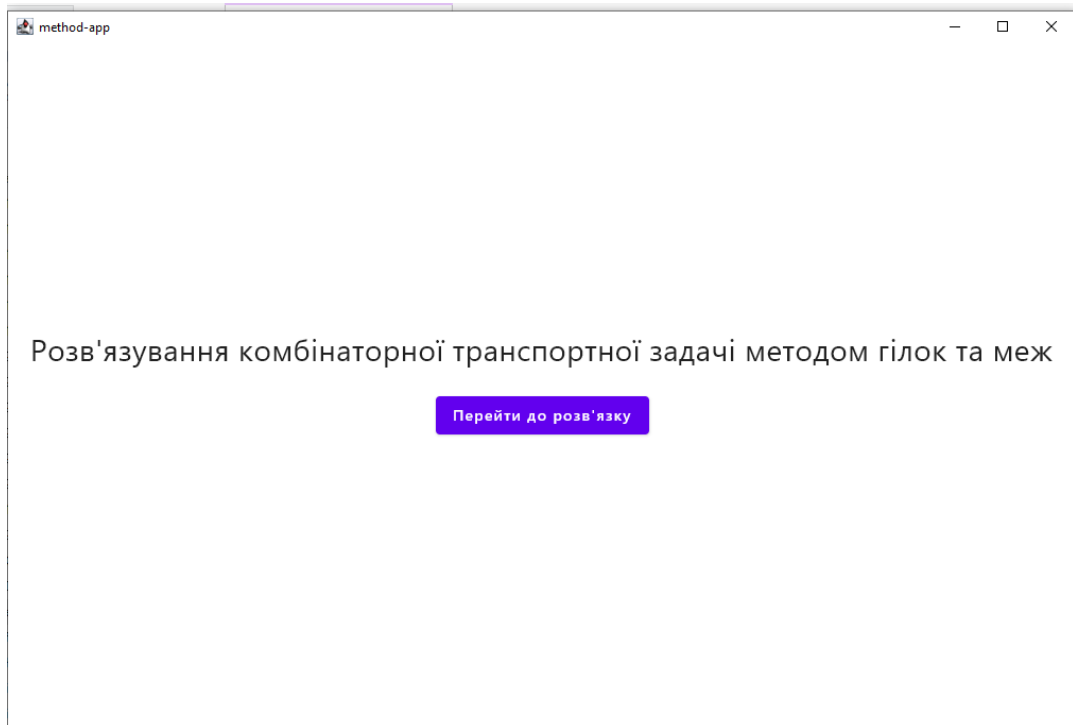


Рисунок 4.5 – Стартова сторінка програми

Після натискання на кнопку виконується функція, яка змінить початковий екран на перший екран розв'язку (рис 4.6).

Кожен крок написано на окремому екрані, кожен екран реалізується на новому класі, клас екрану успадковує Screen, навігація здійснюється натисканням на відповідні кнопки.

Кнопки додаються написаною функцією, вона приймає в якості параметра клас, і при натисканні на кнопку “Наступний крок” відбувається перехід на наступний екран з використанням функції `navigator.push()`, а при натисканні “Попередній крок” ми використовуємо функцію `navigator.pop` яка повертає минулий екран (рис.4.7).

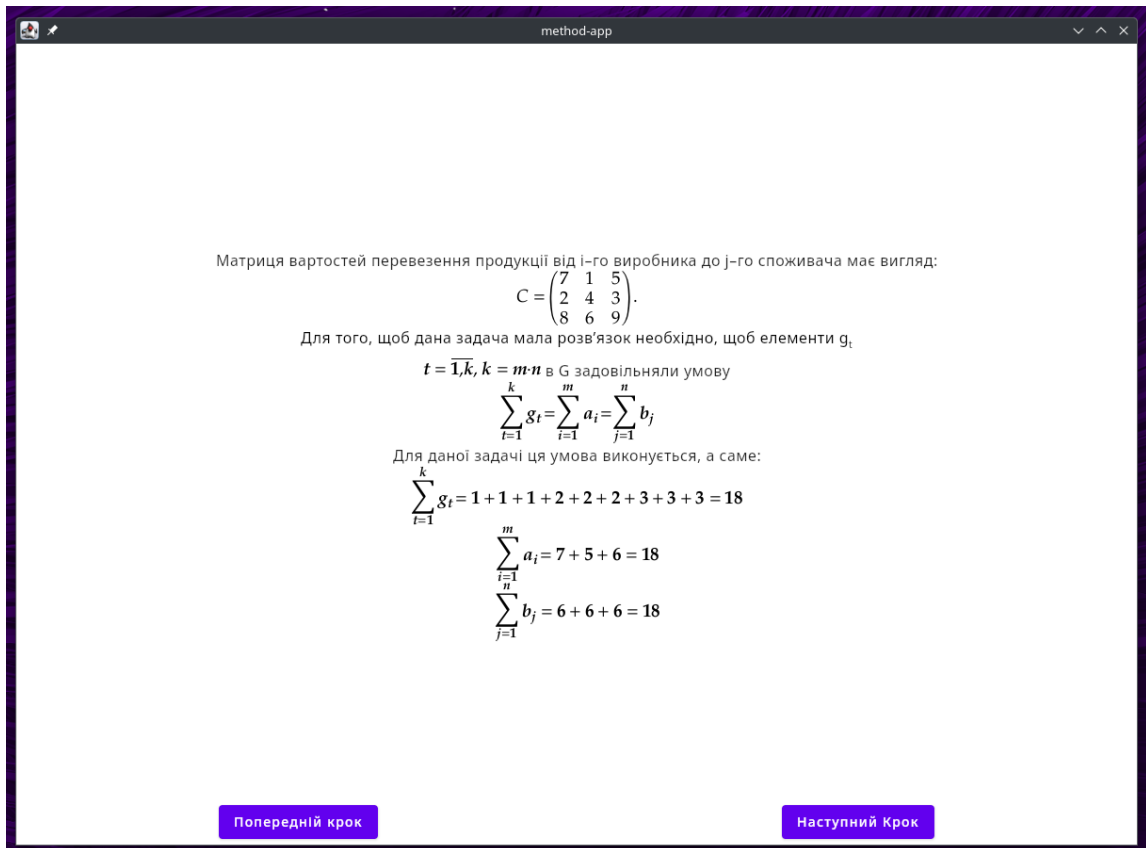


Рисунок 4.6 – Вікно програми на наступному кроці після старту

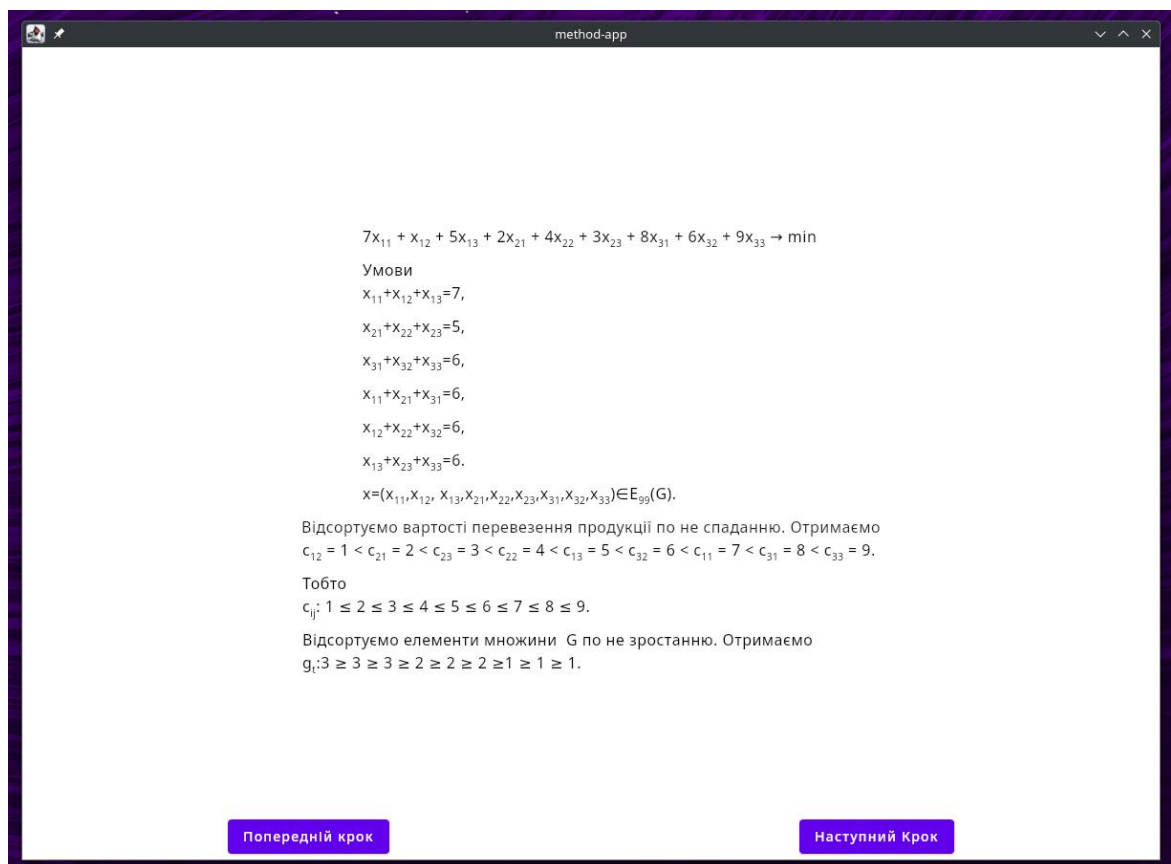


Рисунок 4.7 – Відображення процесу сортування елементів множини тарифів та місткостей

Щоб відображати розв'язок коректно, було написано функцію обробки математичного тексту, при її виклику ми передаємо у якості параметру рядок, функція аналізує зміст отриманого рядка, і у разі якщо знаходить “_n” (замість n – може бути будь-який символ), то ‘n’ стає індексом, а ‘^n’ показником степеню. Функція змінює форматування тексту розміщеного після знаку, його розмір та положення, після чого об'єднує все це в один рядок. У разі якщо форматування не потрібне, використовується звичайна функція Text.

Для зручності розуміння рівнів, було розроблено додатковий функціонал:

Data Клас TreeNode – для створення об'єктів нашого дерева рівнів, він має такі властивості як текст, який буде відображений у дереві, і список нащадкових блоків.

Функція для відображення дерева, їй ми передаємо в якості параметра список елементів типу TreeNode (рис.4.8).

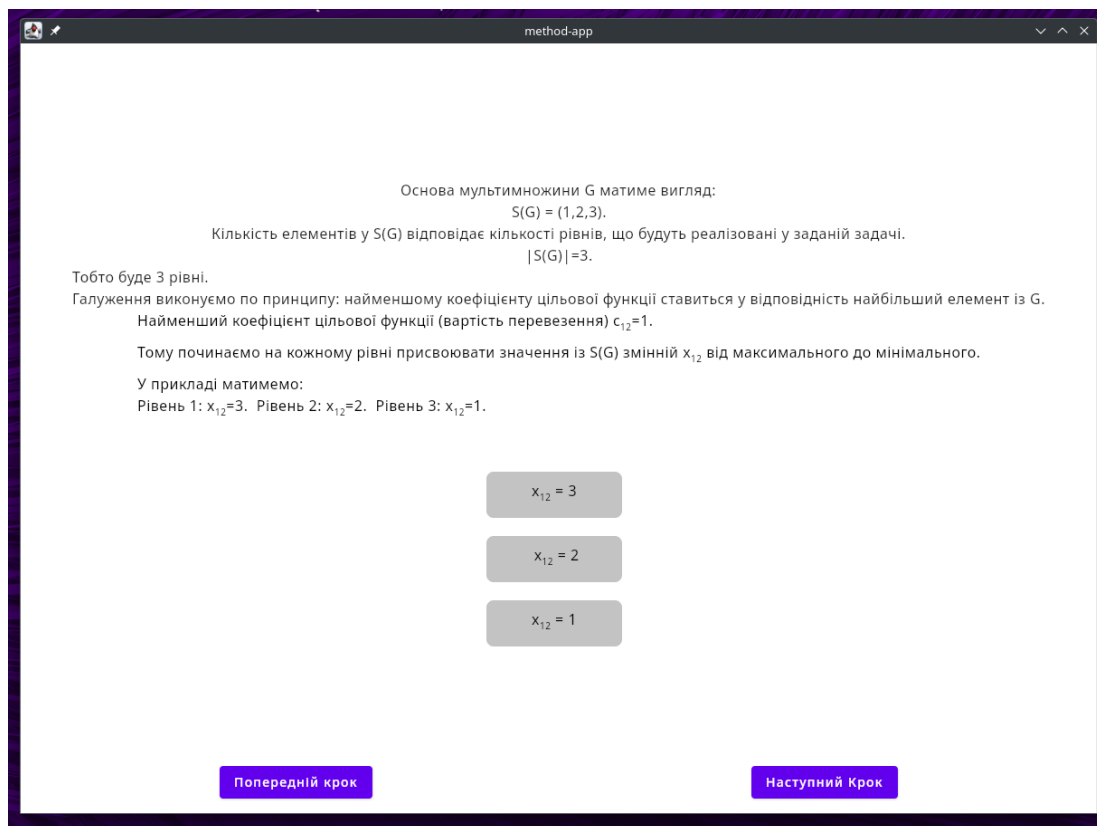


Рисунок 4.8 – Вікно програми з відображенням дерева рівнів

Програма також може відображати таблиці, інформацію необхідно відобразити в таблиці вказується у двох вимірному масиві, кожний елемент масиву відображається у відповідній комірці таблиці. За потреби можна вказувати колір клітинки, можна фарбувати як цілі рядки/стовпці, так і комірки вибірково. Функція створює елементи які вказані, тому якщо необхідно додати можливість введення тексту у таблицю, то ми просто вказуємо замість функції Text використовуємо будь-яку яка створює поле для введення, і функція сама створить потрібну кількість комірок з можливістю введення за заданими умовами (рис.4.9). Все це можна комбінувати на одному екрані (рис.4.10).

Рівень 1. Найменший коефіцієнт цільової функції (вартість перевезення) $c_{12}=1$. Тому присвоювати значення будемо змінній x_{12} . Найбільший елемент із G це 3. Отже, на початку першого рівня матимемо $x_{12}=3$.

$G_8 = \{3\}$ - Множина використаних елементів з G
 $G = \{1,1,1,2,2,2,3,3\}$, де $G = G \cdot G_8$

За умовою
 $x_{11}+x_{12}+x_{13}=7$,
 $x_{21}+x_{22}+x_{23}=5$,
 $x_{31}+x_{32}+x_{33}=6$,
 $x_{11}+x_{21}+x_{31}=6$,
 $x_{12}+x_{22}+x_{32}=6$,
 $x_{13}+x_{23}+x_{33}=6$.

Тоді після підстановки x_{12} маємо (табл. 1):
 $x_{11}+x_{13}=7-3=4$,
 $x_{21}+x_{22}+x_{23}=5$,
 $x_{31}+x_{32}+x_{33}=6$,
 $x_{11}+x_{21}+x_{31}=6$,
 $x_{22}+x_{32}=6-3=3$,
 $x_{13}+x_{23}+x_{33}=6$.

	B_1	B_2	B_3	a_1	Залишок
A_1		3		7	4
A_2				5	5
A_3				6	6
b_1	6	6	6		
Залишок	6	3	6		

Попередній крок Наступний Крок

Рисунок 4.9 – Вікно програми з таблицею

У випадку коли нам потрібно розмістити багато інформації на одному екрані, можна створювати нові вікна для зручності і відображенню інформації за потреби (рис.4.11 - 4.12).

Для відображення дерева рівнів було створено функцію ShowTreeWindow. При її виклику вона створює кнопку яка відкриє вікно з деревом рівнів, а приймає в якості параметра список елементів типу TreeNode.

Рівень 1.1.3. Так як відсікаємо $x_{23}=2$, то розглядаємо наступний найбільший елемент із $\tilde{G}$ – це 1. Отже, далі матимемо $x_{23}=1$.

$G_B=\{1,3,3\}$ – множина використаних елементів з G .

$\tilde{G}=\{1,1,2,2,2,3\}$, де $\tilde{G}=G-G_B$.

Графік рівнів

Таблиця 3. Присвоєння x_{23} на рівні 1.1.3.

	B_1	B_2	B_3	a_i	Залишок
A_1		3		7	4
A_2	3		1	5	1
A_3				6	6
b_j	6	6	6		
Залишок	3	3	5		

Попередній крок Наступний Крок

Рисунок 4.10 – Комбінування відображення графіку та таблиці

Рівень 1.1.3.1. З попереднього кроку маємо

$G_B=\{1,1,2,3,3\}$, $\tilde{G}=\{1,2,2,3\}$.

Наступне найменше значення коефіцієнту цільової функції $c_{13}=5$. Тому присвоювати значення будемо змінній x_{13} .

Найбільший елемент із $\tilde{G}$ після попереднього вибору – це 3. Отже, далі матимемо $x_{13}=3$.

$G_B=\{1,1,2,3,3,3\}$ – множина використаних елементів з G .

$\tilde{G}=\{1,2,2\}$, де $\tilde{G}=G-G_B$.

$x_{11}+x_{13}=4$, $x_{11}=4-3=1 \in \{1,2,2\}$, тоді $\tilde{G}=\{2,2\}$,

$x_{13}+x_{33}=5$, $x_{33}=5-3=2 \in \{2,2\}$, тоді $\tilde{G}=\{2\}$,

$x_{11}+x_{31}=3$, $x_{31}=3-1=2 \in \{2\}$

Отже, $x_{13}=3$, $x_{11}=1$, $x_{33}=2$, $x_{31}=2$.

Таблиця 5. Присвоєння змінних на рівні 1.1.3.1.

	B_1	B_2	B_3	a_i	Залишок
A_1	1	3	3	7	0
A_2	3	1	1	5	0
A_3	2	2	2	6	0
b_j	6	6	6		
Залишок	0	0	0		

Попередній крок Показати графік рівнів Наступний Крок

Рисунок 4.11 – Відображення результатів рівня 1.1.3.1

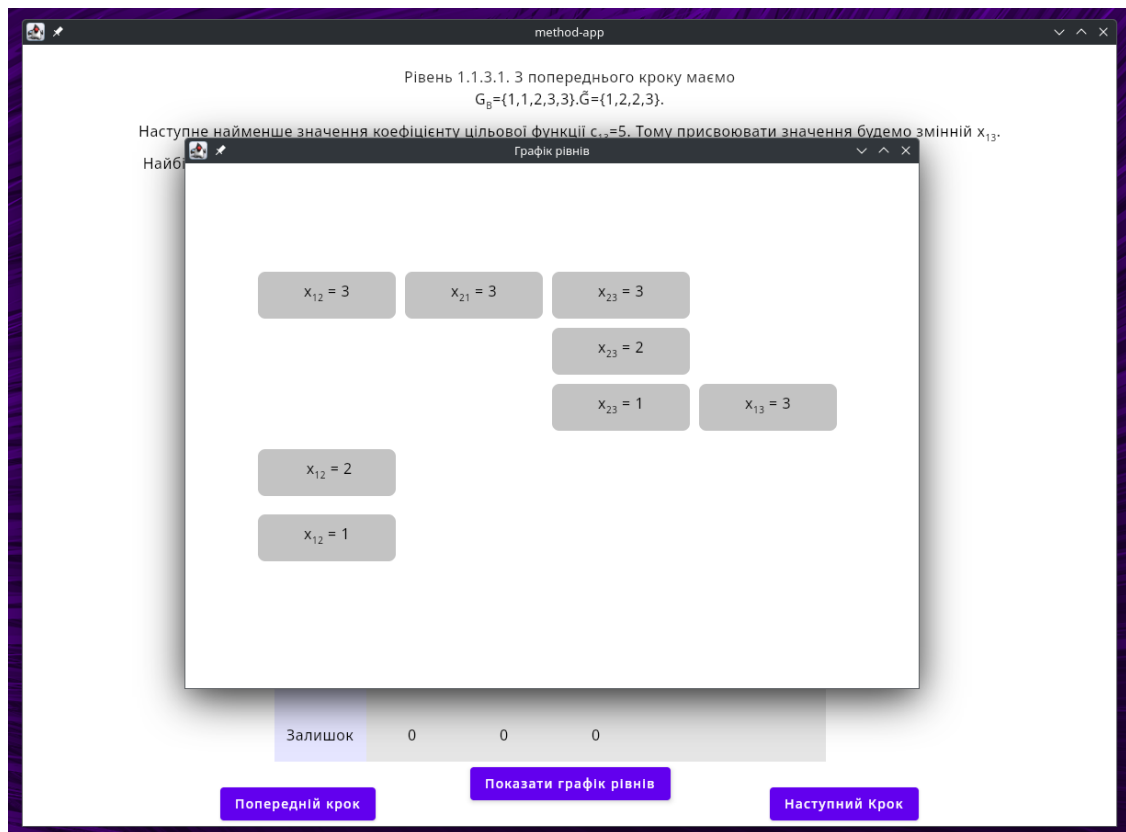


Рисунок 4.12 – Відображення додаткового вікна з графіком рівнів

Таблиці також можна розміщувати в окремих вікнах (рис.4.13 - 4.15).

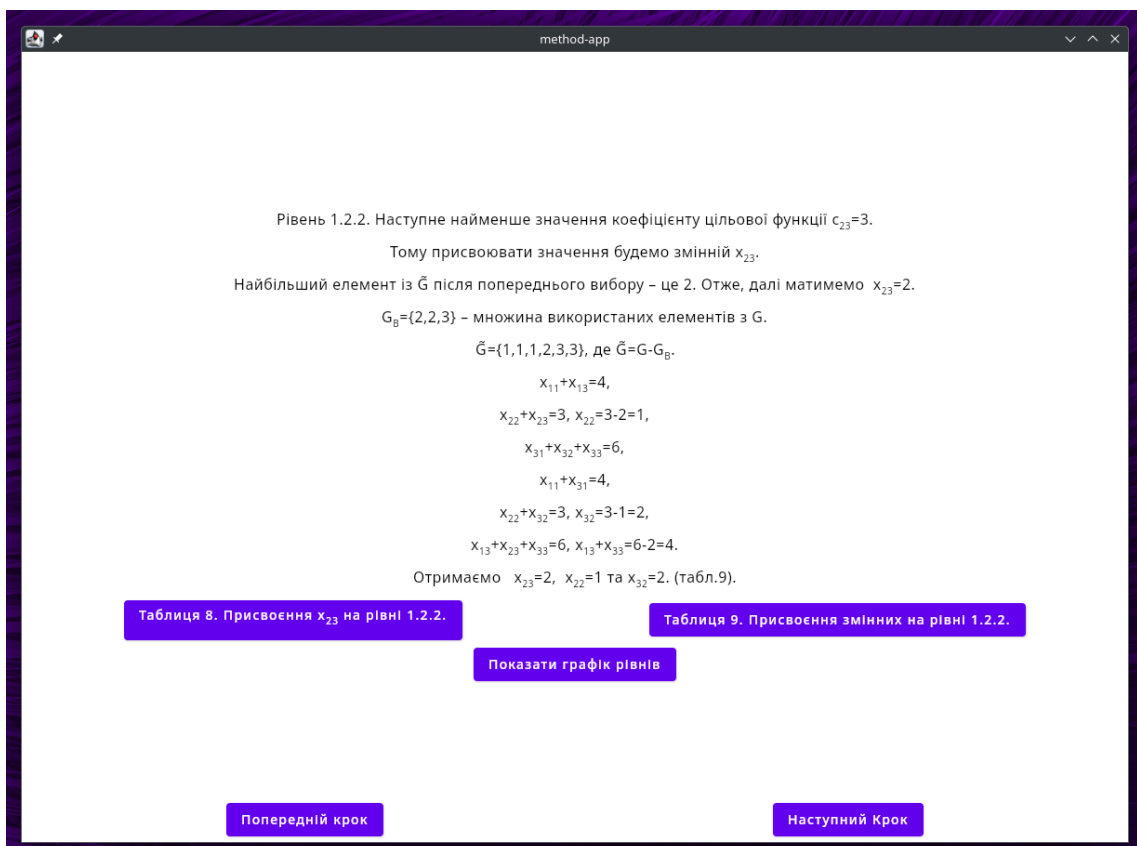
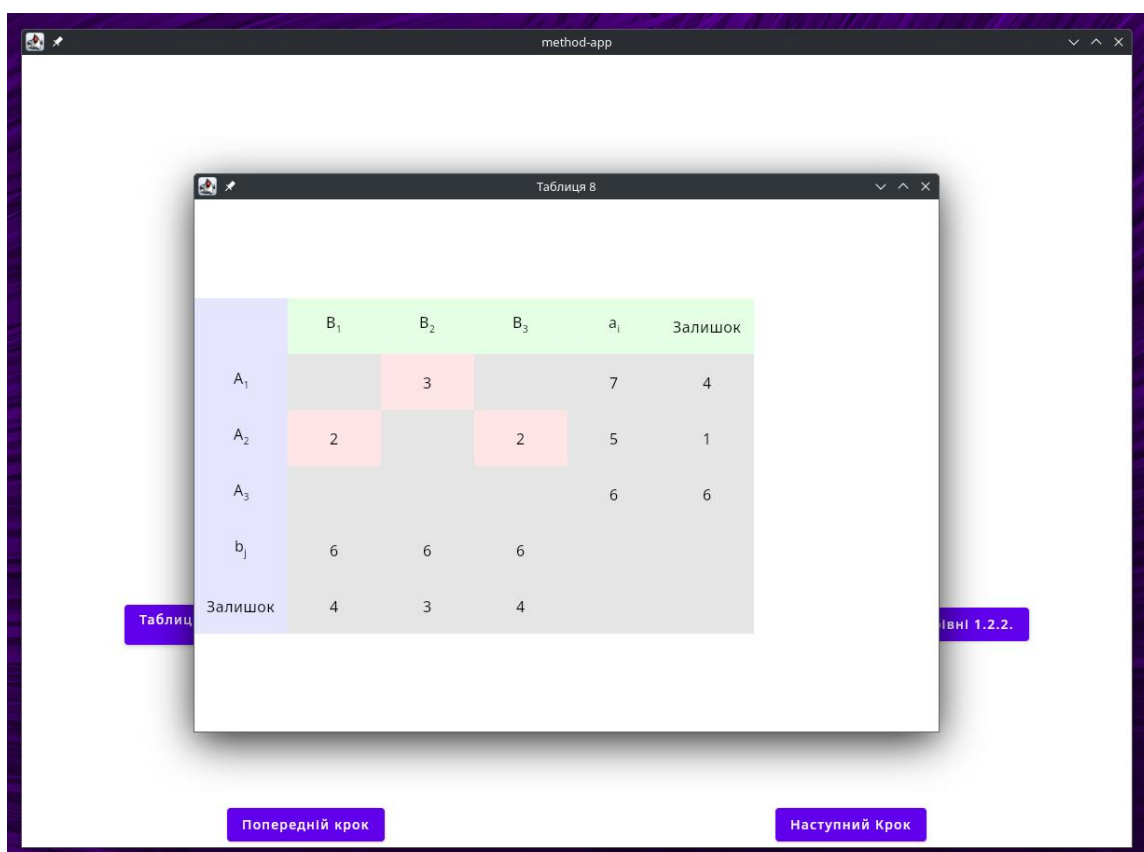


Рисунок 4.13 – Вікно програми з посиланнями на додаткові вікна

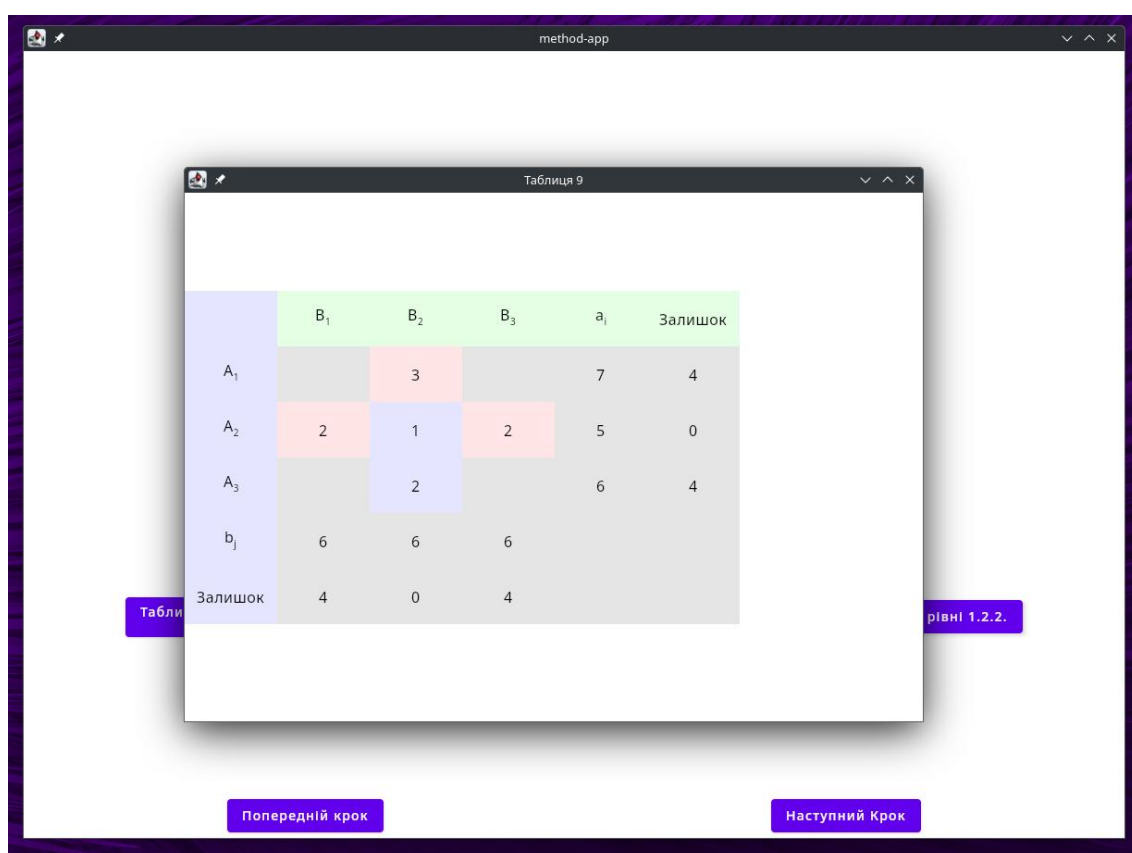


Таблиця 8

	B_1	B_2	B_3	a_i	Залишок
A_1		3		7	4
A_2	2		2	5	1
A_3				6	6
b_j	6	6	6		
Залишок	4	3	4		

Попередній крок Наступний крок

Рисунок 4.14 – Відображення таблиці для введення в окремому вікні



Таблиця 9

	B_1	B_2	B_3	a_i	Залишок
A_1		3		7	4
A_2	2	1	2	5	0
A_3		2		6	4
b_j	6	6	6		
Залишок	4	0	4		

Попередній крок Наступний крок

Рисунок 4.15 – Відображення таблиці з результатами в окремому вікні

Також можна відкривати декілька вікон одразу (рис.4.16).

Figure 4.16 shows two overlapping windows, 'Таблиця 8' and 'Таблиця 9', displaying linear programming tables. Both tables have columns for constraints (B_1, B_2, B_3), right-hand side values (a_i), and slack variables ('Залишок').

Таблиця 8:

	B_1	B_2	B_3	a_i	Залишок
A_1		3		7	4
A_2	2		2	5	1
A_3				6	6
b_j	6	6	6		
Залишок	4	3	4		

Таблиця 9:

	B_1	B_2	B_3	a_i	Залишок
A_1		3		7	4
A_2	2	1	2	5	0
A_3		2		6	4
b_j	6	6	6		
Залишок	4	0	4		

Рисунок 4.16 – Блок-схема алгоритму перевірки відповіді

Під час введення значень самостійного розв'язку, програма перевіряє введені значення на правильність, якщо значення введено неправильно, програма запропонує переглянути розв'язок (рис.4.17). Інакше запропонує перейти на наступний крок (рис.4.18).

Figure 4.17 shows the 'method-app' window with a linear programming table for independent data entry. The table has columns for constraints (B_1, B_2, B_3), right-hand side values (a_i), and slack variables ('Залишок').

Таблиця:

	B_1	B_2	B_3	a_i	Залишок
A_1		1		7	5
A_2				5	5
A_3				6	6
b_j	6	6	6		
Залишок	6	4	6		

Below the table, there are input fields for v and ξ , both containing the value 1.

At the bottom, there is a button labeled 'Показати рішення'.

Рисунок 4.17 – Таблиця для самостійного введення даних

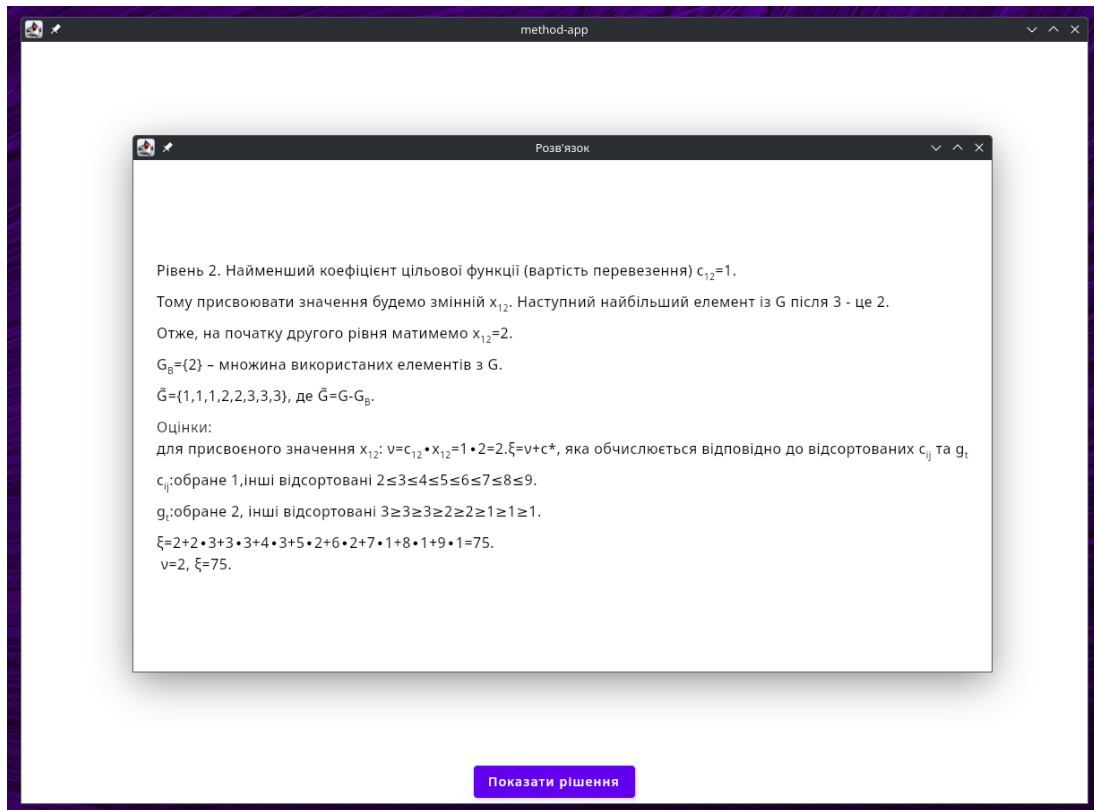


Рисунок 4.18 – Повідомлення в разі помилкового заповнення таблиці

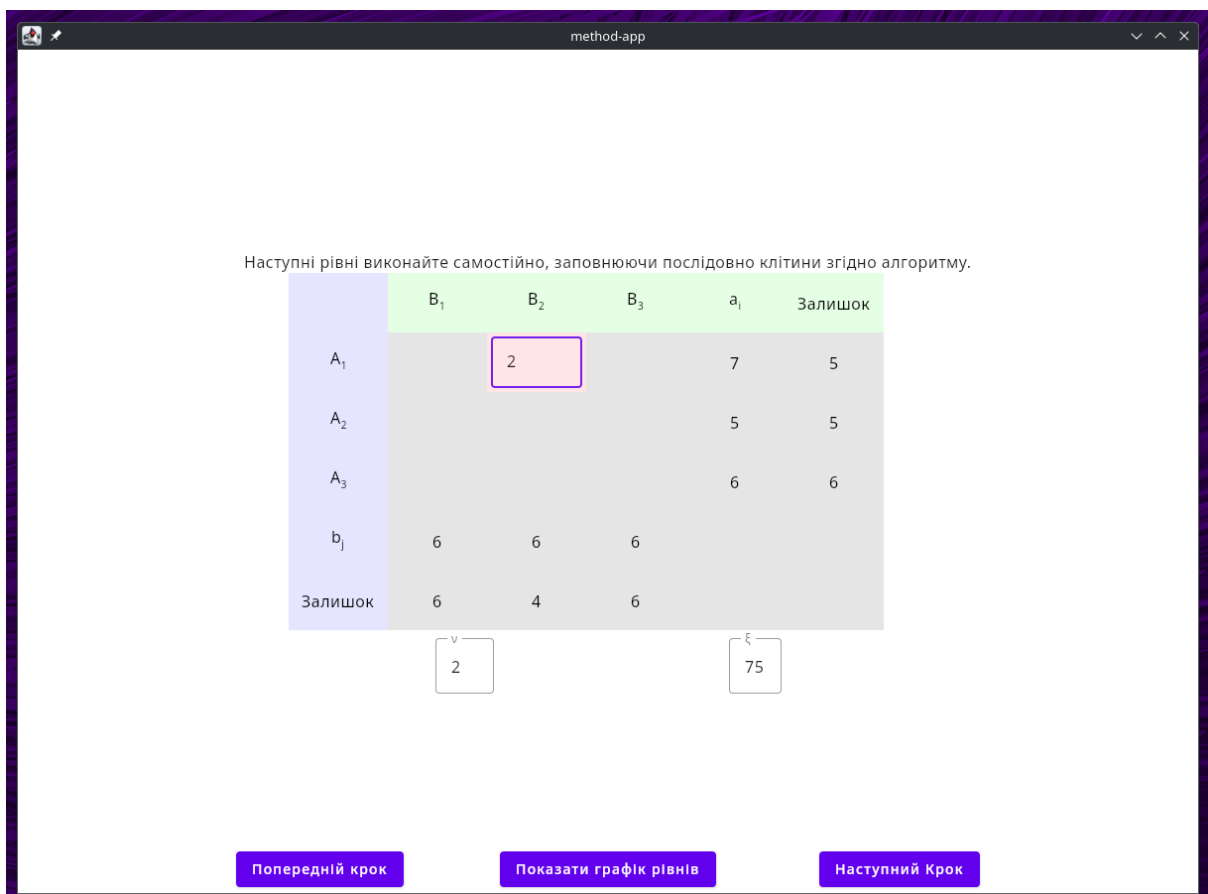


Рисунок 4.19 – Таблиця після коригування після неправильного введення

4.3. Інструкція користувачу

Після запуску програми з'являється стартове вікно (рис.4.5), на якому відображено назва задачі та методу її розв'язування та кнопка для переходу до його реалізації.

Програма має навчальну мету. Тому на початку користувачу надається можливість ознайомитись із описом реалізації на початковому етапі, зокрема, сортування елементів множини тарифів та місткостей. Крім того, описано хід реалізації першого рівня.

В подальшому програмою пропонується користувачу самостійно реалізувати другий та третій рівні, ввівши деякі значення. Для цього він має окремо обчислити всі необхідні оцінки.

4.4. Актуальність вибору мови програмування

Kotlin, розроблений компанією JetBrains, швидко став популярним вибором серед розробників, особливо в контексті Android-розробки. Офіційно визнаний Google як основна мова для Android у 2017 році, Kotlin продовжує зростати в популярності завдяки своїм численним перевагам. Відмітимо деякі з них.

- Сумісність з Java. Kotlin повністю сумісний з Java, що дозволяє розробникам використовувати існуючі Java-бібліотеки та поступово інтегрувати новий код у вже наявні проекти. Це значно спрощує міграцію з Java на Kotlin

- Скорочення обсягу коду. Синтаксис Kotlin є компактнішим і виразнішим, що дозволяє писати менше рядків коду для реалізації тих же функцій. Наприклад, один рядок коду на Kotlin може замінити до 14 рядків на Java.

- Безпека типів. Kotlin має систему типізації, яка допомагає уникнути багатьох поширених помилок, таких як NullPointerException. Це підвищує надійність програмного забезпечення і зменшує витрати на тестування

- Функціональні можливості. Kotlin підтримує функціональне програмування, що робить його ідеальним для створення сучасних додатків. Це також сприяє підвищенню продуктивності при розробці.

Kotlin активно використовується не лише для Android-розробки, але й у веб-розробці, бекенд-розробці та навіть у розробці ігор. Його багатоплатформенність дозволяє створювати програми для різних платформ з однієї кодової бази, що значно скорочує час і ресурси на розробку.

З розвитком технологій штучного інтелекту та мобільних платформ, попит на Kotlin продовжує зростати. Багато компаній переходять на цю мову або використовують її для нових проєктів, що свідчить про її актуальність і перспективність.

Google активно підтримує Kotlin в Android Studio, що забезпечує його подальше домінування у сфері мобільної розробки.

Таким чином, Kotlin є потужним інструментом для сучасних розробників завдяки своїй простоті використання, безпеці та широким можливостям. Його популярність зростає, особливо серед тих, хто працює в екосистемі Android. Вибір Kotlin як основної мови програмування може бути стратегічно вигідним рішенням для нових проєктів у різних сферах програмування.

ВИСНОВКИ

В результаті виконання кваліфікаційної роботи розроблено та програмно реалізовано алгоритм розв'язування комбінаторних транспортних задач на переставленнях на основі методу гілок та меж. Для цього було проведено огляд існуючих методів розв'язання подібних задач, ознайомлено та вивчено методичні рекомендації щодо написання кваліфікаційної роботи та теоретичний матеріал до теми, здійснено огляд можливостей застосування методу гілок та меж. Розроблено алгоритм роботи програми та обрано інструменти програмування для його реалізації.

Реалізовано розроблений алгоритм у вигляді програмного забезпечення з використанням мови програмування Kotlin, фреймворку Compose Desktop та бібліотеки Voyager. Програма забезпечує зручний інтерфейс для введення даних та візуалізації результатів.

Перспективами подальших досліджень може бути розширення функціоналу програмного забезпечення, наприклад, для генерування умов подібних задач різної вимірності.

В цілому, результати кваліфікаційної магістерської роботи відповідають поставленим завданням. Розроблене програмне забезпечення може бути використано в навчальному процесі під час вивчення курсу «Елементи комбінаторної оптимізації».

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Гаркуша С. В. Методичні рекомендації щодо виконання кваліфікаційної роботи студентів спеціальності 122 Комп'ютерні науки освітня програма «Комп'ютерні науки» ступеня магістра / С. В. Гаркуша, О. В. Ольховська, О. О. Черненко. – Полтава: ПУЕТ, 2023. – 68 с.
2. Ємець О.О. Методи оптимізації та дослідження операцій: навчальний посібник / О.О. Ємець. – Полтава: ПУЕТ, 2019. – Ч.2. – 139 с.
3. Колєчкіна Л.М. Властивості задач багатокритеріальної оптимізації на комбінаторних множинах та методи їх розв'язання: Монографія. – Полтава: РВВ ПУСКУ, 2008. – 162 с.
4. Донець Г. П. Екстремальні задачі на комбінаторних конфігураціях : монографія / Г. П. Донець, Л. М. Колєчкіна. – Полтава: РВВ ПУЕТ, 2011. – 309 с.
5. Парфьонова Т.О. Евклідова комбінаторна транспортна задача на переставленнях, її розв'язування методом гілок та меж / Т.О. Парфьонова // Матеріали Другої ювілейної науково-технічної конференції „Комп'ютерна математика в інженерії, науці та освіті”. (CMSEE), Полтава, 29-31 жовтня 2008 р. – Київ: Вид-во НАН України, 2008. – С. 24.
6. Парфьонова Т.О. Застосування методу гілок та меж до комбінаторних транспортних задач / Т.О. Парфьонова // Тези допов. IV міжнародної науково-практичної конференції „Методологія та практика менеджменту на порозі XXI століття: загальнодержавні, галузеві та регіональні аспекти” 15-16 травня 2008 р. – Полтава: РВЦ ПУСКУ, 2008. – С.352-353.
7. Парфьонова Т.О. Транспортні задачі комбінаторного типу, їх властивості та розв'язування / Т.О. Парфьонова // Інформатика та системні науки (ISN2010): Матеріали Всеукраїн. наук.-прак. конф. (18-20 березня 2010, Полтава). – Полтава: РВВ ПУСКУ. – С.153-154

8. Ємець Ол-ра.О. Розв'язування задач комбінаторної оптимізації на нечітких множинах / Ємець Ол-ра.О.: Автореф. дис. ... к.ф.-м.н: 01.05.01. – К: ІК НАНУ, 2009. – 19 с.
9. Крижанівський Б. Використання методу гілок і меж при розв'язанні задач лінійного цілочислового програмування: матеріали науково-технічної конференції «Наука. Освіта. Молодь». УДПУ, 2016. URL: https://library.udpu.edu.ua/library_files/stud_konferenzia/2016_1/85.pdf
10. Джеймс Гартман. Підручник Kotlin для початківців (приклад) URL: <https://www.guru99.com/uk/kotlin-tutorial.html>
11. John Hunt. Beginner's Guide to Kotlin Programming. Springer, 2021.
12. Kotlin Підручник. W3schoolsUA. українською. URL: <https://w3schoolsua.github.io/kotlin/index.html#gsc.tab=0>

ДОДАТОК А. КОД ПРОГРАМИ