

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ  
ФАКУЛЬТЕТ ЕЛЕКТРОНІКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ  
КАФЕДРА КОМП'ЮТЕРНИХ НАУК  
СЕКЦІЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ПРОЕКТУВАННЯ

## КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

на тему: «Моделі оцінювання організаційно-технічних заходів  
забезпечення надійності інформаційної системи в умовах  
небезпечних впливів»

за спеціальністю 122 «Комп'ютерні науки та інформаційні технології»,  
освітньо-професійна програма «Інформаційні технології проектування»

**Виконавець роботи:** студент групи ІТ.м-91 Курочкін Андрій Олегович

**Кваліфікаційну роботу  
захищено на засіданні ЕК  
з оцінкою**

\_\_\_\_\_ «\_\_» \_\_\_\_\_ 2020 р.

Науковий керівник

\_\_\_\_\_

(підпис)

д.т.н., проф., Лавров Є.А.

Голова комісії

\_\_\_\_\_

(підпис)

Шифрін Д. М.

Засвідчую, що у цій дипломній роботі немає  
запозичень з праць інших авторів  
без відповідних посилань.

Студент \_\_\_\_\_  
(підпис)

Суми-2020

Сумський державний університет  
Факультет електроніки та інформаційних технологій  
Кафедра комп'ютерних наук  
Секція інформаційних технологій проектування  
Спеціальність 122 «Комп'ютерні науки та інформаційні технології»  
Освітньо-професійна програма «Інформаційні технології проектування»

**ЗАТВЕРДЖУЮ**

Зав. секцією ІТП

\_\_\_\_\_ В. В. Шендрик  
«\_\_\_\_\_» \_\_\_\_ 2020 р.

## **ЗАВДАННЯ**

**на кваліфікаційну роботу магістра студентів**

*Курочкін Андрій Олегович*  
(прізвище, ім'я, по батькові)

**1 Тема проекту** Моделі оцінювання організаційно-технічних заходів забезпечення надійності інформаційної системи в умовах небезпечних впливів

затверджена наказом по університету від «\_\_» \_\_\_\_\_ 2020 р. № \_\_\_\_\_

**2 Термін здачі студентом закінченого проекту** «16» \_\_\_\_\_ грудня \_\_\_\_\_ 2020 р.

**3 Вхідні дані до роботи** Організаційно-технічні заходи, модель оптимізації вибору заходів, вразливості інформаційної системи, система взаємопов'язаних баз даних, граф подій усунення негативного впливу на базу даних

**4 Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)** вступ, аналіз предметної області, моделювання процесів розробки методу вибору технологій забезпечення надійності баз даних інформаційної системи в умовах конфліктної взаємодії, розробка моделі вибору технологій забезпечення надійності баз даних інформаційної системи в умовах конфліктної взаємодії, висновки, список використаної літератури

**5 Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)** Спроектвана модель ІС залізничного транспорту, дерево декомпозиції на класи еквівалентності, прототип інформаційної технології, результат роботи програми, граф подій життєвого циклу негативного впливу на базу даних, ТФС «Робоча – контроль функціонування», Результати обчислень

## 6. Консультанти розділів роботи:

| Розділ | Консультант | Підпис, дата   |                  |
|--------|-------------|----------------|------------------|
|        |             | Завдання видав | Завдання прийняв |
|        |             |                |                  |

Дата видачі завдання \_\_\_\_\_.

Керівник \_\_\_\_\_  
(підпис)

Завдання прийняв до виконання \_\_\_\_\_  
(підпис)

## КАЛЕНДАРНИЙ ПЛАН

| № п/п | Назва етапів кваліфікаційної роботи   | Строк виконання етапів роботи | Примітка |
|-------|---|-------------------------------|----------|
| 1.    | Ознайомлення з методами оптимізації та ідентифікація ідеї проекту   | 27.04.20-05.05.20             |          |
| 2.    | Детальний аналіз методів оптимізації процесів функціонування людино-машинної системи, визначення ресурсів, ризиків та вимог | 06.05.20-19.05.20             |          |
| 3.    | Розробка задачі оптимізації вибору забезпечення надійності інформаційної системи в умовах небезпечних впливів               | 20.05.20-14.07.20             |          |
| 4.    | Розробка методу ранжування БД за ступенем впливу на надійність системи управління   | 15.07.20-08.09.20             |          |
| 5.    | Розробка комплексного методу для вибору стратегій управління надійністю системи БД  | 09.09.20-01.12.20             |          |
| 6.    | Створення документації  | 02.12.20-03.12.20             |          |
| 7.    | Презентація проекту   | 21.12.20-22.12.20             |          |
| 8.    | Архівація   | 23.12.20-23.12.20             |          |

Магістрант \_\_\_\_\_

Курочкін А.О.

Керівник роботи \_\_\_\_\_

д.т.н., проф., Лавров Є.А.

## РЕФЕРАТ

Тема магістерської роботи: «Моделі оцінювання організаційно-технічних заходів забезпечення надійності інформаційної системи в умовах небезпечних впливів».

Пояснювальна записка складається зі вступу, 4 розділів, висновків, списку використаних джерел із 31 найменування, додатків. Загальний обсяг роботи – 84 сторінки, у тому числі 55 сторінок основного тексту, 4 сторінки списку використаних джерел, 25 сторінок додатків.

Кваліфікаційну роботу магістра присвячено розробці комплексної моделі оцінювання організаційно-технічних заходів, яка складається із моделі ранжування коефіцієнтів впливу баз даних, графової моделі життєвого циклу небезпечних впливів у базі даних, постановки та розв'язання задачі оптимізації.

В роботі проведено аналіз взаємодії небезпечного впливу та інформаційних систем.

У роботі виконано моделювання процесів розробки методу надання надійності інформаційної системи.

Результатом проведеної роботи є реалізовані моделі оцінювання організаційно-технічних заходів забезпечення надійності інформаційної системи в умовах небезпечних впливів.

Практичне значення роботи полягає у використанні моделі як основи системи забезпечення надійності інформаційних систем.

Ключові слова: задача оптимізації, небезпечні впливи, джерело негативного впливу, ранжування елементів системи, надійність інформаційної системи.

## ЗМІСТ

|   |    |
|---|----|
| ВСТУП .....   | 7  |
| 1. АНАЛІЗ КОНФЛІКТІВ НАДАННЯ НАДІЙНОСТІ<br>ІНФОРМАЦІЙНИМ СИСТЕМАМ В УМОВАХ НЕБЕЗПЕЧНИХ ВПЛИВІВ 8  |    |
| 1.1 Характеристика забезпечення надійності інформаційних систем в умовах<br>небезпечних впливів .....   | 8  |
| 1.2. Оцінювання надійності баз даних інформаційної системи в умовах<br>небезпечних впливів .....  | 12 |
| 2. ПОСТАНОВКА ЗАДАЧІ ТА МЕТОДИ ДОСЛІДЖЕННЯ.....   | 15 |
| 2.1. Мета та задачі дослідження .....   | 15 |
| 2.2. Методи дослідження.....  | 15 |
| 3. ПЛАНУВАННЯ ПРОЦЕСУ РОЗРОБКИ МОДЕЛЕЙ ОЦІНЮВАННЯ<br>ОРГАНІЗАЦІЙНО-ТЕХНІЧНИХ ЗАХОДІВ ЗАБЕЗПЕЧЕННЯ НАДІЙНОСТІ<br>ІНФОРМАЦІЙНОЇ СИСТЕМИ В УМОВАХ НЕБЕЗПЕЧНИХ ВПЛИВІВ....                                | 17 |
| 3.1. Структурно функціональне моделювання процесів розробки моделей<br>оцінювання організаційно-технічних заходів забезпечення надійності<br>інформаційної системи в умовах небезпечних впливів ..... | 17 |
| 3.2. Моделювання варіантів використання .....   | 23 |
| 4. РОЗРОБКА МОДЕЛЕЙ ОЦІНЮВАННЯ ОРГАНІЗАЦІЙНО-ТЕХНІЧНИХ<br>ЗАХОДІВ ЗАБЕЗПЕЧЕННЯ НАДІЙНОСТІ ІНФОРМАЦІЙНОЇ СИСТЕМИ В<br>УМОВАХ НЕБЕЗПЕЧНИХ ВПЛИВІВ .....   | 25 |
| 4.1. Загальні відомості про напівмарківські процеси.....  | 25 |
| 4.2. Модель ранжування коефіцієнтів впливу елементів системи .....  | 27 |
| 4.3. Ранжування елементів інформаційної системи баз даних для<br>забезпечення надійності на прикладі створеної системи .....  | 31 |
| 4.4. Інформаційна технологія ранжування елементів інформаційної системи<br>на основі транзитивного замикання .....  | 38 |
| 4.5. Графова модель життєвого циклу небезпечного впливу бази даних .....  | 47 |

|   |    |
|---|----|
| 4.6. Розробка математичної моделі вибору рішення виявлення та усунення небезпечного впливу..... | 49 |
| 4.7. Приклад розв'язання задачі оптимізації забезпечення надійності бази даних.....             | 53 |
| ВИСНОВКИ .....  | 55 |
| СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ.....   | 56 |
| ДОДАТОК А. ПЛАНУВАННЯ РОБІТ.....  | 60 |
| ДОДАТОК Б. ЛІСТИНГ ПРОГРАМИ.....  | 68 |
| ДОДАТОК В. КОМП'ЮТЕРНІ ЕКСПЕРИМЕНТИ .....   | 76 |
| ДОДАТОК Г. ДИПЛОМИ ТА ГРАМОТИ.....  | 82 |

## ВСТУП

**Актуальність.** З плином часу, кількість кібератак, вірусів та негативного зовнішнього впливу на інформаційні системи збільшується. Незалежно від кількості проведених досліджень та розроблених систем захисту інформації, задача оптимізації забезпечення надійності інформаційних систем залишається невирішеною.

**Предмет дослідження.** Задача оптимізації вибору організаційно-технічних заходів в умовах небезпечних впливів.

**Об'єкт дослідження.** Надійність бази даних інформаційної системи.

**Мета дослідження.** Розробити моделі оцінювання заходів по забезпеченню надійності баз даних в умовах небезпечних впливів

**Наукова новизна.** Розроблені моделі дозволяються аналізувати процес життєвого циклу небезпечного впливу, а також обирати оптимальні рішення, які допоможуть підтримувати працездатність інформаційної системи використовуючи мінімальні витрати та час.

**Апробація.** Результати дослідження доповідалися на конференціях:

- всеукраїнський конкурс студентських наукових робіт з напрямку «Кібербезпека» (Київ, 5 квітня 2019 року).
- всеукраїнський конкурс студентських наукових робіт з напрямку «Комп'ютерні науки» (Харків, 25 квітня 2019 року);

**Публікації.** Наукова робота була опублікована на сайті студентських конкурсів наукових робіт [19].

**Участь в конкурсі студентських наукових робіт.** Робота була переможцем:

- II туру всеукраїнського конкурсу студентських наукових робіт з напрямку «Кібербезпека» (Київ, 5 квітня 2019 року);
- II туру всеукраїнського конкурсу студентських наукових робіт з напрямку «Комп'ютерні науки» (Харків, 25 квітня 2019 року). Копії дипломів додаються.

# 1. АНАЛІЗ КОНФЛІКТІВ НАДАННЯ НАДІЙНОСТІ ІНФОРМАЦІЙНИМ СИСТЕМАМ В УМОВАХ НЕБЕЗПЕЧНИХ ВПЛИВІВ

## 1.1 Характеристика забезпечення надійності інформаційних систем в умовах небезпечних впливів

Внаслідок широкомасштабного розширення мережевих підключень спостерігається швидке зростання кількості кібератак на корпорації та урядові установи, що призвели до зривів у ділових операціях, завдаючи шкоди репутації, а також фінансовій стабільності цих корпорацій. Нещодавній випадок кібератаки на Target Corp ілюструє, як ці порушення безпеки можуть бути серйозно впливають на прибуток та вартість акціонерів. Згідно з повідомленням Secunia [30], кількість повідомлень про вразливі місця безпеки в 2013 році зросли на 32% відносно 2012 р. Однак, незважаючи на ці зростаючі темпи нападів на корпоративні та державні системи, корпорації впали відставання від нарощування оборони через обмеженість бюджетів, а також слабку практику безпеки.

Однією з головних перешкод, що наразі стикаються з вимірюванням безпеки, є розробка механізму вибору організаційно-технічних заходів забезпечення надійності всіх систем в мережі для оцінки загальної безпека мережі.

Надійність - властивість об'єкта виконувати в часі заданий функції.

Ознаки класифікації, що відрізняють одну відновлювальну роботу від іншої (які впливають на тривалість):

- стан системи на початку роботи;
- стан системи в кінці роботи;
- момент початку роботи.

На початку роботи система може бути:



- непрацездатною (робота зупинена через аварію), в цьому випадку маємо справу з аварійною роботою;
- працездатною (робота зупинена для профілактики), в цьому випадку в системі проводиться попереджувальна робота.

Якщо в кінці роботи по відновленню системи повністю оновлена, то ця робота являється роботою повного оновлення системи.

У випадку відомого моменту початку відновлювальної роботи мається на увазі планова робота, якщо відомих немає - відновна.

Назви відновлювальних робіт:

- аварійна і попереджувальна;
- повна і неповна;
- планова і позапланова.

У завданнях надійності також можливе застосування теорії керованих напівмарковських процесів.

Наприклад, INFOSEC [31] визначив показники безпеки одним із таких 8 основних пріоритетів досліджень безпеки. Подібним чином і Консультативний комітет з питань ІТ з питань кібербезпеки визначив цю сферу серед 10 найкращих пріоритетів досліджень безпеки.

Підходи, що використовуються в державному регулюванні питань надійності використання інформаційних технологій у рамках діючих систем, не враховують динаміку небезпечного впливу у базах даних інформаційних, а також не враховують небезпечний вплив від самого джерела небезпечного впливу. Підходи (не закріплені в державних нормативних документах), які навчають дані динамічних процесів, проводять симуляцію процесів без потрібного навчання на основі важливих показників. Тому, з однієї сторони, залишається відкритим питання про оцінку параметрів, характерних цих процесів, що є не ясным, яким чином за допомогою цих

підходів численно оцінити надійність конкретних інформаційних систем під час реальних ситуацій.

У якості універсальної синтетичної методології досліджень в сферах забезпечення надійності інформаційної системи (у той час баз даних) або сукупності систем доцільно розглядати методи математичного та комп'ютерного моделювання динаміки змін стану інформаційних процесів і систем, що описуються на концептуальних моделях конфліктних взаємодій і дозволяючи вчити всі найважливіші фактори таких взаємодій. Таким чином, тема дипломної роботи присвячена розробці моделей виявлення та усуненню небезпечного впливу для забезпечення надійності в рамках інформаційних систем.

Для створення дієвої системи кіберзахисту (ДСК) критично важливих інформаційних систем (КВІС), вибору і впровадження адекватних технічних компонентів повинен передувати етап опису, аналізу та моделювання кіберзагроз та вразливостей КВІС. Таким чином, цілком очевидно, що спочатку кіберзагрози повинні бути розпізнані, ідентифіковані та класифіковані.

Неповнота первинних даних про кіберзагрози для КВІС має двояке якість. Це, в першу чергу, відсутність передісторії (іноді, часткове), в тому числі, на рівні даних про структуру всього об'єкта кібератаки [10, 11, 15, 16], до початку дій атакуючої сторони. А, по-друге, обмежені можливості спостереження конкретної мети кібератаки і ідентифікації загроз, що належать певному класу. У граничному випадку заздалегідь відомо лише загальне безліч загроз для інформаційної безпеки (ІБ) КВІС і способів їх реалізації. При це неповнота спостережень і оцінки за інцидентами в області ІБ в несприятливих випадках означає, що суб'єкт може лише оцінювати відгуки об'єкта з точки зору своєї переваги. Однак при виникненні нових кіберзагроз та вразливостей для КВІС подібний підхід не завжди може сприяти ефективному захисту від атак. Нехай існує набір кіберзагроз для КВІС, загальна класифікація

загроз приведена в [2, 4, 12, 16]. Показник небезпеки кожної кіберзагрози залежить від значень набору факторів, що підвищують або знижують захищеність КВІС від даної загрози. Показники, що знижують захищеність КВІС прийнято вважати показниками ризику [16], а підвищують її - показниками захищеності [4, 6]. Для формалізації залежності ступеня захищеності КВІС від відповідних показників можна застосувати один з наступних підходів [13, 14, 26]:

1) кіберзагроз в рамках класу залежить від одного показника, т. е. між ступенем загрози і значеннями показника (фактор) існують відносини взаємо-однозначної відповідності;

2) кіберзагрози залежать від значень багатьох показників;

3) одні й ті ж показники впливають на ступінь захищеності КВІС не від одного, а від багатьох видів кіберзагроз.

Для забезпечення однозначності, повноти і надійності класифікації, введемо наступні вимоги до класифікації кіберзагроз:

- непересічні класи загроз (визначає однозначність вибору класу на основі зовнішнього правила, що дозволяє прийняти рішення);
- застосовність (додавання класу не повинно викликати дроблення більш одного класу на дві частини);
- об'єктивність (наявність або відсутність класу має підтверджуватися відомими класифікаціями);
- розширюваність (додавання класу можливо шляхом дроблення існуючих класів);
- кількість класів кіберзагроз.

У загальному випадку, проблематика розпізнавання аномалій, кібератак або погроз для КВІС полягає в наступному [1, 3, 7, 11, 16]. Досліджується деякий безліч об'єктів, в нашому випадку це  $P_A$  – число можливих цілей атакуючої КВІС боку.

Множин даних об'єктів має вигляд ознак  $\{S_{ax1} \dots S_{ax1}\}$ , представлених, наприклад, в двійковій формі. Відомо, що безліч  $P_A$  представлена як унітарні підмножини класів кіберзагроз для КВІС. Відмінною рисою розглянутих в роботі логічних процедур, полягає в можливості отримати достовірний результат, коли відсутня апріорна інформація про функції розподілу наявних значень ознак загрози, кібератаки або аномалії. Такі процедури будемо надалі називати логічними процедурами. При цьому немає необхідності задавати опис об'єктів у просторі, що характеризують кожен клас. Таким чином, для кожної ознаки кібератаки (аномалії, загрози, вразливості) існує деяка функція бінарної наближення між його обчисленнями, що має можливість виділяти об'єкти і їх уявлення.

В якості інформативних в АСРК доцільно використовувати тільки ті фрагменти, які відображають характерні закономірності в описах об'єктів, що використовуються для навчання. Отже, присутність (відсутність) таких фрагментів в класифікуються об'єктів дозволяє судити про його приналежності до класу. При використанні логічних процедур розпізнавання кібератак (ЛПК) також інформативними визнаємо фрагменти, що з'являються у описі об'єкту одного класу кібератак, але відсутні у об'єктних описах інших класів. Використовувані фрагменти також включають змістовне опис закономірностей в термінах проектування АСРК.

## **1.2. Оцінювання надійності баз даних інформаційної системи в умовах небезпечних впливів**

На сьогодні у вільному доступі є величезна чисельність методів аналізування надійності в умовах небезпечних впливів як для будь-якого роду кіберзагроз, так і для окремих інформаційних систем. Навпроти від широкої кількості робіт, які були

присвячені оцінюванню виявленню та усуненню проблем програмного забезпечення, у даному проекті основна увага сфокусована на аналізі методів, які таким чи іншим чином приділяють увагу питанню спроможності усунення вразливостей програмного забезпечення для як зовнішніх, так і внутрішніх небезпечних впливів, що порушують працеспроможність інформаційних систем. Дані методи розділяють на 3 види:

1. методи, які мають закріплений міжнародний статус та які прив'язані до офіційних нормативних документів.

2. методи, які використовують у вільному доступі для забезпечення комп'ютерної безпеки.

3. методи, що лише знаходяться у вигляді наукових робіт.

При виконання порівняння описаних підходів, потрібно мати на увазі, як доцільно вони рахуються з реальними умовами функцій інформаційних систем при присутності навмисних небезпечних впливів. Тобто яку кількість факторів вони аналізують, які саме це показники і як вони обчислюються. Підход до аналізування надійності при навмисних небезпечних впливах потрібно порівнювати за такими факторами:

1. врахування станів працездатності інформаційної системи.

2. врахування процесів інформаційної системи.

3. врахування видів процесів: детермінований або недетермінований процеси.

4. врахування залежних параметрів процесу.

5. як оцінюються та враховуються параметри (оцінювання або на основі наявної статистики або на основі прогнозу).

Виконуючи порівняння методів аналізу надійності при навмисних небезпечних впливах застосовуючи перший критерій, їх розділяють на статичні підходи і динамічні підходи. До статичних підходів належать такі методи, які мають можливість обчислювати лише певний стан інформаційної системи, здебільшого - це

поточний стан інформаційної системи. Таким чином проводиться аналіз певних умов функціонування інформаційних систем, і на основі даних аналізу проводиться оцінка надійності ІС. Але планується, що дані певні умови функціонування інформаційної системи будуть перебувати у проектуванні системи без змін, а якщо змін все рівно настали, то ці самі зміни будуть виконуватися експертною групою, яка відповідає за забезпечення надійності інформаційної системи, у результаті чого з'являється можливість оперативно зробити оцінку надійності інформаційної системи.

Головною проблемою даного підходу є те, що не передбачена зміна умов функціонування виконується експертною групою. Динамічні підходи аналізують фактори характерів процесів, які впливають на надійність інформаційної системи, а не фактори, що описують певний стан інформаційної системи.

## **2. ПОСТАНОВКА ЗАДАЧІ ТА МЕТОДИ ДОСЛІДЖЕННЯ**

### **2.1. Мета та задачі дослідження**

З плином часу, кількість кібератак, вірусів та негативного зовнішнього впливу на інформаційні системи збільшується. Незалежно від кількості проведених досліджень та розроблених систем захисту інформації, задача оптимізації забезпечення надійності інформаційних систем залишається невирішеною.

Аналізування предметної області дипломної роботи дає можливість виділити наступні задачі проекту:

- побудова графової моделі процесу введення БД в умовах негативної взаємодії;
- аналіз альтернативних варіантів виявлення і усунення порушень;
- ранжування баз даних за ступенем впливу на надійність системи управління баз даних;
- формування напівмарківського процесу прийняття рішень для вибору стратегії управління надійністю БД.

### **2.2. Методи дослідження**

Для розроблення моделі вибору організаційно-технічних засобів були застосовані Microsoft Excel 2020, та Microsoft Office Visio 2020 та мова програмування C#.

Microsoft Excel - це програмне забезпечення, яке використовується для роботи з електронними таблицями, що входить до набору програм Microsoft Office. Електронні таблиці надають значення, які можна математично змінити,

використовуючи як основні, так і складні арифметичні операції. На додаток до стандартних функцій електронної таблиці, Excel пропонує підтримку програмування за допомогою Microsoft Visual Basic for Applications (VBA), можливість доступу до даних із зовнішніх джерел за допомогою Microsoft Dynamic Data Exchange (DDE).

Навпроти текстовому редактору, такого як Microsoft Word, Excel упорядковує дані у стовпці та рядки. Рядки і стовпці перетинаються в просторі, який називається клітиною. Кожна комірка може містити окремі дані, такі як текст, числове значення або формула.

Excel - це інструмент для організації та проведення обчислень даних. Він може аналізувати дані, обчислювати статистику, генерувати зведені таблиці та представляти дані як діаграму чи графік. Наприклад, ви можете створити електронну таблицю Excel, яка обчислює щомісячний бюджет, відстежує пов'язані витрати та сортує дані за критеріями у вигляді інтерактивного режиму.

Visio - це програма, яка являється частиною Microsoft Office Suite продуктів. Вона використовується для багатьох речей, які використовують макети, схеми та діаграми. Графіка, що використовується у Visio, - це стандартні зображення, що використовуються блок-схемами, схемами прийняття рішень, ігровими книгами та навіть мережевими діаграмами. Microsoft Visio має достатньо діаграм, щоб зробити планування плану приміщення для конференції точним і простим.



### **3. ПЛАНУВАННЯ ПРОЦЕСУ РОЗРОБКИ МОДЕЛЕЙ ОЦІНЮВАННЯ ОРГАНІЗАЦІЙНО-ТЕХНІЧНИХ ЗАХОДІВ ЗАБЕЗПЕЧЕННЯ НАДІЙНОСТІ ІНФОРМАЦІЙНОЇ СИСТЕМИ В УМОВАХ НЕБЕЗПЕЧНИХ ВПЛИВІВ**

#### **3.1. Структурно функціональне моделювання процесів розробки моделей оцінювання організаційно-технічних заходів забезпечення надійності інформаційної системи в умовах небезпечних впливів**

Існує широкий спектр методів для діаграмних процесів, таких як блок-схеми та діаграми потоків даних, але небагато з них є суворо визначеними. SADT, або техніка структурованого аналізу та проектування, була передана у відкрите надбання як IDEF0 Американським Міністерством Оборони як стандарт для визначення процесу. З тих пір він був прийнятий багатьма організаціями, які шукають простий, але суворий спосіб опису своїх процесів.

IDEF0 - це багато в чому дуже простий метод. Кожне поле представляє окремий процес, як і в інших підходах, але IDEF0 відрізняється використанням та розміщенням стрілок. Окрім звичайних входів і виходів, існують ще два типи стрілок, які представляють "елементи керування" та "механізми".

Елементи управління використовуються для керування діяльністю в процесі. Іноді виникає певна міра невизначеності щодо того, чи є елемент предметом введення чи контролю. Простий спосіб розрізнити їх полягає в тому, що входи перетворюються або змінюються якимось чином, щоб створити виходи, тоді як елементи управління рідко змінюються. Стандарти, плани, шаблони та контрольні списки - це всі форми контролю.

Механізми у свою чергу являють собою інструменти, які потрібні для закінчення процесу моделі. У механізми можуть входити як працівники із потрібними вміннями, машинами та іншими інструментами.

Чотири типи стрілок, входи, елементи керування, виходи та механізми, спільно називаються ICOM, а IDEF - це аббревіатура визначення ICOM.. Різні стрілки ICOM ідентифікуються збоку вікна активності, до якого вони торкаються. Таким чином, входи знаходяться ліворуч, елементи управління - зверху, виходи - праворуч, а механізми - знизу. Це може ускладнити малювання діаграм, але спростити їх читання.

Для розуміння процесу, який забезпечують моделі оцінювання організаційно-технічних заходів забезпечення надійності, на рис. 3.1 наведена контекстна діаграма.

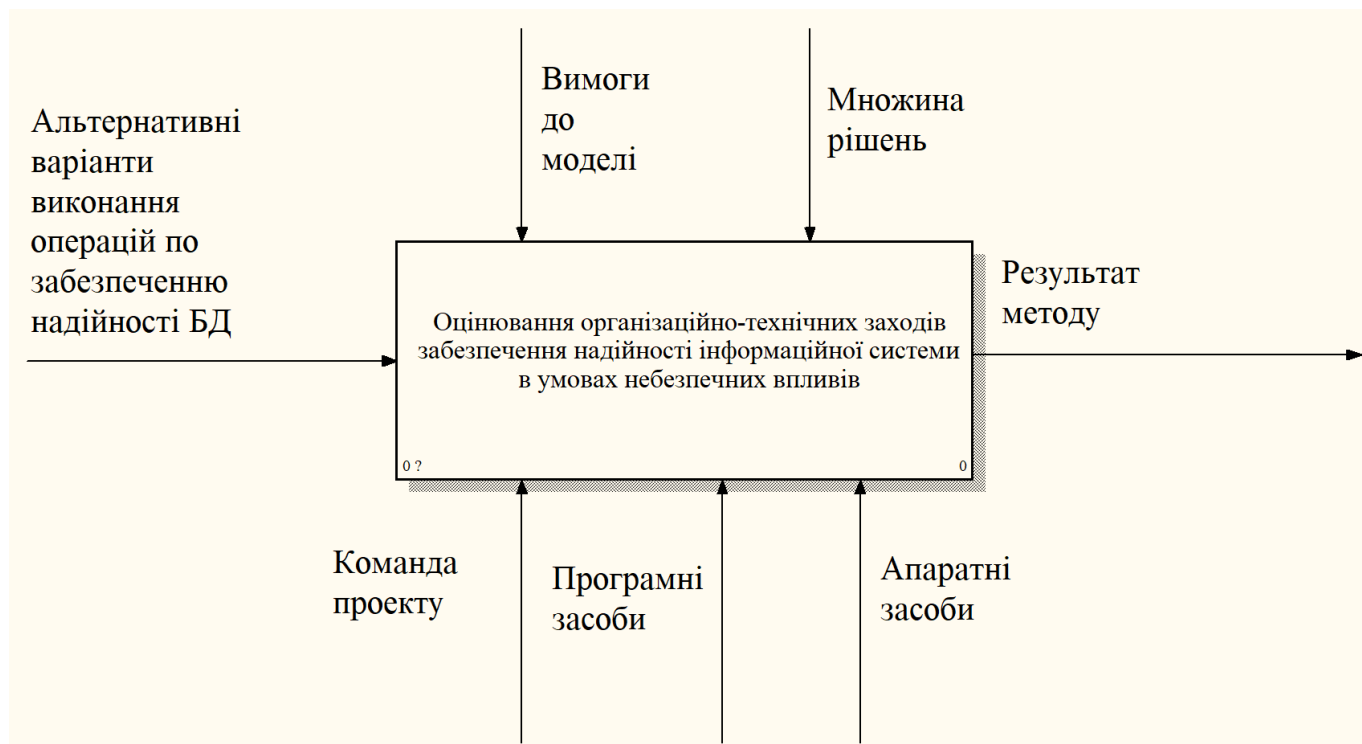


Рисунок 3.1 – Контекстна діаграма IDEF0

Контекстна діаграма розбита на 7 підфункцій, такі як: побудова графової моделі процесу ведення БД в умовах негативної взаємодії, змістовний аналіз альтернативних варіантів виявлення і усунення порушень, формування НМП прийняття рішень для вибору стратегії управління надійністю БД, формальна постановка задачі на НМП при виборі стратегії управління надійністю БД, аналіз поставленої задачі, розробка методу ранжування баз даних за ступенем впливу на надійність системи управління, розробка комплексного методу для вибору стратегії управління надійністю системи БД.

- перший етап: побудова графової моделі процесу ведення БД в умовах негативної взаємодії;
- другий етап: змістовний аналіз альтернативних варіантів виявлення і усунення порушень;
- третій етап: формування НМП прийняття рішень для вибору стратегії управління надійністю БД;
- четвертий етап: формальна постановка задачі на НМП при виборі стратегії управління надійністю БД;
- п'ятий етап: аналіз поставленої задачі;
- шостий етап: розробка методу ранжування баз даних за ступенем впливу на надійність системи управління
- сьомий етап: розробка комплексного методу для вибору стратегії управління надійністю системи БД

Опис першого етапу:

- вхідні дані: альтернативні варіанти;
- вихідні дані: граф моделі життєвого циклу БД побудований;
- управління: вимоги до моделі, планування розробки;
- механізми: команда проекту та програмні засоби.

Опис другого етапу:

- вхідні дані: граф моделі життєвого циклу БД побудований;
- вихідні дані: список порушень з'ясований;

- управління: вимоги до моделі, планування розробки;
- механізми: команда проекту.

Опис третього етапу:

- вхідні дані: список порушень з'ясований;
- вихідні дані: напівмарківський процес сформований;
- управління: вимоги до моделі, планування розробки;
- механізми: команда проекту.

Опис четвертого етапу:

- вхідні дані: напівмарківський процес сформований;
- вихідні дані: задача на напівмарківському процесу при виборі стратегії поставлена;
- управління: вимоги до моделі, планування розробки;
- механізми: команда проекту.

Опис п'ятого етапу:

- вхідні дані: задача на напівмарківському процесу при виборі стратегії поставлена;
- вихідні дані: задача вирішена;
- управління: вимоги до моделі, планування розробки;
- механізми: команда проекту.

Опис шостого етапу:

- вхідні дані: задача лінійного програмування вирішена;
- вихідні дані: метод ранжування БД за ступенем впливу розроблений;
- управління: вимоги до моделі, планування розробки;
- механізми: команда проекту, апаратні засоби, програмні засоби.

Опис сьомого етапу:

- вхідні дані: метод ранжування БД за ступенем впливу розроблений;
- вихідні дані: оцінювання організаційно-технічних заходів виконане;
- управління: вимоги до моделі, планування розробки;
- механізми: команда проекту, апаратні засоби, програмні засоби.

Діаграма декомпозиції контекстної діаграми зображена на рис.3.2.

Потім були описані п'ятий, шостий та сьомий етапи. Дані декомпозиції зображені на рис. 3.3-3.5 відповідно.

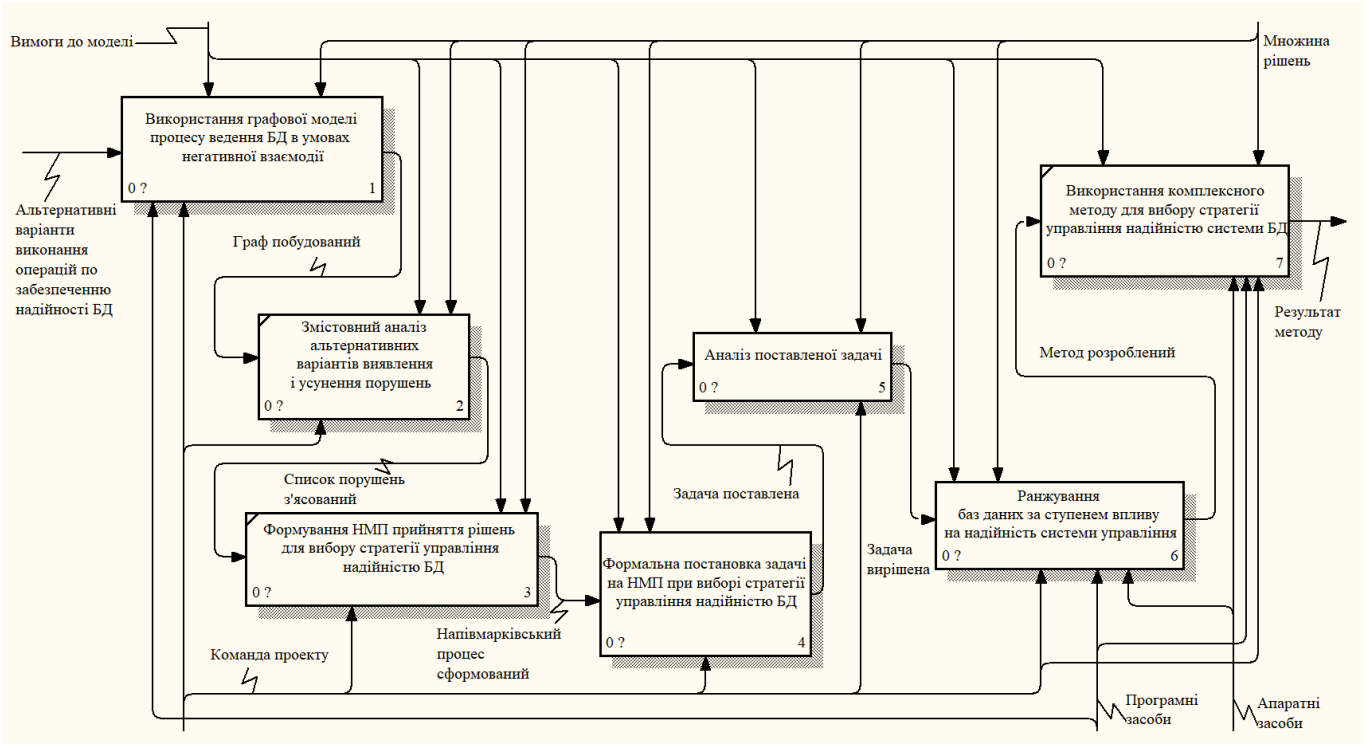


Рисунок 3.2 - Діаграма другого рівня

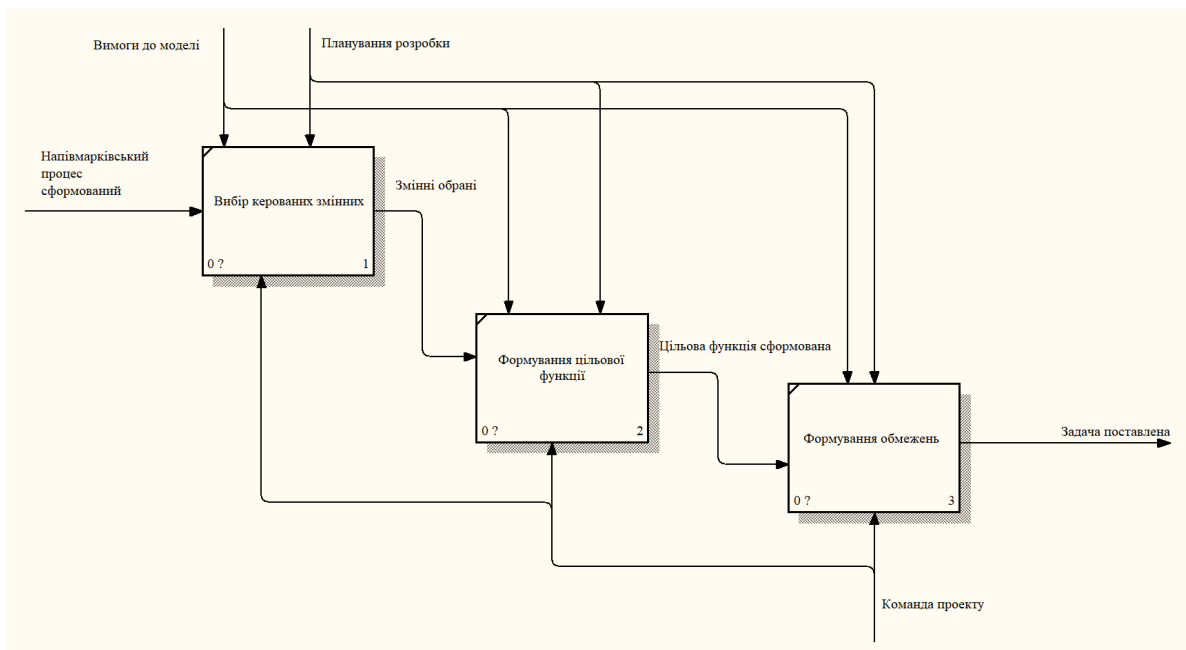


Рисунок 3.3 - Діаграма декомпозиції четвертого етапу

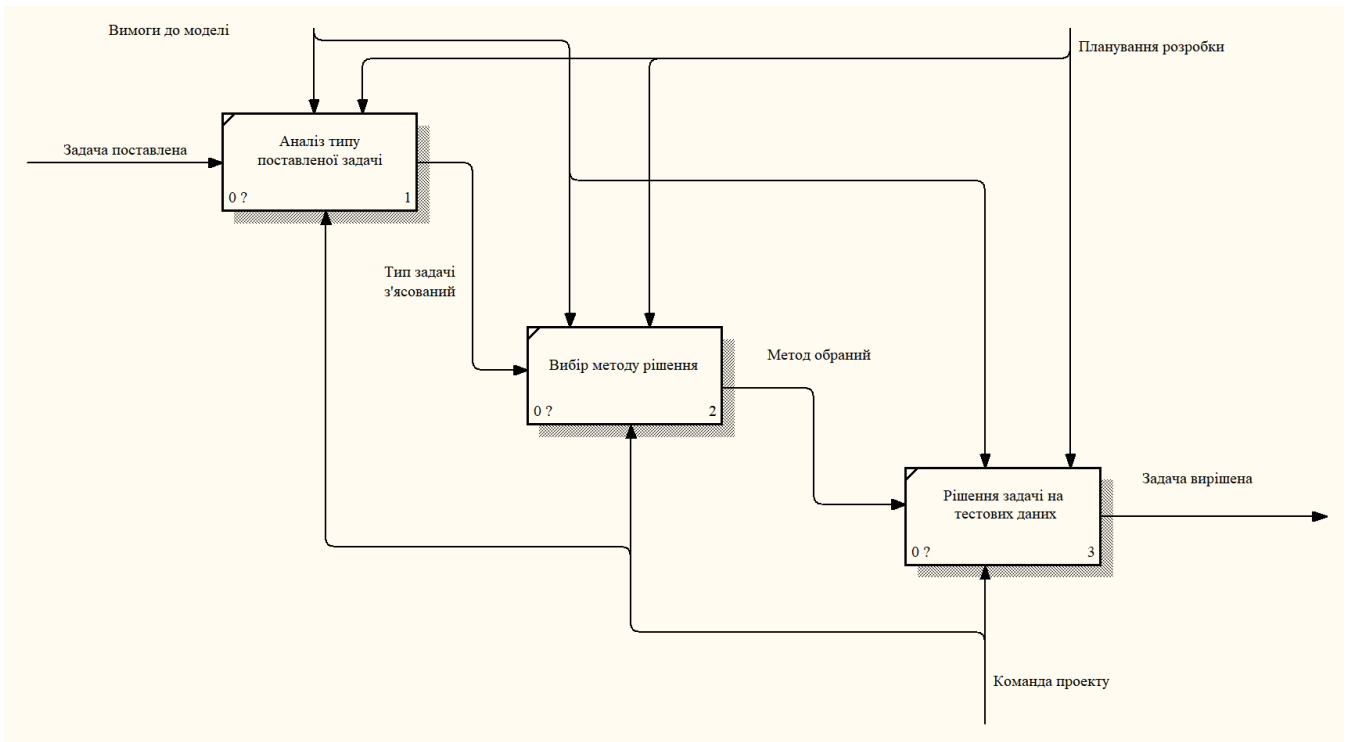


Рисунок 3.4 – Діаграма декомпозиції п'ятого етапу

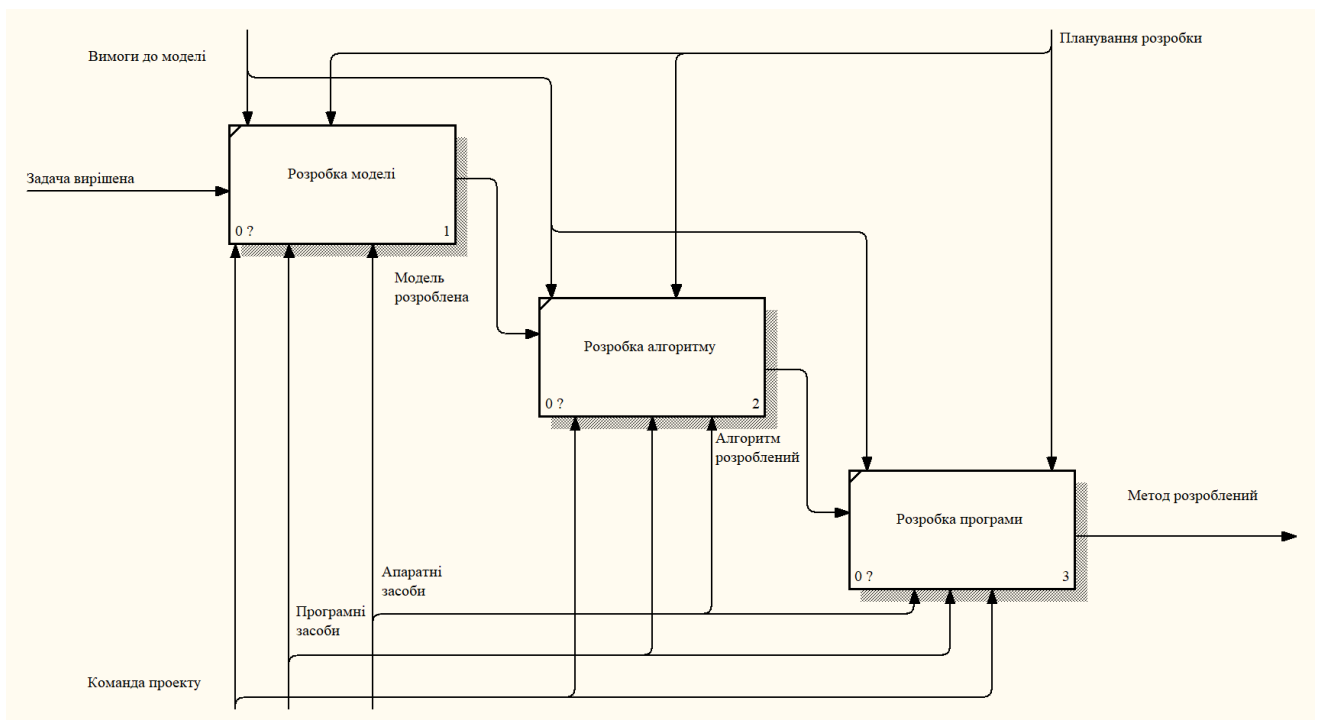


Рисунок 3.5 - Діаграма декомпозиції шостого етапу

### 3.2. Моделювання варіантів використання

Уніфікована мова моделювання (UML) визначає галузеві стандартні позначення та семантику для правильного застосування цієї нотації для програмного забезпечення, побудованого з використанням об'єктно-орієнтованої (ОО) або технології на основі компонентів. UML пропонує загальну та послідовну нотацію, за допомогою якої можна описати ОО та компоненти програмних систем, зменшуючи криву навчання для розробників, оскільки їм потрібно вивчити лише одну мову моделювання (принаймні в теорії).

Діаграма UML зазвичай використовується в програмній інженерії та інших бізнес-процесах, де моделювання є корисним.

Перед розробкою діаграми UML були визначені наступні актори:

Model – модель в цілому;

User- користувач, який використовує дану модель.

Далі наведений список варіантів використання. Він відповідає вимогам, яким повинна у свою чергу відповідати модель, що розробляється:

- формування цільової функції;
- аналіз поставленої задачі;
- аналіз типу поставленої задачі;
- рішення задачі на тестових даних;
- визначення рангів баз даних;
- формування обмежень цільової функції.

Реалізована діаграма UML (варіантів використання) зображена на рис. 3.6.

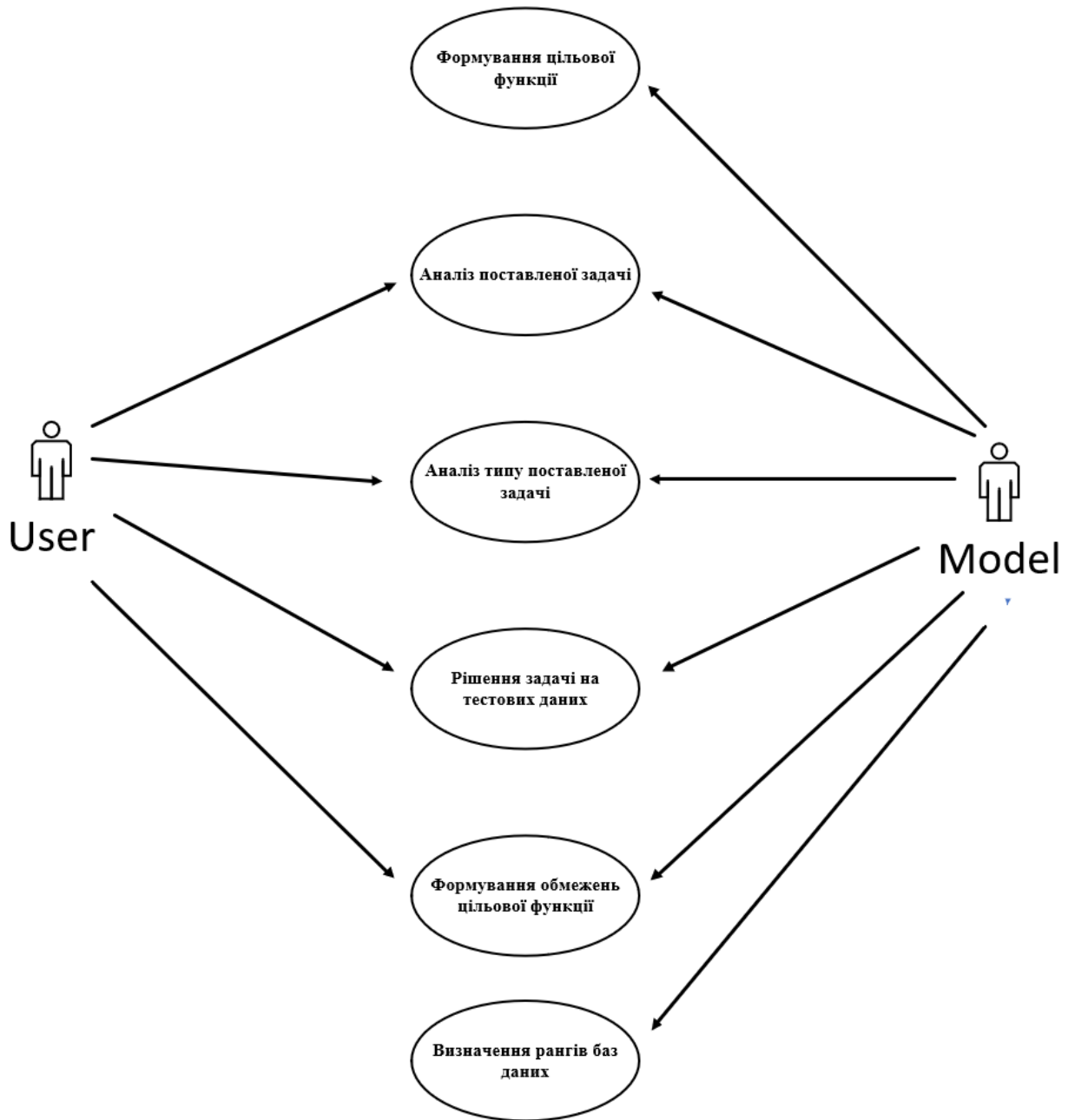


Рисунок 3.6 – Діаграма UML



## 4. Розробка моделей оцінювання організаційно-технічних заходів забезпечення надійності інформаційної системи в умовах небезпечних впливів

### 4.1. Загальні відомості про напівмарківські процеси

Напівмарковський процес (НМП) або, як його іноді називають, процес марковського відновлення поєднує в собі властивості марковських процесів і процесів відновлення. НМП - це такий випадковий процес, який описує переходи одного стану до іншого відповідно до заданих розподілами ймовірностей, а час перебування процесу у будь-якому стані є випадковою величиною, розподіл якої залежить як від цього стану, так і від стану, в яке буде здійснено наступний перехід процесу.

Процес відновлення, ланцюг Маркова з дискретним часом і однорідний марковський процес з безперервним часом є окремими випадками НМП. Так, процес відновлення - НМП з одним станом. Ланцюг Маркова з дискретним часом - НМП, у якого тривалості перебувань в кожному з станів рівні одиниці, а однорідний Марковський процес з безперервним часом є НМП, у якого час перебування в кожному стані має показовий розподіл, залежне лише від номера цього стану.

НМП визначається двовимірним ланцюгом Маркова:

$$(\xi_n, \theta_n), 0 \leq n \leq \infty, \xi_n \in E = \{1, 2, \dots, N\}, \theta_n \in [0; \infty).$$

Причому, розподілу  $P\{\theta_0 = 0\} = 1, p_i = P\{\xi_0 = i\} \geq 0, \sum_{i \in E} p_i = 1$ .

Ланцюг Маркова задається переходними можливостями, які називаються напівмарковським ядром:

$$P\{\xi_{n+1} = j, \theta_{n+1} < t | \xi_n = i, \theta_n < \tau\} = P\{\xi_{n+1} = j, \theta_{n+1} < t | \xi_n = i\} = Q_{ij}(t).$$

Будемо припускати, що ця перехідна ймовірність НЕ залежить від  $n$  (номера кроку), це означає однорідність ланцюга Маркова.

Напівмарковське ядро - ймовірність того, що НМП перейде в стан  $j$  та час до цього переходять не перевершить  $t$  за умови, що процес перебуває в стані  $i$ .

Властивості напівмарковських ядер  $P\{\xi_{n+1} = j, \theta_{n+1} < t | \xi_n = i\} = Q_{ij}(t)$ :

$$1) Q_{ij}(t) \geq 0, i, j \in E = \{1, 2, \dots, N\}.$$

$$2) p_{ij} = \lim_{t \rightarrow \infty} Q_{ij}(t) = P\{\xi_{n+1} = j | \xi_n = i\}, i, j \in E = \{1, 2, \dots, N\}.$$

$$3) \sum_{j \in E} P\{\xi_{n+1} = j, \theta_{n+1} < t\} = \sum_{j \in E} Q_{ij}(t) = P\{\theta_{n+1} < t | \xi_n = i\}, i \in E = \{1, 2, \dots, N\}.$$

$$4) F_{ij}(t) = P\{\theta_{n+1} < t | i, \xi_{n+1} = j\} = \frac{Q_{ij}(t)}{p_{ij}}, p_{ij} \neq 0, i, j \in E = \{1, 2, \dots, N\}.$$

5)  $Q_{ij}(t)$  являє собою невід'ємну функцією, аналогічно  $F_{ij}(t)$

$$6) \sum_{j \in E} p_{ij} = 1, i \in E = \{1, 2, \dots, N\}.$$

Ймовірності переходів в моменти стрибків визначаються стохастичною  $N \times N$  матрицею  $P = \|p_{ij}\|$ , яка задає вкладений ланцюг Маркова, де  $p_{ij}$ - ймовірність переходу зі стану  $i$  в стан  $j$ . Якщо процес зі стану  $i$  перейде в стан  $j$ , то час перебування процесу в стані  $i$  НЕ перевершує  $t$  і зі стану  $i$  процес переходить в стан  $j$ .

Послідовність побудови напівмарковських процесу.

1. Визначення марковського моментів.
2. Визначення станів процесу.
3. Обчислення напівмарковських ядер  $Q_{ij}(t)$ .
4. Перевірка властивостей напівмарковських ядер

## 4.2. Модель ранжування коефіцієнтів впливу елементів системи

Поставлену задачу ранжування приведено до принципу автоматичного сортування, використовуючи транзитивне замикання та нечітке відношення у вигляді основи моделі.

Використовуємо підходи, які спираються на причинно-спадковий зв'язок: зовнішній і внутрішній.

Зовнішній підхід спирається на реакцію надійності системи до зміни чутливості її елементів. Далі наведена функція чутливості:

$$P_s = f(P_1, P_2, \dots, P_n) \quad (1)$$

Описана вище функція поєднує показники ймовірності працездатної роботи системи ( $P_s$ ) та її елементів ( $P_i$ ). Перетворимо дану функцію у вигляді представлення ряду:

$$P_s = b_0 + \sum_{i=1}^n b_i P_i + \sum_{i=1}^n b_{ij} P_i P_j + \dots \quad (2)$$

Варто зазначити, що коефіцієнти даного ряду є частковими похідними:

$$b_i = \frac{dP_s}{dP_i}, b_{ij} = \frac{d^2 P_s}{dP_i dP_j} \quad (3)$$

Коефіцієнт  $b_i$  визначає коефіцієнт впливу поточного елемента, а також відповідає коефіцієнту важливості мінімального впливу  $i$ -го і  $j$ -го елементів. Обчислення важливості елементів системи, для якої проводиться ранжування на основі структурного функціонування виглядає як:

$$\alpha_s = f_L(\alpha_1, \alpha_2, \dots, \alpha_n) \quad (4)$$

де  $\alpha_s$  ( $\alpha_i$ ) = 1 (0), якщо система ( $i$ -й елемент) має працездатний (не працездатний стан),  $f_L$  - булева функція.

Внутрішній підхід використовує як основу структурну інформацію системи, а саме структуру елементів. Таким чином виникає можливість використовувати інформацію про небезпечних вплив у одному елементі та виявлення небезпечного впливу у іншому елементі. Наприклад, «аналіз параметрів  $i$ -го елемента підштовхує систему до зникнення параметрів  $j$ -го елемента, і таким чином  $k$ -го елемент починає перебувати у непрацездатному стані». Тому потрібно також враховувати «ефект доміно». Матриця зв'язків визначення рангів система є джерелом інформації.

$$A = [a_{ij}], i, j = 1, 2, \dots, n \quad (5)$$

У матриці,  $a_{ij} = 1$  (0), якщо один елемент має зв'язок (не має зв'язку) з іншим елементом. Ранг одного елемента визначається у вигляді суми елементів цього (наприклад першого) рядка матриці

$$D = A + A^2 \quad (6)$$

Ранг матриці також враховує однокрокові (двокрокові) небезпечні впливи в поточному елементі системи.

Далі запропоноване розв'язання задачі ранжування елементів на основі нечіткого відношення подібності і транзитивного замикання.

Нехай  $\{x_1, x_2, \dots, x_n\}$  – множина елементів інформаційної системи. Значення впливу елемента  $x_i \in X$  на інші елементи позначаємо у вигляді нечіткої множини:

$$I_i = \left\{ \frac{\mu_{i1}}{x_1}, \frac{\mu_{i2}}{x_2}, \dots, \frac{\mu_{in}}{x_n} \right\} \quad (7)$$

де  $\mu_{ij}$ - число в інтервалі  $[0,1]$ , що характеризує рівень впливу елемента  $x_i \in X$  на елемент  $x_j \in X$ ;  $i, j = 1, 2, \dots, n$ .

Зробимо припущення, що вплив елемента  $x_i \in X$  на  $x_i$  дорівнює нулю, тобто:

$$\mu_{ii} = 0, i = 1, 2, \dots, n. \quad (8)$$

Множина ступенів впливу  $\mu_{ij}$  з (7) для всіх елементів  $x_i \in X$  визначає значення нечіткого відношення впливу  $I$ , яке задається на декартовому добутку  $X \times X$ , тобто  $I \subset X \times X$ :

$$I = \left[ \frac{\mu_{ij}}{(x_i, x_j)} \right], i, j = 1, 2, \dots, n \quad (9)$$

Число  $\mu_{ij}$ , яке відповідає будь-якій парі елементів  $(x_i, x_j)$ , може задаватися або експертною групою, або методом найменшого впливу.

Зауважимо, що дану процедуру обчислення ступенів приналежностей потрібно обчислювати з високою точністю.

Якщо  $f_{ij}$  – метрика впливу елемента системи  $x_i \in X$  на елемент  $x_j \in X$ , тоді виконується умова: «чим більше метрика  $f_{ij}$ , тим більше рівень ступеню впливу  $\mu_{ij}$ », тобто:

$$\frac{\mu_{i1}}{f_{i1}} = \frac{\mu_{i2}}{f_{i2}} = \dots = \frac{\mu_{il}}{f_{il}} = \dots = \frac{\mu_{in}}{f_{in}} \quad (10)$$

Також варто зазначити, що в залежності з (8):

$$f_{ii} = 0, i = 1, 2, \dots, n \quad (11)$$

Нехай  $x_l$  – елемент системи, на який елемент  $x_i$  має мінімальний вплив, то разом із (10) отримуємо:

$$\mu_{ij} = \mu_{il} \frac{f_{i1}}{f_{il}}, \mu_{i2} = \mu_{il} \frac{f_{i2}}{f_{il}}, \dots, \mu_{ij} = \mu_{il} \frac{f_{in}}{f_{il}} \quad (12)$$

Виконуючи підстановку у (12), отримуємо:

$$\mu_{ij} + \mu_{ij} + \dots + \mu_{ij} = 1, i = 1, 2, \dots, n,$$

У результаті обчислюємо ступінь найменшого впливу елемента  $x_i \in X$  у системі:

$$\mu_{i1} = \left( \frac{f_{i1}}{f_{il}} + \frac{f_{i1}}{f_{il}} + \dots + \frac{f_{in}}{f_{il}} \right)^{-1} \quad (13)$$

Формули (12) та (13) дають право на обчислення ступеня впливу у нечіткому відношенні подібності (9) у вигляді порівняння рівнів впливів  $f_{ij}$  з найменшою метрикою впливу  $f_{il}$  для кожного елемента  $x_i \in X$ . Для цього користуємося 9-бальною шкалою Сааті:

$$\frac{f_{ii}}{f_{il}} = 1,3,5,7,9, \quad (14)$$

Якщо вплив « $ij$ » (елемента  $x_i$  на елемент  $x_j$ ) по порівнянню з найменшим впливом « $il$ » (елемента  $x_i$  на елемент  $x_l$ ): 1 – вплив рівномірний, 3 - вплив більше рівномірного, але не занадто, 5 - вплив великий, 7 -вплив дуже великий, 9 - вплив дуже великий та більше оцінки 7 (також існує можливість використовувати оцінки 8,6,4,2).

Метрику подібності за ступенем впливу між елементами  $x_i \in X$  і  $x_j \in X$  обчислюємо значенням  $r_{ij}$ :

$$r_{ij} = 1 - d_{ij} \quad (15)$$

де  $d_{ij}$  - відстань між нечіткими множинами впливу елементів  $x_i$  і  $x_j$ :

$$I_i = \left\{ \frac{\mu_{i1}}{x_1}, \frac{\mu_{i2}}{x_2}, \dots, \frac{\mu_{in}}{x_n} \right\}, I_j = \left\{ \frac{\mu_{j1}}{x_1}, \frac{\mu_{j2}}{x_2}, \dots, \frac{\mu_{jn}}{x_n} \right\}$$

Для обчислення  $d_{ij}$  використовуємо відстань за Хеммінгом ( $d_{ij}^{(h)}$ ) або за Евклідом ( $d_{ij}^{(e)}$ ):

$$d_{ij}^{(h)} = \frac{1}{n} \sum_{k=1}^n |\mu_{ik} - \mu_{jk}| \quad (16)$$

$$d_{ij}^{(e)} = \frac{1}{n} \sqrt{\sum_{k=1}^n (\mu_{ik} - \mu_{jk})^2} \quad (17)$$

Множина значень  $r_{ij}$  для всіх пар елементів системи  $(x_i, x_j) \in X \times X$  складає нечітке відношення подібності  $R \subset X \times X$ :

$$R = \left[ \frac{r_{ij}}{x_i, x_j} \right] \quad (18)$$

Якому притаманні властивості:

- рефлексивність, тобто  $r_{ij} = 1$ , для всіх  $x_i \in X$ ,
- симетричність, тобто  $r_{ij} = r_{ji}$ , для всіх  $x_i \in X, x_j \in X$ .

Для розкладання множини  $X$  на непересічні класи елементів системи, схожих за значенням ступенем впливу, надаємо вихідному нетранзитивному відношенню подібності  $R$  властивість транзитивності. Дане перетворення виконується процедурою транзитивного замикання нечіткого відношення.

### **4.3. Ранжування елементів інформаційної системи баз даних для забезпечення надійності на прикладі створеної системи**

Перед реалізацією інформаційної технології ранжування інформаційної системи, пропонується виконати ранжування у писемному вигляді. Нехай постановка задачі буде наступна:

Існує задача розбиття фінансових обмежених ресурсів на забезпечення працездатності інформаційної системи управління транспортом, наприклад залізничним. Змодельована система наведена на рис 4.2.1.

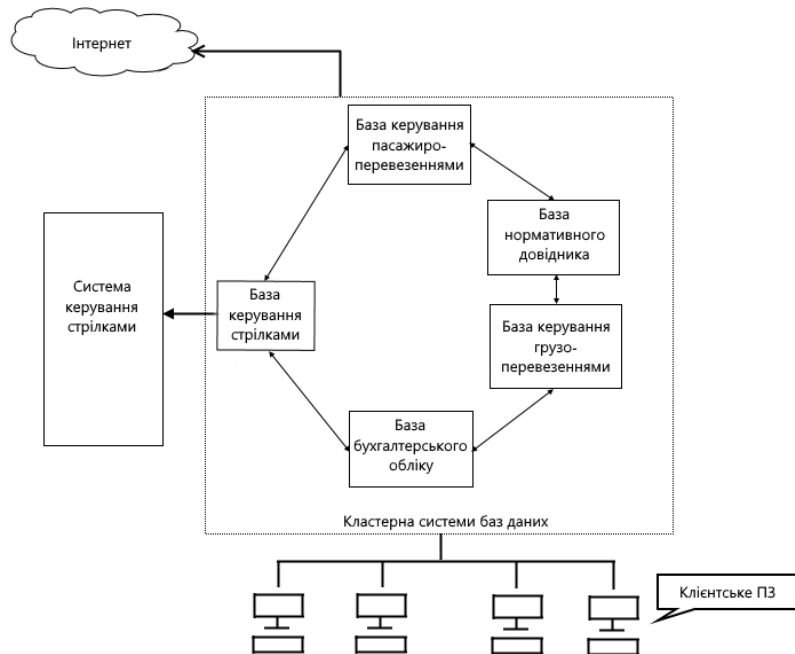


Рисунок 4.2.1 – Модель баз даних поставленої задачі

Тобто описується приватна система, у склад якої входять п'ять баз даних інформаційної системи. Експертна інформація, необхідна для обчислення відношення методом найменшого впливу, представлена у таблиці 3.2.1. Кожному значенню  $x$  відповідає своя база даних.

Таблиця 4.2.1 – Вихідні дані для методу найменшого впливу

| $x_i$ | $x_l$ | $ij/ i_l$ |       |       |       |       |
|-------|-------|-----------|-------|-------|-------|-------|
|       |       | $x_1$     | $x_2$ | $x_3$ | $x_4$ | $x_5$ |
| $x_1$ | $x_2$ | 0         | 1     | 5     | 3     | 1     |
| $x_2$ | $x_1$ | 1         | 0     | 9     | 3     | 1     |
| $x_3$ | $x_5$ | 9         | 3     | 0     | 2     | 1     |
| $x_4$ | $x_2$ | 1         | 1     | 7     | 0     | 3     |
| $x_5$ | $x_1$ | 1         | 5     | 9     | 7     | 0     |



У стовпці таблиці  $x_l$  знаходяться елементи, які мають мінімальний вплив на інші бази даних, наприклад:  $x_5$  має мінімальний вплив на  $x_1$ ,  $x_4$  має мінімальний вплив на  $x_2$ , і т.д. Дану інформацію визначає експертна група.

Використовуючи дані з вихідних даних для методу найменшого впливу (табл. 4.2.1) і формулами, визначаємо кількісну характеристику ступеней впливу  $\mu_{ij}$  для відношення. Варто зазначити, що  $\mu_{ii} = 0$ :

$$\mu_{11} = 0$$

$$\mu_{12} = \left( \frac{f_{11}}{f_{12}} + \frac{f_{12}}{f_{12}} + \frac{f_{13}}{f_{12}} + \frac{f_{14}}{f_{12}} + \frac{f_{15}}{f_{12}} \right)^{-1} = \frac{1}{0 + 1 + 5 + 3 + 1} = \frac{1}{10}$$

$$\mu_{13} = \mu_{12} \frac{f_{13}}{f_{12}} = \frac{1}{10} 5 = \frac{5}{10}$$

$$\mu_{12} = \mu_{12} \frac{f_{14}}{f_{12}} = \frac{1}{10} 3 = \frac{3}{10}$$

$$\mu_{12} = \mu_{12} \frac{f_{15}}{f_{12}} = \frac{1}{10} 1 = \frac{1}{10}$$

За прикладом продовжуємо визначення ступеней приналежностей, які разом створюють таблицю значень нечітких відношень:

Таблиця 4.2.2 – значення нечітких відношень

|       | $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ |          |
|-------|-------|-------|-------|-------|-------|----------|
| $x_1$ | 0     | 1/10  | 5/10  | 3/10  | 1/10  | max=5/10 |
| $x_2$ | 1/14  | 0     | 9/14  | 3/14  | 1/14  | max=9/14 |
| $x_3$ | 9/15  | 3/15  | 0     | 2/15  | 1/15  | max=9/15 |
| $x_4$ | 1/12  | 1/12  | 7/12  | 0     | 3/12  | max=7/12 |
| $x_5$ | 1/22  | 5/22  | 9/22  | 7/22  | 0     | max=9/22 |

Після цього виконує нормалізацію значень нечітких відношень, тобто ділимо елементи на максимальне значення кожного рядка і таким чином створюємо таблицю 4.2.3.

Таблиця 4.2.3 – нормалізоване значення нечітких відношень

|       | $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ |
|-------|-------|-------|-------|-------|-------|
| $x_1$ | 0     | 0.2   | 1     | 0.6   | 0.2   |
| $x_2$ | 0.11  | 0     | 1     | 0.33  | 0.11  |
| $x_3$ | 1     | 0.33  | 0     | 0.22  | 0.11  |
| $x_4$ | 0.14  | 0.14  | 1     | 0     | 0.43  |
| $x_5$ | 0.11  | 0.56  | 1     | 0.78  | 0     |

Наступним етапом обчислюємо значення нечіткого відношення подібності. Результат наведений у таблиці 4.2.4.

Таблиця 4.2.4 – нечітке відношення подібності

|       | $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ |
|-------|-------|-------|-------|-------|-------|
| $x_1$ | 1     | 0.87  | 0.48  | 0.76  | 0.83  |
| $x_2$ | 0.87  | 1     | 0.53  | 0.83  | 0.44  |
| $x_3$ | 0.48  | 0.53  | 1     | 0.48  | 0.44  |
| $x_4$ | 0.79  | 0.83  | 0.48  | 1     | 0.67  |
| $x_5$ | 0.83  | 0.78  | 0.44  | 0.67  | 1     |

Далі обчислюємо значення ступеней приналежностей використовуючи відстань по Хеммінгу:

$$\begin{aligned}
 d_{12} &= \frac{1}{5} [(0,0,2,,1,0.6,0.2) - (0.11,0.1,0.33,0.1)] = \\
 &= \frac{1}{5} [|0 - 0,11| + |0,2 - 0| + |1 - 1| + |0,6 - 0,33| + |0,2 - 0,1|] \\
 &= \frac{1}{5} [0,11 + 0,1 + 0 + 0,27 + 0,09] = 0,13
 \end{aligned}$$

У результаті використання нечіткого відношення подібності, отримуємо таблицю значень транзитивного відношення (таблиця 3.2.5):

Таблиця 4.2.5 – Обчислення значення транзитивного замикання відношення подібності

|       | $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ |
|-------|-------|-------|-------|-------|-------|
| $x_1$ | 1     | 0,87  | 0,53  | 0,83  | 0,83  |
| $x_2$ | 0,87  | 1     | 0,53  | 0,83  | 0,83  |
| $x_3$ | 0,53  | 0,53  | 1     | 0,53  | 0,53  |
| $x_4$ | 0,83  | 0,83  | 0,53  | 1     | 0,79  |
| $x_5$ | 0,83  | 0,83  | 0,53  | 0,79  | 1     |

Після визначення транзитивного замикання ми бачимо, що воно має вигляд:

$$R_{ij} = \max(\min(r_{i1}, r_{1j}), \min(r_{i2}, r_{2j}), \dots, \min(r_{in}, r_{nj}))$$

$$\hat{R} = R \cup R^2 \cup R^3 \cup \dots \cup R^k \cup \dots = R^3,$$

А саме збігається з нашим відношенням. Важливим, але не останнім етапом являється визначення значення рядків матриці. Як результат, отримуємо кількісні значення рангів для кожного елементу системи:

$$p_1 = 0 + 0,87 + 0,53 + 0,83 = 4,06$$

$$p_2 = 4.06, p_3 = 3.12, p_4 = 4.02, p_5 = 4.02$$

Нечітке відношення (26) розкладаємо по  $\alpha$ -рівням визначеності:

$$\hat{R} = \bigcup_{\alpha} \alpha R_{\alpha} = 0,53 R_{0,53} \bigcup 0,83 R_{0,83} \bigcup 0,87 R_{0,87} \bigcup R_1$$

Елемент  $x_i$  позначаємо у вигляді  $i$ ,  $i=1,2,\dots, 5$ . Чіткі відношення  $\alpha$ -рівня визначеності формують класи елементів, які були відсортовані за ступенем важливістю (таблиця. 4.2.6).

Таблиця 4.2.6 – Класи елементів, еквівалентних по важливості

| Рівень          | Число класів | Класи елементів                               |
|-----------------|--------------|---|
| $\alpha = 0,53$ | 1            | $\{x_1, x_2, x_3, x_4, x_5\}$                 |
| $\alpha = 0,83$ | 2            | $\{x_1, x_2, x_4, x_5\}, \{x_3\}$             |
| $\alpha = 0,87$ | 4            | $\{x_1, x_2\}, \{x_3\}, \{x_4\}, \{x_5\}$     |
| $\alpha = 1$    | 5            | $\{x_1\}, \{x_2\}, \{x_3\}, \{x_4\}, \{x_5\}$ |

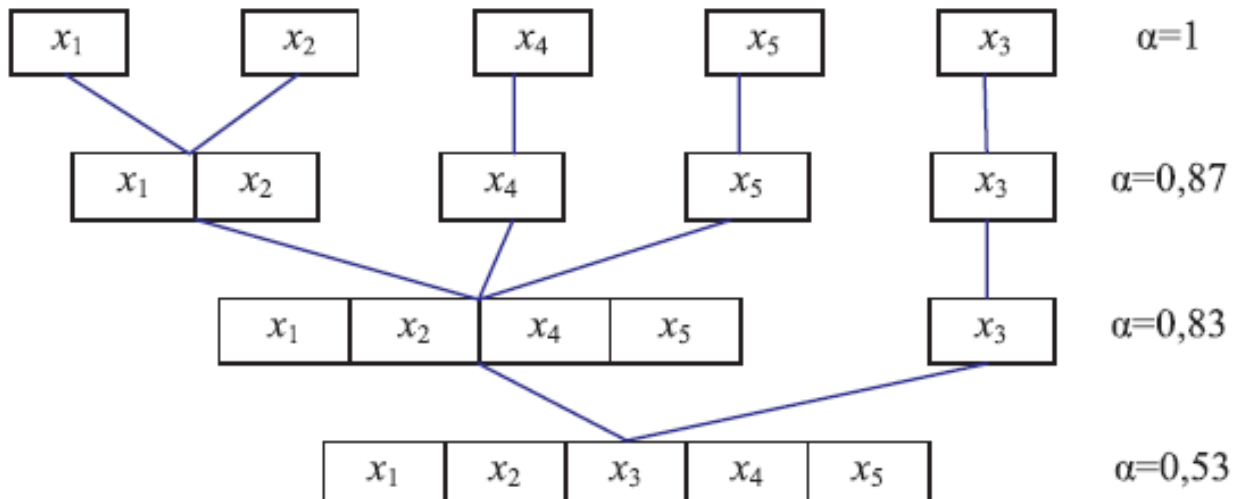


Рисунок 4.2.3 – Дерево декомпозиції на класи еквівалентності

Число  $\alpha$  може інтерпретуватися як рівень визначеності наших знань про систему, а  $(1 - \alpha)$  - рівень невизначеності. Тобто чим складніше система і чим більше число реалій не враховується при моделюванні, тим більше невизначеність і нижче число  $\alpha$ .

У відповідності з результатами обчислення (рис. 4.2.3) зрозуміло,

Що при рівні визначеності  $\alpha = 1$  будь-який з елементів системи (бази даних) становить свій особливий рівень важливості. Але при значенні  $\alpha = 0,53$  ця особливість зникає (усі елементи системи мають один і той же ранг). Але якщо використовувати рівень  $\alpha = 0,83$  як базис, отримуємо дохідливі значення кількісних оцінок:

$$p_3 = 3, p_1 = p_2 = p_4 = p_5 = 4,;$$

Якщо  $C_0$  – це обмежені фінансові ресурси, які виділяють на забезпечення надійності баз даних, тоді підставимо отримані кількісні оцінки до обчислених рангів елементів системи:

$$\sum_{i=1}^5 C_i = C_0, \quad C_1 = C_2 = C_4 = C_5 = \frac{4}{19} C_0, \quad C_3 = \frac{3}{19} C_0$$

#### **4.4. Інформаційна технологія ранжування елементів інформаційної системи на основі транзитивного замикання**

Перед створенням програмного забезпечення потрібно узгодити основні функціональні вимоги і вигляд прототипу. Основними вимогами до програми є:

- введення кількості елементів аналізованої інформаційної системи;
- введення коефіцієнтів впливу баз даних у матрицю оцінок впливу;
- введення коефіцієнтів мінімального впливу до потрібної таблиці;
- виведення результатів у вигляді коефіцієнтів та класів, до яких вони відносяться;
- введення можливого значення допустимих витрат на забезпечення надійності інформаційної системи;
- можливість очищення введених даних для нового обчислення коефіцієнтів;
- розмістити замітку щодо використання інформаційної технології головному вікні у вигляді кнопки.

Після етапу узгодження вимог, розробник інформаційної технології приступає до виконання ескізу головного вікна. Ескіз вікна зображений на рис. 4.3.1.

Вибір кількості БД

Очистить

Расчитать

Таблиця мінімального впливу

Матриця оцінок впливу

Таблиця коефіцієнтів та класів, до яких вони відносяться

Вибір значення витрат

Таблиця результатів

Автор

Рисунок 4.3.1 – Прототип інформаційної технології

Наступний кроком являється вибір основних компонентів програмного забезпечення, що розробляється. Так як було обрано використовувати мову програмування C#, то доцільно використовувати у вигляді основи проект Windows Forms. Далі наведені усі обрані компоненти програми (таблиця 4.3.1).

Таблиця 4.3.1 – Таблиця обраних компонентів

| №  | Компонент     | Детальний опис компоненту   |
|----|---------------|---|
| 1. | Button        | Елемент кнопки, після натискання виконується потрібна процедура (очищення даних, обчислення даних за методом ранжування або загальна підказка щодо користування програмою |
| 2. | Label         | Компонент, який підказує користувачу, за що відповідає підписаний компонент вікна   |
| 3. | NumericUpDown | Елемент поля введення числового значення із допоміжними стрілками, що можуть змінити введене значення на 1 (  |
| 4. | DataGridView  | Елемент таблиці, що відповідає матрицям введення або отримання результатів  |

Після визначення основних компонентів, розробник приступив до розроблення основного вікна у середовищі програмування Visual Studio. Робочий прототип головного вікна зображений на рис. 4.3.2



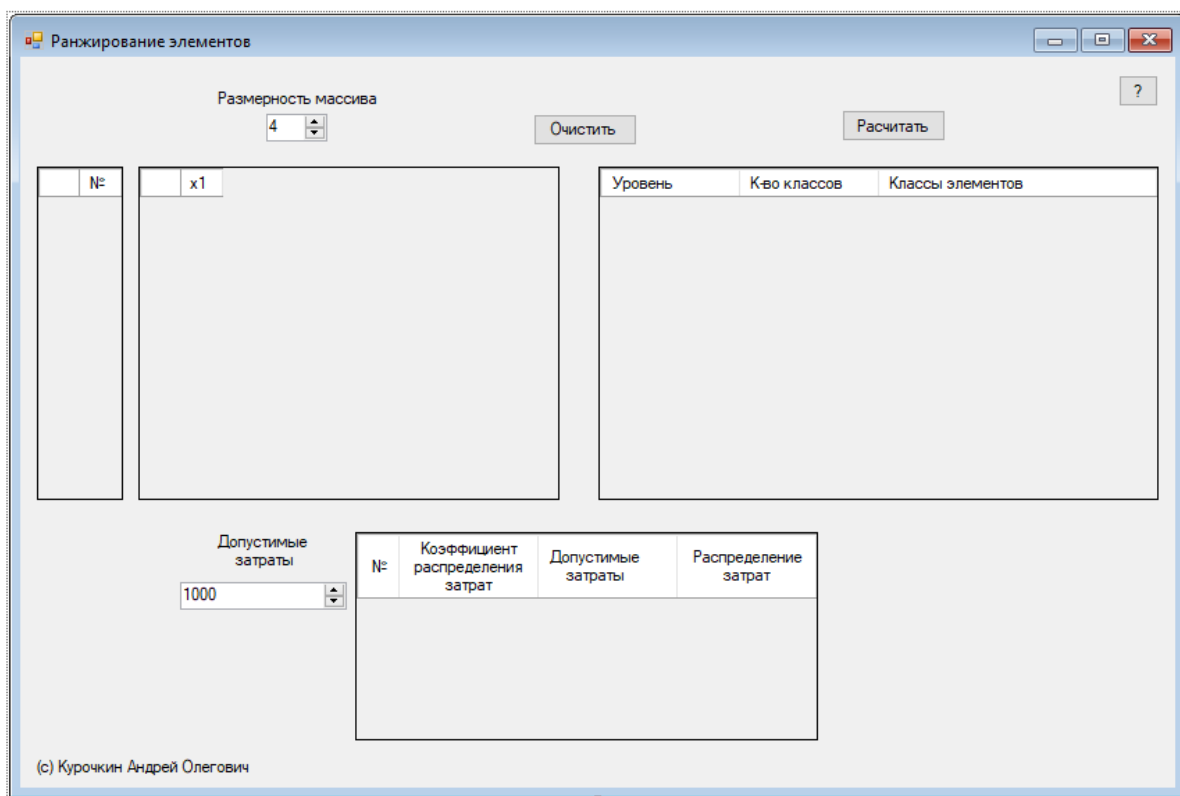


Рисунок 4.3.2 – Прототип головного вікна

Передостаннім етапом являється реалізація логіки методу ранжування згідно визначених раніше етапів обчислення коефіцієнтів впливу за допомогою мови програмування C#. Детальний код реалізації наведений у додатку Б. Далі наведений опис реалізованих функцій програмного забезпечення (таблиця 4.3.2).

Таблиця 4.3.2. – Опис реалізовано лістингу коду

| №  | Функція (назва) | Детальний опис функції                      |
|----|-----------------|---|
| 1. | resource()      | Обчислення значень впливу елементів таблиці |

## Продовження таблиці 4.3.2.

|     |                           |  |
|-----|---------------------------|--|
| 2.  | RT2                       | Створення обмежень для обраного масиву   |
| 3.  | Array_check               | Перевірка масиву числових значень на коректність введення чисел  |
| 4.  | Class_Evaluation          | Обчислення значень рівнів визначеності   |
| 5.  | Compos                    | Визначення нечіткої композиції ступеней приналежності  |
| 6.  | Form1()                   | Ініціалізація програми та кастомізація стилю таблиць   |
| 7.  | Normalization             | Виконання нормалізації коефіцієнтів впливу баз даних   |
| 8.  | fuzzy_sim                 | Обчислення відстані Хеммінга для отримання ступеней приналежності  |
| 9.  | value_size_ValueChanged   | Динамічна зміна розмірності матриці вхідних значень. При зміні розмірності, елемент DataGridView одразу змінює свій розмір |
| 10. | datagridView_RowPostPaint | Видаляє елемент DataGridView, а саме стрілку, яка зображує рядок, який обирає користувач                                   |
| 11. | button1_Click             | При натисненні виконується алгоритм ранжування   |

## Продовження таблиці 4.3.2.

|     |                             |  |
|-----|-----------------------------|--|
| 12. | button2_Click               | Викликає вікно із інструкцією щодо використання програмою            |
| 13. | Min_i_CellContentClick      | Зв'язує значення таблиці вхідних даних і таблиці мінімального впливу |
| 14. | Button3_Click               | Відповідає за очищення даних   |
| 15. | numericUpDown1_ValueChanged | Динамічна зміна допустимих витрат у таблиці результатів              |
| 16. | Button4_Click               | Кнопка автоматичного заповнення таблиці вхідних даних                |

Інформаційна технологія реалізована згідно всіх поставлених вимог та проведена серія комп'ютерних експериментів. Наведено приклад роботи з програмою включаючи реакцію ПЗ на введення некоректних даних. Детальні комп'ютерні експерименти наведені у додатку D.

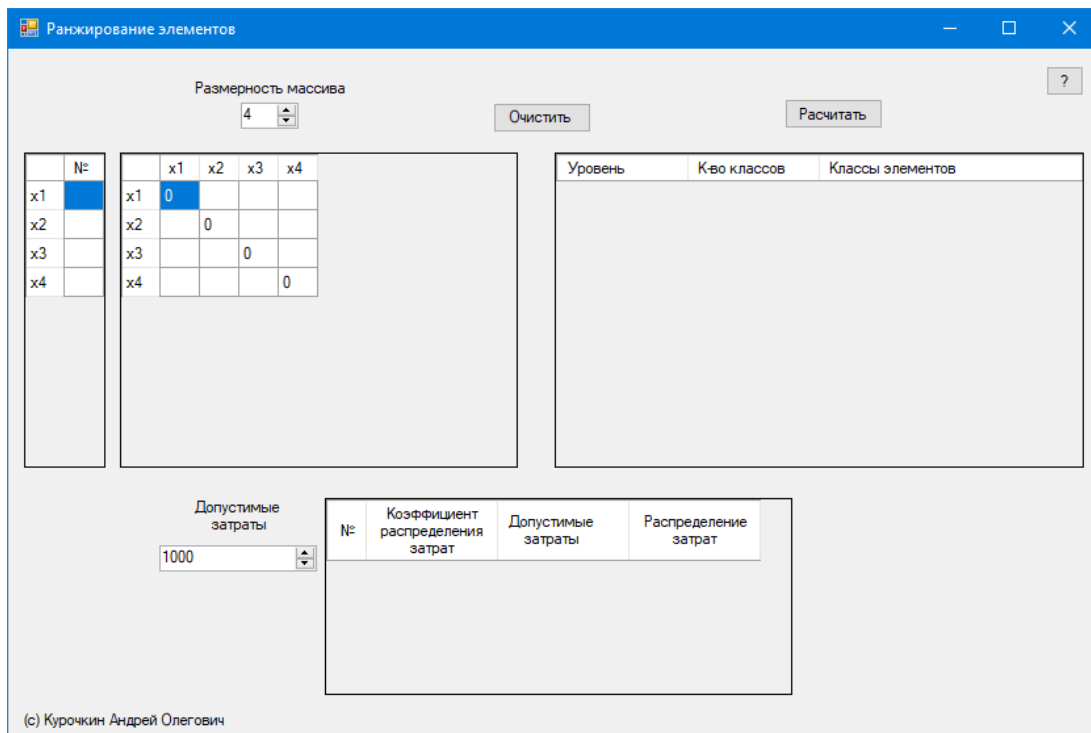


Рисунок 4.3.3 – Основне вікно інформаційної технології

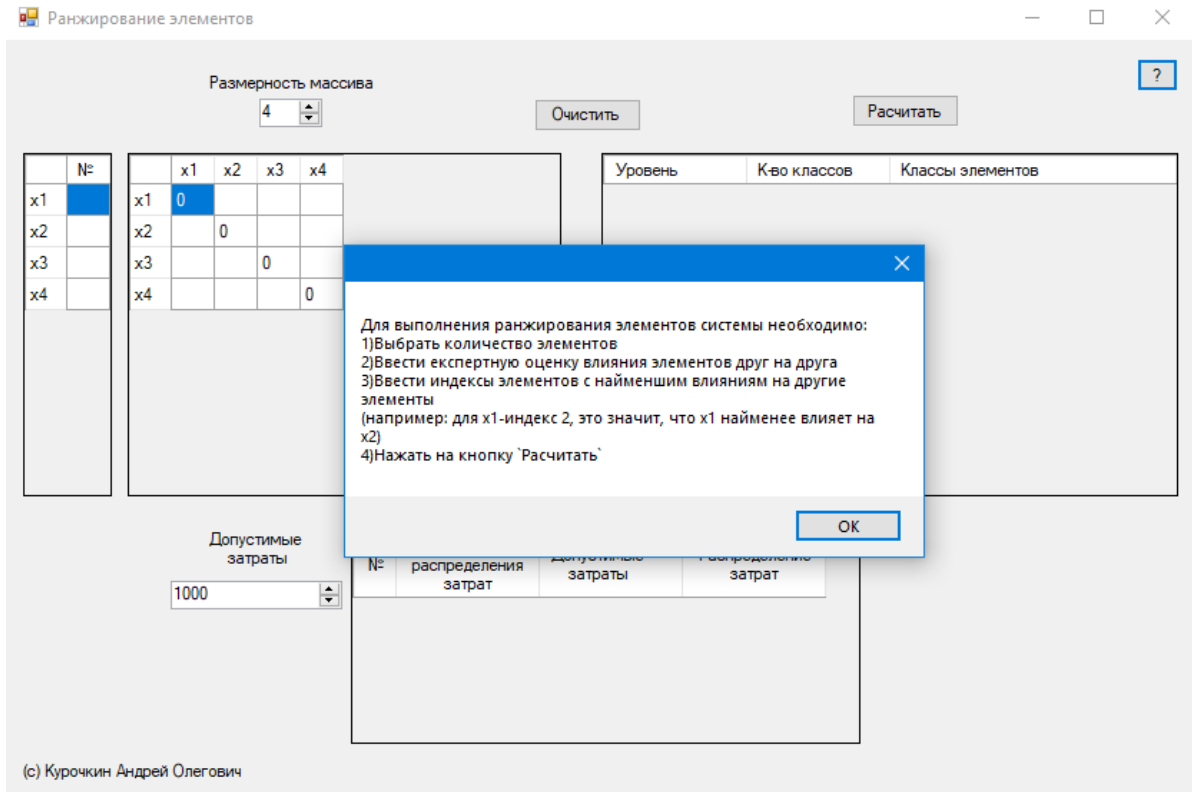


Рисунок 4.3.4 – Інструкція щодо користування ПЗ

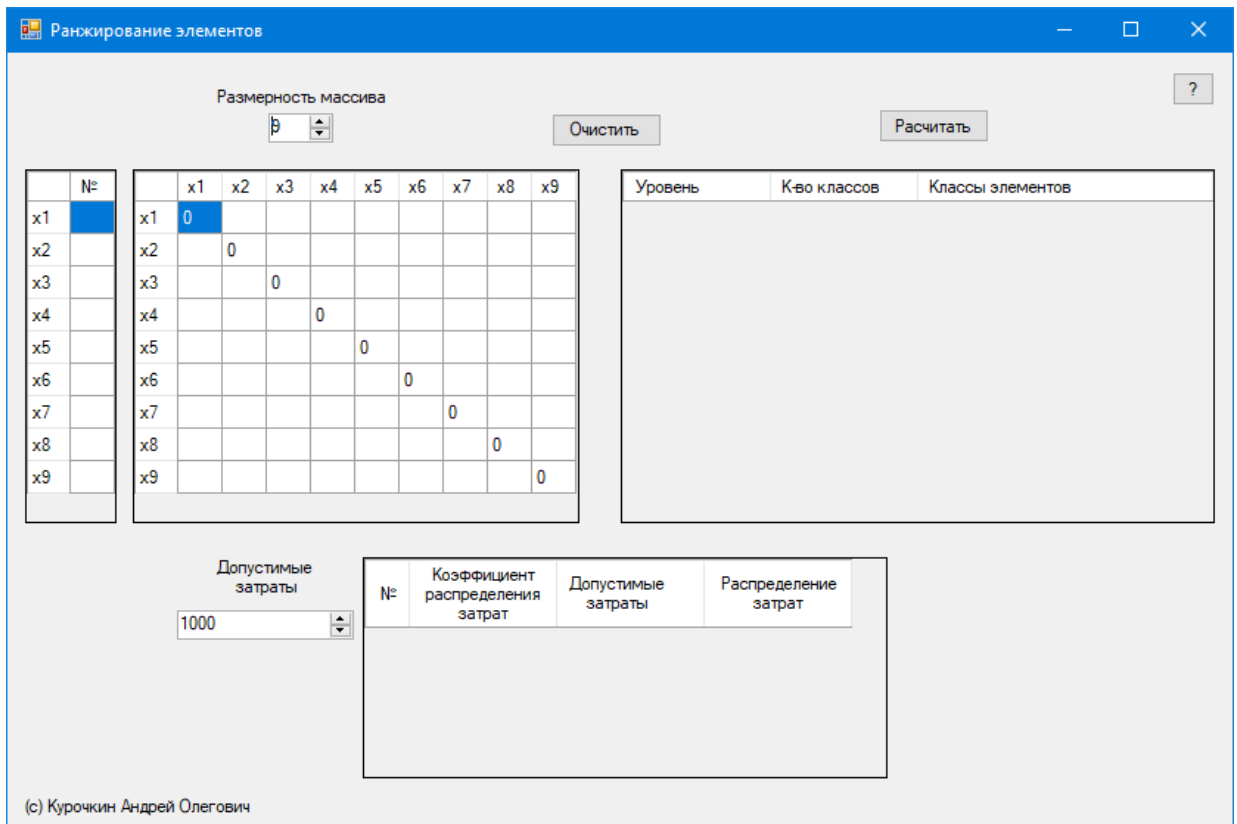


Рисунок 4.3.5 – Процес зміни розмірності масиву

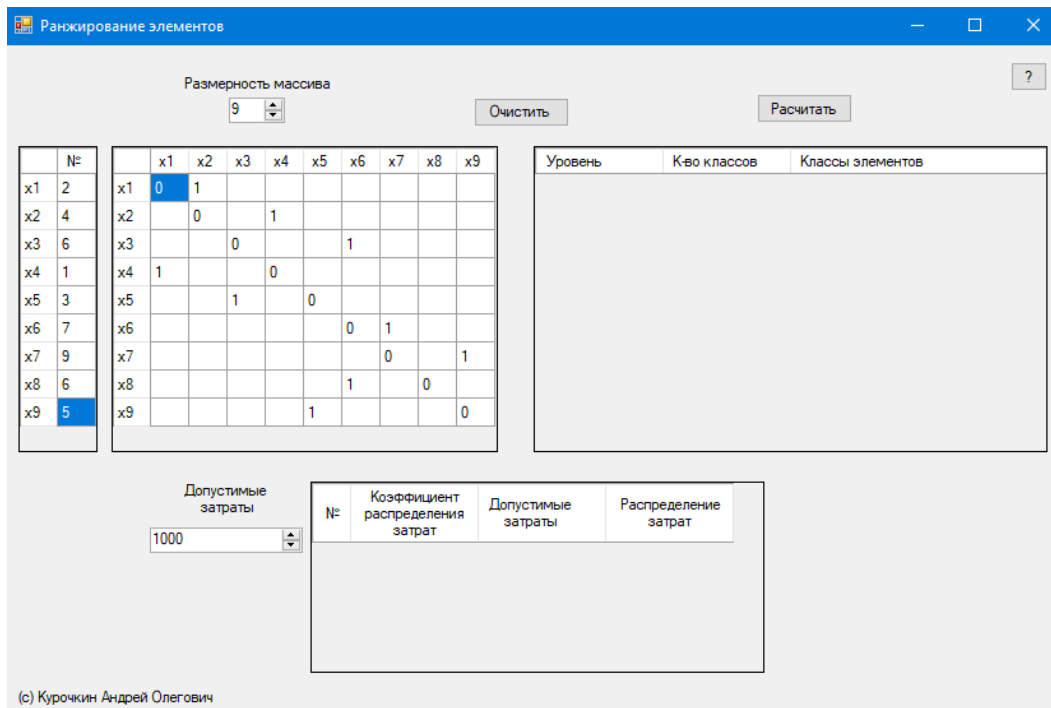


Рисунок 4.3.6 – Заповнення таблиці мінімального впливу

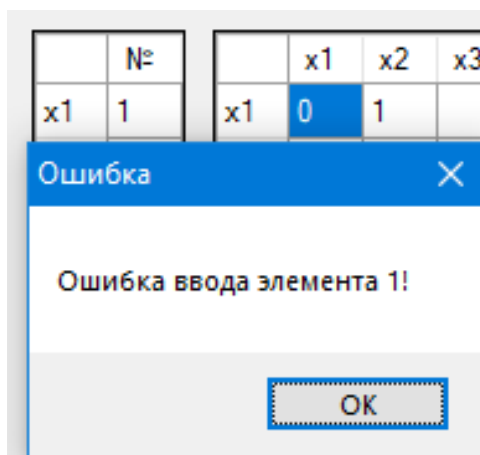


Рисунок 4.3.7 – Введення некоректного числа у таблицю мінімального впливу

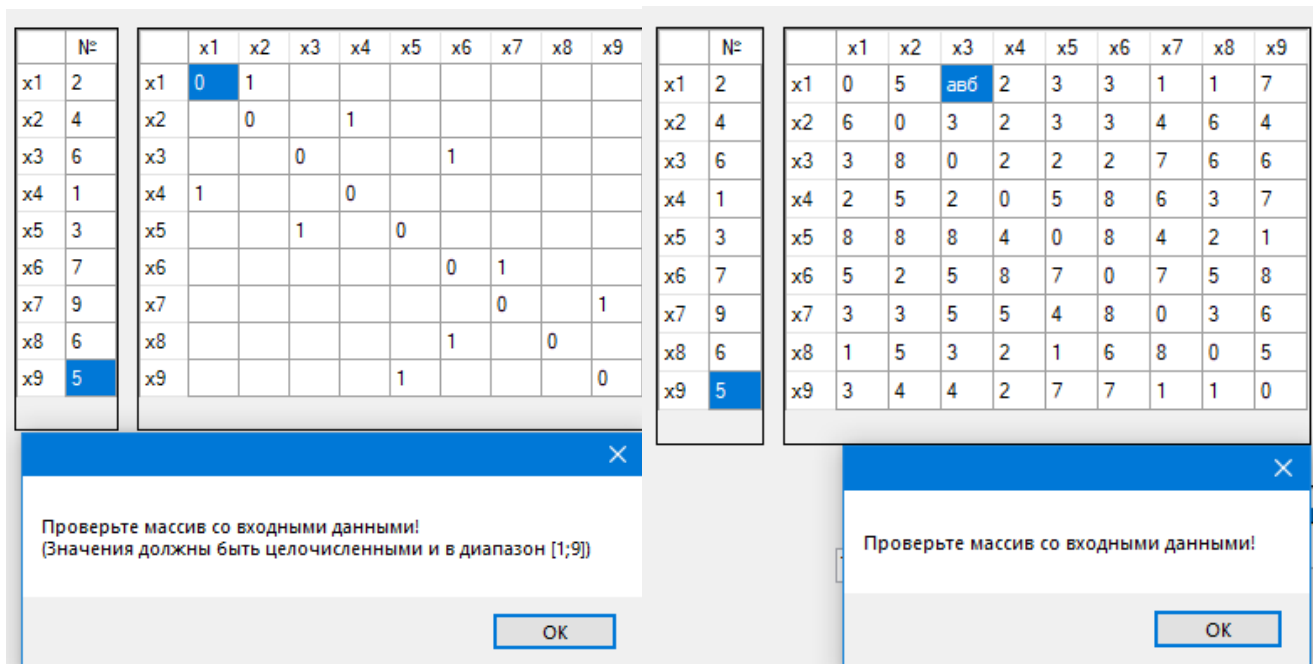


Рисунок 4.3.8 – Выведення повідомлень помилки при обчислення методу ранжування при некоректних даних (не заповнення та строкові значення)

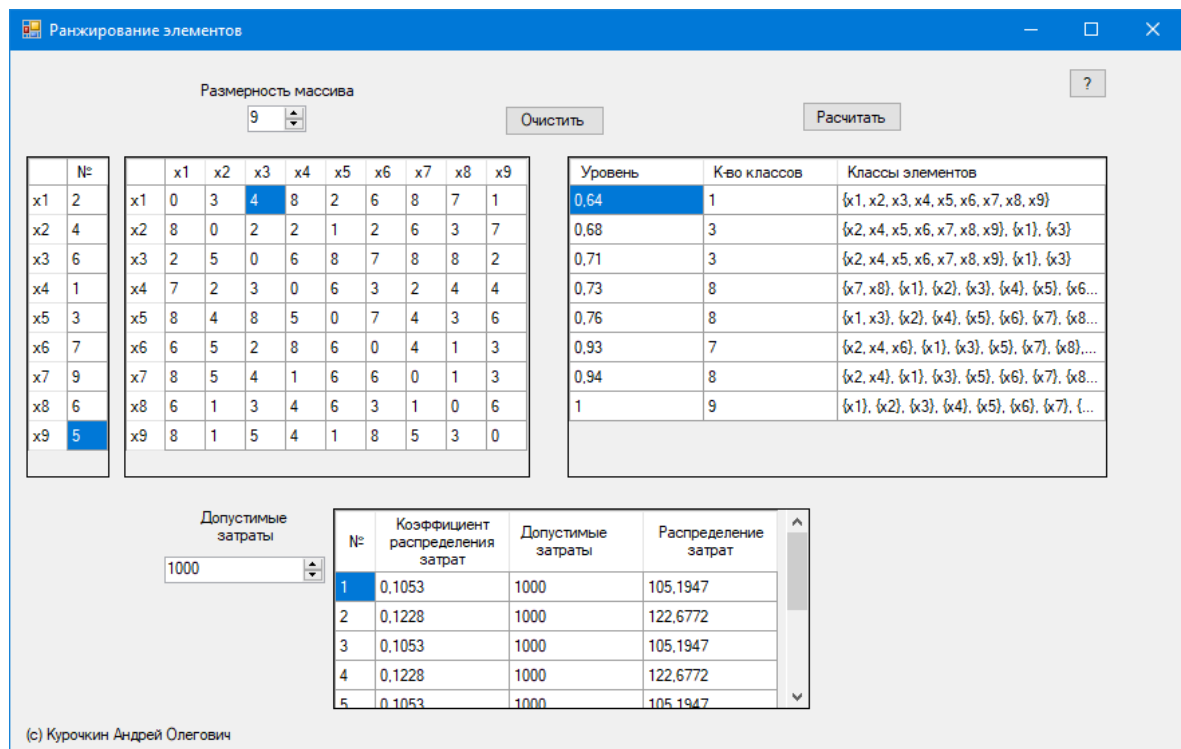


Рисунок 4.3.9 – Результат роботи програми

| Допустимые затраты |  | № | Коэффициент распределения затрат | Допустимые затраты | Распределение затрат |
|--------------------|--|---|----------------------------------|--------------------|----------------------|
| 15364              |  | 1 | 0,1053                           | 15364              | 1617,8292            |
|                    |  | 2 | 0,1228                           | 15364              | 1886,6992            |
|                    |  | 3 | 0,1053                           | 15364              | 1617,8292            |
|                    |  | 4 | 0,1228                           | 15364              | 1886,6992            |
|                    |  | 5 | 0,1053                           | 15364              | 1617,8292            |

Рисунок 4.3.10 – Приклад динамічної зміни фінансових витрат

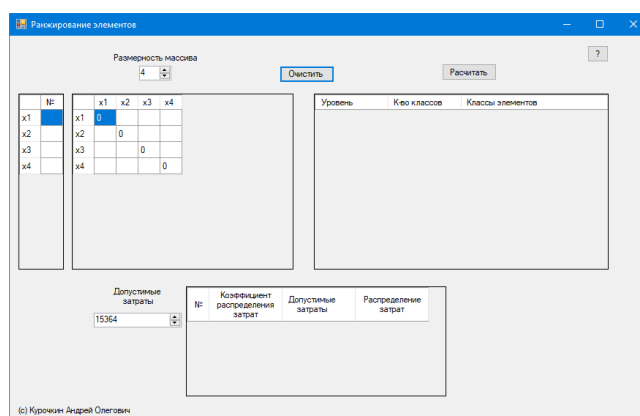


Рисунок 4.3.11 – Приклад натиснення кнопки «Очистить»

#### 4.5. Графова модель життєвого циклу небезпечного впливу бази даних

Після визначення допустимих витрат розробленою інформаційною технологією за допомогою методу ранжування постає формулювання задачі оптимізації НМП прийняття рішень. Для того, щоб мати можливість обирати організаційно-технічні заходи для забезпечення надійності бази даних, потрібно створити загальну модель, що характеризує розвиток негативного впливу та їх усунення, коли система знаходиться в умовах конфліктних взаємодій. На базі цієї моделі був розроблений граф подій НМП прийняття рішень, який зображений на рис. 4.4.1.

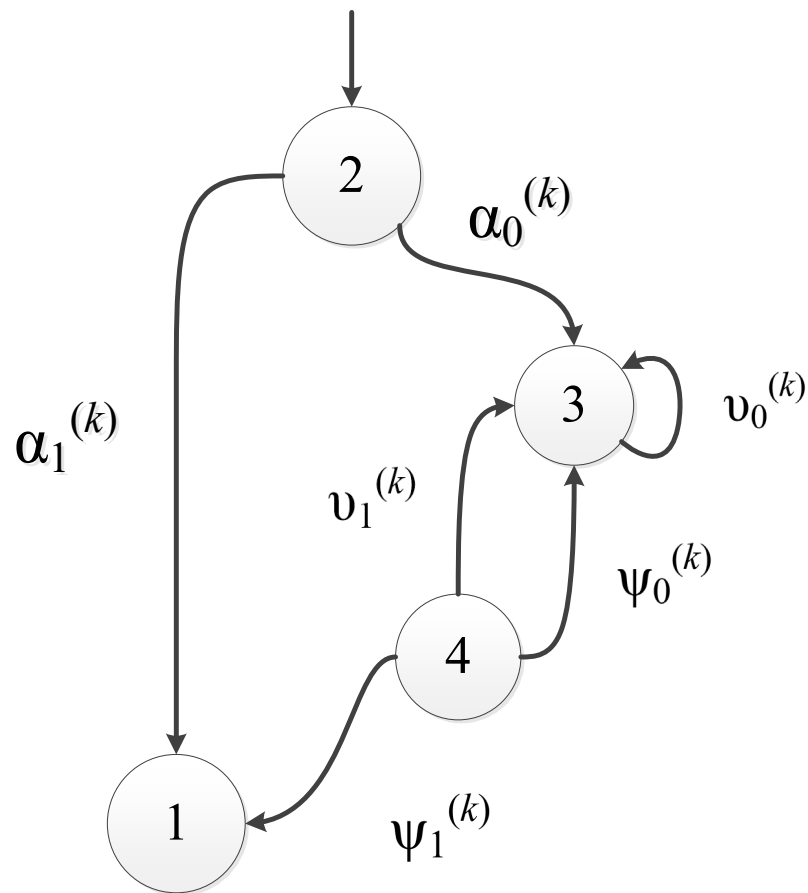


Рис. 4.4.1. - Граф подій життєвого циклу негативного впливу на базу даних

Кожній вершині графу відповідає свій унікальний стан взаємодії між базою даних та негативним впливом, а саме:

- 1) Стан, який відповідає працездатній базі даних
- 2) Стан, який відповідає початку впливу джерела негативного впливу (ДНВ) на інформаційну систему
- 3) Стан, який відповідає порушенню працездатності БД і невиявленого негативного впливу
- 4) Стан, який відповідає порушенню працездатності БД і виявленого негативного впливу

А стрілки від вершини до вершини являють собою ймовірності переходів:

$\alpha_1^{(k)}$  – ймовірність усунення небезпечного впливу системами автоматизованого забезпечення надійності при застосуванні  $k$ -ї стратегії,



$\alpha_0^{(k)}$  – пошкодження працездатності бази даних шляхом ДНВ;

$\nu_1^{(k)}$  – ймовірність не проявленої небезпечної кіберзагрози;

$\nu_0^{(k)}$  – ймовірність не проявленої небезпечної впливу експертною групою, яка відповідає за надійність інформаційної системи;

$\psi_1^{(k)}$  – ймовірність усунення небезпечної впливу без помилок експертною групою.

$K_p$  – безліч рішень (стратегій) виявлення та усунення порушення системним адміністратором або експертною групою, яка відповідає за надійність інформаційної системи;

$\psi_0^{(k)}$  – ймовірність усунення небезпечної впливу експертною групою з помилкою обраної стратегії.

Після цього, була розроблена загальна модель процесів переходів, яка фокусується на максимізації ймовірності переходів у необхідний стан, яка зображена на рис.

#### **4.6. Розробка математичної моделі вибору рішення виявлення та усунення небезпечної впливу.**

Допустимо, що існує інформаційна система, що складається із баз даних забезпечення працездатності деякого підприємства. Системний адміністратор виявляє, що працездатність системи була порушена джерелом негативного впливу. Надходить заявка на усунення негативного впливу. У результаті виконання заявки потрібно отримання певного рішення усунення небезпечної впливу, процес досягнення якого описується алгоритмічно. Існує безліч рішень, які можна призначати на етапи технологічного процесу. Необхідно обрати рішення, яке буде

забезпечувати максимум безпомилковості виконання усунення негативного впливу при обмеженні на середній час виявлення та усунення джерела негативного впливу.

Задача оптимізації може проводитися на базі двох можливих типів моделей функціонування: на графі робіт (функціональної мережі) на графі подій (з використанням напівмарковських процесів (ПМП)).

Первинною і більш природною формою подання алгоритму функціонування (АФ) є граф робіт. Для отримання графа подій кожній вершині графа робіт необхідно зіставити події, що відповідають початку або закінчення виконання відповідної роботи або операції. Результатів виконання робіт може бути декілька, у найпростішому випадку два: працездатний і непрацездатний стан. Кожному результату відповідає певна дуга (лінія) на графі. Кожен з наступних станів відповідає різним наслідкам виконання АФ, у найпростішому випадку: правильне і неправильне. Реалізований граф робіт на граф подій зображений на рис.4.5.1 для алгоритмічного процесу, який видається типовою функціональною структурою (ТФС) «робоча - контроль функціонування».

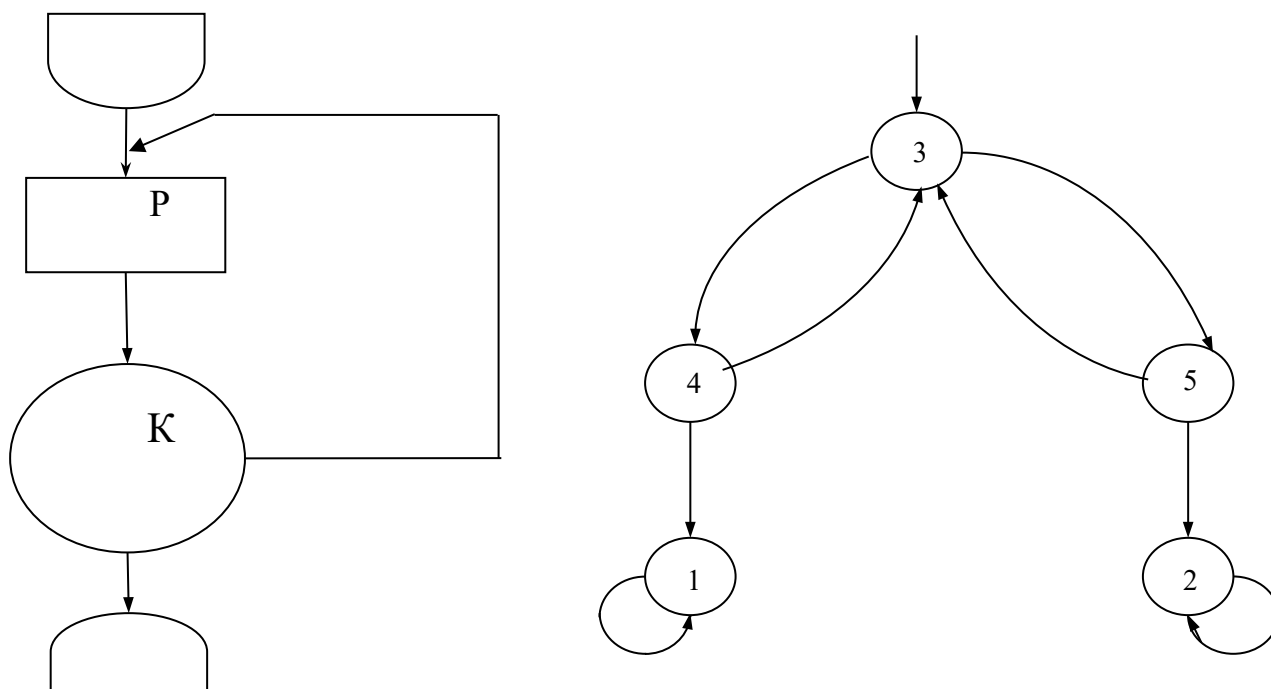


Рис. 4.5.1. ТФС «Робоча - контроль функціонування». Граф робіт (а), граф подій (б).

Позначимо через  $K$  безліч усіх рішень усунення негативного впливу. Кожному варіанту закінчення функціонування процесу діяльності на графі подій ставимо у відповідність своє поглинаючий стан, наприклад, «працездатний стан системи» процесу і «непрацездатний стан системи». Поглинаючі вершини нумеруємо першими  $r$  натуральними числами ( $r$  - кількість поглинаючих вершин). В їх число входять позитивні результати  $r_i$ . Для початкових вершин, які нумеруються числами з числової послідовності після перших  $r$  поглинаючих вершин, задамо вектор початкових ймовірностей, тобто ймовірності знаходження системи в початкових станах у відповідних вершинах графа подій:

$$a = (a_{r+1}, a_{r+2}, \dots, a_n), \text{ при цьому :} \\ \sum_{i=r+1}^N a_i = 1 \quad (1)$$

$N$  - загальна кількість вершин, з яких перші  $r$  - поглинають.

У кожній вершині  $i$  може бути  $K_i$  альтернатив (у кожному варіанті розвитку подій можуть бути використані  $K_i \in K$  рішень). Кожній альтернативі відповідає свій набір переходів, який характеризується ймовірністю переходу до різних вершин із використанням рішення  $k$ ,  $K_i \in K$ . В умовах завдання закріплення вибору рішення за вступниками функціями  $k$ -а альтернатива означає призначення рішення  $k \in K_i \in K$  до рівня процесу інформаційної системи.  $P_{ij}^{(k)}$  – ймовірність міграції до різних вершин (із однієї до іншої вершини) , але варто зазначити, що:

$$\sum_j p_{ij}^{(k)} = 1 \text{ при всіх } i \text{ та всіх } k \in K_i \quad (2)$$

Вершини графа  $l, m, \dots, n$  будемо називати залежними, якщо в них повинні прийматися однакові рішення.

Під оптимізацією системи розуміється вибір в кожній вершині такого рішення (організаційно-технічного заходу), щоб цільової функції доставлявся екстремум (максимум). Цільова функція, у нашому випадку -  $P_r$ - ймовірність поглинання в  $r$ -стані (або в станах  $r$ -го типу).

Результатами оптимізації є:

1. Значення цільової функції  $P_r$ ;

2. Номер рішення в кожній вершині, який доставляє екстремум цільової функції.

При постановці завдання оптимізації у вигляді задачі лінійного програмування цільова функція мають вигляд:

$$P_r = \sum_l \sum_{i=r+1}^N \sum_{k \in K_i} P_{il}^{(k)} * x_i^{(k)} \quad (3)$$

У формулі (3) - змінна, яка характеризує вибір рішення:  $> 0$  в тому випадку, якщо в  $i$ -й вершині для виконання роботи вибрано  $k$ -е рішення,  $i = 0$  в іншому випадку.

Для обліку залежних вершин вводимо булеві змінні і додаємо обмеження, що вимагають, щоб для залежних станів альтернативи збігалися.

Тоді задача оптимізації виявлення та усунення негативного впливу матиме вигляд:

$$P_r = \sum_l \sum_{i=r+1}^N \sum_{k \in K_i} P_{il}^{(k)} * x_i^{(k)} \rightarrow \max \quad (4)$$

$$\sum_{k \in K} x_j^{(k)} - \sum_{i=r+1}^N \sum_{k \in K} P_{ij}^{(k)} * x_i^{(k)} = a_j, j = r + 1, r + 2, \dots, N \quad (5)$$

$$\sum_{k \in K_i} \delta_i^{(k)} = 1 \text{ При всіх } i \quad (6)$$

$$\delta_l^{(k)} = \delta_m^{(k)} = \dots = \delta_i^{(k)} \text{ при всіх } k \in K_i \quad (7)$$

$$x_i^{(k)} - M * \delta_i^{(k)} \leq 0 \text{ при всіх } i \text{ та всіх } k \in K_i \quad (8)$$

$$x_i^{(k)} - m * \delta_i^{(k)} \geq 0 \text{ при всіх } i \text{ та всіх } k \in K_i \quad (9)$$

$$\sum_{j=1}^r \sum_{i=r+1}^N \sum_{k \in K_i} P_{ij}^{(k)} * x_i^{(k)} = 1 \quad (10)$$

$$x_i^{(k)} \geq 0 \text{ при всіх } i \text{ та всіх } k \in K_i \quad (11)$$

Де  $m$  і  $M$  - досить мале і досить велике числа.

Сенс зазначених обмежень наступний:

- 1) обмеження (6) що лише одна і одна змінна  $\delta_i^{(k)}$  була рівна одиниці ;
- 2) обмеження (8) потрібне для того, щоб лише одне значення  $x_k$  не дорівнюватиме нулю (разом із обмеженням (11));

3) обмеження (10) являє собою нормовану умову, яка потребує аби 1 процес виконав поглинання.

Тому, обмеження (4)-(11) мають на увазі те, що лише для одного  $x_i^{(k)}$  та  $\delta_i^{(k)}$  існує значення, яке не дорівнюватиме нулю. Якщо дана умова виконується, то це означає що у  $i$ -й вершині для оптимального усунення небезпечного впливу обирається  $k$ -е рішення. В умовах задачі це означає, що на етапі технологічного процесу, що моделюються  $i$ -ми вершинами графа подій, призначається рішення  $k \in K_i \in K$ . Якщо необхідно, щоб весь технологічний процес використовував одне рішення, то обмеження (7) поширюється на всі вершини графа.

#### 4.7. Приклад розв'язання задачі оптимізації забезпечення надійності бази даних.

Для вирішення складеної оптимізаційної задачі використовуємо плагін MS Excel «Пошук рішення». Для цього на робочому аркуші MS Excel встановлюємо цільову комірку і заносимо в цей осередок формулу розрахунку цільової функції, задаємо обмеження на змінні і виконуємо рішення (рис.4.6.1).

|                          |  | Матриці перехідних ймовірностей |      |      |      |      |                             |  |       |      |       |      |      |
|--------------------------|--|---------------------------------|------|------|------|------|-----------------------------|--|-------|------|-------|------|------|
| Перше рішення $P_{ij} =$ |  | 1                               | 0    | 0    | 0    | 0    | Друге рішення $P_{ij} =$    |  | 1     | 0    | 0     | 0    | 0    |
|                          |  | 0                               | 1    | 0    | 0    | 0    |                             |  | 0     | 1    | 0     | 0    | 0    |
|                          |  | 0                               | 0    | 0    | 0,91 | 0,09 |                             |  | 0     | 0    | 0     | 0,9  | 0,1  |
|                          |  | 0,92                            | 0    | 0,08 | 0    | 0    |                             |  | 0,93  | 0    | 0,07  | 0    | 0    |
|                          |  | 0                               | 0,05 | 0,95 | 0    | 0    |                             |  | 0     | 0,03 | 0,97  | 0    | 0    |
| Третє рішення $P_{ij} =$ |  | 1                               | 0    | 0    | 0    | 0    | Четверте рішення $P_{ij} =$ |  | 1     | 0    | 0     | 0    | 0    |
|                          |  | 0                               | 1    | 0    | 0    | 0    |                             |  | 0     | 1    | 0     | 0    | 0    |
|                          |  | 0                               | 0    | 0    | 0,99 | 0,01 |                             |  | 0     | 0    | 0     | 0,93 | 0,07 |
|                          |  | 0,98                            | 0    | 0,02 | 0    | 0    |                             |  | 0,935 | 0    | 0,065 | 0    | 0    |
|                          |  | 0                               | 0,06 | 0,94 | 0    | 0    |                             |  | 0     | 0,05 | 0,95  | 0    | 0    |

Рисунок 4.6.1 – Внесення вхідних даних

Використовуючи напівмарківську модель разом із графом робіт та графом подій як основу структури, вводимо обмеження як на цільову функцію, так додаткові обмеження (на досить велике і мале число, діапазон чисел  $x$  та  $\delta$  тощо). На рис. 4.6.2

– 4.6.3 зображена надбудова «Пошук рішення», Заповненні формули та обмеження, а також результати обчислень.

|        |      |      |  |  |          |       |      |
|--------|------|------|--|--|----------|-------|------|
|        | 0,99 | 0,01 |  |  | рішення  | 0     | 0    |
| $\sum$ | 0    | 0    |  |  | $R_{ij}$ | 0,935 | 0    |
|        | 0    | 0    |  |  |          |       | 0,05 |

|        |         |         |         |         |         |         |         |
|--------|---------|---------|---------|---------|---------|---------|---------|
|        | $x_1^4$ | $x_1^1$ | $x_2^2$ | $x_2^3$ | $x_3^4$ | $x_3^1$ | $x_3^2$ |
| $\sum$ | 0,0000  | 1,0811  | 0,0000  | 0,0000  | 0,0000  | 0,1069  | 0,0000  |
|        | 0       | 0       | 0       | 0       | 0       | 0       | 0       |

|        |              |              |              |              |              |              |              |
|--------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
|        | $\delta_3^4$ | $\delta_4^1$ | $\delta_4^2$ | $\delta_4^3$ | $\delta_4^4$ | $\delta_5^1$ | $\delta_5^2$ |
| $\sum$ | 0            | 1            | 0            | 0            | 0            | 1            | 0            |
|        | 1            | 1            | 1            | 1            | 1            | 1            | 1            |
|        | Ціле         | Ціле         | Ціле         | Ціле         | Ціле         | Ціле         | Ціле         |

|  |  |          |
|--|--|----------|
| Цільова функція:   | $x_1^{(1)}+P_{11}^{(1)}+x_2^{(2)}+P_{11}^{(2)}+x_3^{(3)}+P_{11}^{(3)}+x_4^{(4)}+P_{11}^{(4)}+x_5^{(5)}+P_{11}^{(5)}+x_6^{(6)}+P_{11}^{(6)}+x_7^{(7)}+P_{11}^{(7)}+x_8^{(8)}+P_{11}^{(8)}+x_9^{(9)}+P_{11}^{(9)}+x_{10}^{(10)}+P_{11}^{(10)}$ | 0,994654 |
| Обмеження на $x_i^{(k)}$ і $P_i^{(k)}$   | Ліва частина   | Значення |
| $x_1^{(1)}+x_1^{(2)}+x_1^{(3)}+x_1^{(4)}+x_1^{(5)}+x_1^{(6)}+x_1^{(7)}+x_1^{(8)}+x_1^{(9)}+x_1^{(10)}=1$ | 1,0000   |          |
| $x_2^{(2)}+x_2^{(3)}+x_2^{(4)}+x_2^{(5)}+x_2^{(6)}+x_2^{(7)}+x_2^{(8)}+x_2^{(9)}+x_2^{(10)}=0$           | 0,0000   |          |
| $x_3^{(4)}+x_3^{(1)}+x_3^{(2)}+x_3^{(3)}+x_3^{(5)}+x_3^{(6)}+x_3^{(7)}+x_3^{(8)}+x_3^{(9)}+x_3^{(10)}=0$ | 0,0000   |          |
| $x_4^{(4)}+x_4^{(1)}+x_4^{(2)}+x_4^{(3)}+x_4^{(5)}+x_4^{(6)}+x_4^{(7)}+x_4^{(8)}+x_4^{(9)}+x_4^{(10)}=1$ | 1,0000   |          |

|   |  |   |
|---|--|---|
| Обмеження на цілісність змінних                                 |  |   |
| $\delta_3^{(1)}+\delta_3^{(2)}+\delta_3^{(3)}+\delta_3^{(4)}=1$ |  | 1 |
| $\delta_4^{(1)}+\delta_4^{(2)}+\delta_4^{(3)}+\delta_4^{(4)}=1$ |  | 1 |
| $\delta_5^{(1)}+\delta_5^{(2)}+\delta_5^{(3)}+\delta_5^{(4)}=1$ |  | 1 |

Параметри пошуку рішення

Оптимізувати цільову функцію:

До:  Максимум  Мінімум  Значення:

Змінити значення параметрів:

Відповідності обмеженням:

- 58526 >= 58527
- 58530 >= 58527
- 58530 >= 58531
- 58535 >= 58555
- 58530 >= 58527
- 58530 >= 58527
- 58530 >= 58527
- 58549 >= 58549
- 58557 >= 58557
- 58536 >= 58527
- 58552 >= 58552
- 58552 >= 58555

Сделайте переменные без ограничений неотрицательными

Выберите метод решения: Поиск решения нелинейных задач методом ОПГ

Метод решения: Для гладких нелинейных задач используйте поиск решения нелинейных задач методом ОПГ, для линейных задач - поиск решения линейных задач симплекс-методом, а для негладких задач - эволюционный поиск решения.

Рисунок 4.6.2 – Надбудова «Пошук рішення»

|                                 |      |      |      |   |      |      |   |   |   |   |                        |      |   |      |   |     |     |   |   |   |
|---------------------------------|------|------|------|---|------|------|---|---|---|---|------------------------|------|---|------|---|-----|-----|---|---|---|
| Матриці перехідних ймовірностей |      |      |      |   |      |      |   |   |   |   |                        |      |   |      |   |     |     |   |   |   |
| Перше рішення $R_{ij}$          | 1    | 0    | 0    | 0 | 0    | 0    | 0 | 0 | 0 | 0 | Друге рішення $R_{ij}$ | 1    | 0 | 0    | 0 | 0   | 0   | 0 | 0 | 0 |
|                                 | 0    | 1    | 0    | 0 | 0    | 0    | 0 | 0 | 0 | 0 |                        | 0    | 1 | 0    | 0 | 0   | 0   |   |   |   |
|                                 | 0    | 0    | 0    | 0 | 0,91 | 0,09 | 0 | 0 | 0 | 0 |                        | 0    | 0 | 0    | 0 | 0,9 | 0,1 |   |   |   |
|                                 | 0,92 | 0    | 0,08 | 0 | 0    | 0    | 0 | 0 | 0 | 0 |                        | 0,93 | 0 | 0,07 | 0 | 0   | 0   |   |   |   |
| 0                               | 0,05 | 0,95 | 0    | 0 | 0    | 0    | 0 | 0 | 0 | 0 | 0,03                   | 0,97 | 0 | 0    | 0 |     |     |   |   |   |

|                        |      |      |      |   |      |      |   |   |   |   |                           |       |   |       |   |      |      |   |   |   |
|------------------------|------|------|------|---|------|------|---|---|---|---|---------------------------|-------|---|-------|---|------|------|---|---|---|
| Третє рішення $R_{ij}$ | 1    | 0    | 0    | 0 | 0    | 0    | 0 | 0 | 0 | 0 | Четверте рішення $R_{ij}$ | 1     | 0 | 0     | 0 | 0    | 0    | 0 | 0 | 0 |
|                        | 0    | 1    | 0    | 0 | 0    | 0    | 0 | 0 | 0 | 0 |                           | 0     | 1 | 0     | 0 | 0    | 0    |   |   |   |
|                        | 0    | 0    | 0    | 0 | 0,89 | 0,01 | 0 | 0 | 0 | 0 |                           | 0     | 0 | 0     | 0 | 0,93 | 0,07 |   |   |   |
|                        | 0,98 | 0    | 0,02 | 0 | 0    | 0    | 0 | 0 | 0 | 0 |                           | 0,935 | 0 | 0,065 | 0 | 0    | 0    |   |   |   |
| 0                      | 0,06 | 0,94 | 0    | 0 | 0    | 0    | 0 | 0 | 0 | 0 | 0,05                      | 0,95  | 0 | 0     | 0 |      |      |   |   |   |

|                |         |         |         |         |         |         |         |         |         |         |         |         |
|----------------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| Змінні         |         |         |         |         |         |         |         |         |         |         |         |         |
| Имя переменной | $x_1^1$ | $x_2^2$ | $x_2^3$ | $x_3^4$ | $x_4^1$ | $x_4^2$ | $x_4^3$ | $x_4^4$ | $x_5^1$ | $x_5^2$ | $x_5^3$ | $x_5^4$ |
| Значения       | 1,1881  | 0,0000  | 0,0000  | 0,0000  | 1,0811  | 0,0000  | 0,0000  | 0,0000  | 0,1069  | 0,0000  | 0,0000  | 0,0000  |
| Нижня граница  | 0       | 0       | 0       | 0       | 0       | 0       | 0       | 0       | 0       | 0       | 0       | 0       |

|                    |              |              |              |              |              |              |              |              |              |              |              |              |
|--------------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| Цілісність змінних |              |              |              |              |              |              |              |              |              |              |              |              |
| Имя переменной     | $\delta_3^1$ | $\delta_3^2$ | $\delta_3^3$ | $\delta_3^4$ | $\delta_4^1$ | $\delta_4^2$ | $\delta_4^3$ | $\delta_4^4$ | $\delta_5^1$ | $\delta_5^2$ | $\delta_5^3$ | $\delta_5^4$ |
| Значения           | 1            | 0            | 0            | 0            | 1            | 0            | 0            | 0            | 1            | 0            | 0            | 0            |
| Верхня граница     | 1            | 1            | 1            | 1            | 1            | 1            | 1            | 1            | 1            | 1            | 1            | 1            |
| Цілісність         | Ціле         | Ціле         | Ціле         | Ціле         | Ціле         | Ціле         | Ціле         | Ціле         | Ціле         | Ціле         | Ціле         | Ціле         |

|  |  |          |               |
|--|--|----------|---------------|
| Цільова функція:   | $x_1^{(1)}+P_{11}^{(1)}+x_2^{(2)}+P_{11}^{(2)}+x_3^{(3)}+P_{11}^{(3)}+x_4^{(4)}+P_{11}^{(4)}+x_5^{(5)}+P_{11}^{(5)}+x_6^{(6)}+P_{11}^{(6)}+x_7^{(7)}+P_{11}^{(7)}+x_8^{(8)}+P_{11}^{(8)}+x_9^{(9)}+P_{11}^{(9)}+x_{10}^{(10)}+P_{11}^{(10)}$ | 0,994654 | Максимум      |
| Обмеження на $x_i^{(k)}$ і $P_i^{(k)}$   | Ліва частина   | Значення | Права частина |
| $x_1^{(1)}+x_1^{(2)}+x_1^{(3)}+x_1^{(4)}+x_1^{(5)}+x_1^{(6)}+x_1^{(7)}+x_1^{(8)}+x_1^{(9)}+x_1^{(10)}=1$ | 1,0000   | =        | 1             |
| $x_2^{(2)}+x_2^{(3)}+x_2^{(4)}+x_2^{(5)}+x_2^{(6)}+x_2^{(7)}+x_2^{(8)}+x_2^{(9)}+x_2^{(10)}=0$           | 0,0000   | =        | 0             |
| $x_3^{(4)}+x_3^{(1)}+x_3^{(2)}+x_3^{(3)}+x_3^{(5)}+x_3^{(6)}+x_3^{(7)}+x_3^{(8)}+x_3^{(9)}+x_3^{(10)}=0$ | 0,0000   | =        | 0             |
| $x_4^{(4)}+x_4^{(1)}+x_4^{(2)}+x_4^{(3)}+x_4^{(5)}+x_4^{(6)}+x_4^{(7)}+x_4^{(8)}+x_4^{(9)}+x_4^{(10)}=1$ | 1,0000   | =        | 1             |

|                                       |   |  |   |   |   |  |  |  |  |  |
|---------------------------------------|---|--|---|---|---|--|--|--|--|--|
| Обмеження на цілісність змінних       |   |  |   |   |   |  |  |  |  |  |
| $\sum_{k \in K_i} \delta_i^{(k)} = 1$ | $\delta_3^{(1)}+\delta_3^{(2)}+\delta_3^{(3)}+\delta_3^{(4)}=1$ |  | 1 | = | 1 |  |  |  |  |  |
|                                       | $\delta_4^{(1)}+\delta_4^{(2)}+\delta_4^{(3)}+\delta_4^{(4)}=1$ |  | 1 | = | 1 |  |  |  |  |  |
|                                       | $\delta_5^{(1)}+\delta_5^{(2)}+\delta_5^{(3)}+\delta_5^{(4)}=1$ |  | 1 | = | 1 |  |  |  |  |  |

|   |   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|---|
| Додаткові обмеження                     |   |   |   |   |   |   |   |   |   |   |
| $x_j^{(k)} - M * \delta_j^{(k)} \leq 0$ | $x_4^{(1)} - M * \delta_4^{(1)} \leq 0$ | $x_4^{(2)} - M * \delta_4^{(2)} \leq 0$ | $x_4^{(3)} - M * \delta_4^{(3)} \leq 0$ | $x_4^{(4)} - M * \delta_4^{(4)} \leq 0$ | $x_5^{(1)} - M * \delta_5^{(1)} \leq 0$ | $x_5^{(2)} - M * \delta_5^{(2)} \leq 0$ | $x_5^{(3)} - M * \delta_5^{(3)} \leq 0$ | $x_5^{(4)} - M * \delta_5^{(4)} \leq 0$ | $x_5^{(5)} - M * \delta_5^{(5)} \leq 0$ | $x_5^{(6)} - M * \delta_5^{(6)} \leq 0$ |
| $x_j^{(k)} - m * \delta_j^{(k)} \geq 0$ | $x_4^{(1)} - m * \delta_4^{(1)} \geq 0$ | $x_4^{(2)} - m * \delta_4^{(2)} \geq 0$ | $x_4^{(3)} - m * \delta_4^{(3)} \geq 0$ | $x_4^{(4)} - m * \delta_4^{(4)} \geq 0$ | $x_5^{(1)} - m * \delta_5^{(1)} \geq 0$ | $x_5^{(2)} - m * \delta_5^{(2)} \geq 0$ | $x_5^{(3)} - m * \delta_5^{(3)} \geq 0$ | $x_5^{(4)} - m * \delta_5^{(4)} \geq 0$ | $x_5^{(5)} - m * \delta_5^{(5)} \geq 0$ | $x_5^{(6)} - m * \delta_5^{(6)} \geq 0$ |

Рисунок 4.6.3 – Результати обчислень

У результаті розв’язання задачі оптимізації, ми отримали декілька значень змінних  $x$  та цільової функції:  $P1 = 0,995991$ ;  $x3(4) = 1,1454 > 0$ ;  $x4(4) = 1,0652 > 0$ ;  $x5(4) = 0,0802 > 0$ . Отриманий результат дає змогу зробити висновок, що при виборі четвертого рішення ми отримуємо максимум ймовірності безпомилкового виконання алгоритму функціонування.

## ВИСНОВКИ

На сьогодні, кількість кібератак, вірусів та негативного та небезпечного впливу на інформаційні системи збільшується. Незважаючи на кількість проведених наукових досліджень, задача оптимізації вибору організаційно технічних засобів в умовах небезпечних впливів вирішена не до кінця.

Для моделювання процесу життєвого циклу небезпечних впливів на інформаційну систему були використані напівмарківські процеси прийняття рішень.

Побудована напівмарківська модель процесу оцінювання організаційно-технічних заходів по забезпеченню надійності інформаційної системи дає можливість обирати стратегії та рішення щодо виявлення та усунення небезпечного впливу.

Створена модель приведена до задачі лінійного програмування, і була реалізована у вигляді інформаційної технології.

Результати можуть бути використані як основа системи забезпечення надійності інформаційних систем для підприємств, вразливих для негативного або небезпечного зовнішнього впливу.

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Jyothsna V. International Journal of Computer Applications. A review of anomaly based intrusion detection systems / V. Jyothsna, S. Prasad. // 28. – 2011. – №7. – С. 26–35.
2. Baddar S. A.-H., Merlo A., Migliardi M. Anomaly detection in computer networks: a state-of-the-art review. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*. 2014. Vol. 5, no. 4. P. 29—64.
3. Gyanchandani M. Taxonomy of anomaly based intrusion detection system: a review. *International Journal of Scientific and Research Publications*. 2012. Vol. 2, no. 12. P. 1—13.
4. Vinchurkar V. A review of intrusion detection system using neural network and machine learning technique. *International Journal of Engineering Science and Innovative Technology*. 2012. Vol. 1, no. 2. P. 54—63.
5. Tsai C. Intrusion detection by machine learning: a review. *Expert Systems with Applications*. 2013, Vol. 36, no. 10. P. 11994—12000.
6. Omar S. Machine learning techniques for anomaly detection: an overview. *International Journal of Computer Applications*. 2013. Vol. 79, no. 2. P. 33—41.
7. Guan Y. Y-means: a clustering method for intrusion detection. *Canadian Conference on Electrical and Computer Engineering. Toward a Caring and Humane Technology*. 2013. Vol. 2, no. 1. P. 1083—1086.
8. Khan L. A new intrusion detection system using support vector machines and hierarchical clustering. *The VLDB Journal*. 2003. Vol. 2, no. 1. P. 507—521.
9. Kabiri P. Research on intrusion detection and response: a survey. *International Journal of Network Security*. 2005. Vol. 1, no. 2. P. 84—102.



10. El Hassani A., El Kalam A. Integrity-OrBAC: a new model to preserve Critical Infrastructures integrity. *International Journal of Information Security*. 2014. Vol. 14, no. 4.
11. Al-Jarrah O. Network Intrusion Detection System using attack behavior classification. *2014 5th International Conference on Information and Communication Systems (ICICS)*. 2014. P. 1—6.
12. Zhou Y. P. Hybrid Model Based on Artificial Immune System and PCA Neural Networks for Intrusion Detection. *Asia-Pacific Conference on Information Processing*. 2009. Vol. 1, no. 1. P. 21—24.
13. Zhan Z. Characterizing Honeypot-Captured Cyber Attacks: Statistical Framework and Case Study. *IEEE Transactions on Information Forensics and Security*. 2014. Vol. 8, no. 11. P. 21—24.
14. Raiyn J. A survey of Cyber Attack Detection Strategies. *International Journal of Security and Its Applications*. 2014. P. 247—256.
15. Fuzzy multicriterial selection of alternatives : The worst-case method, *International journal of intelligent systems* / A. Rotshtein et al. 2010. Vol. 01. P. 948—957.
16. Ротштейн А.П. Ранжирование элементов системы на основе нечетких отношений: метод наименьшего влияния [Электронный ресурс] / А. П. Ротштейн Режим доступа до ресурсу: <https://www.dependability.ru/jour/article/view/100/273>
17. Why Is C# Among The Most Popular Programming Languages in The World?: [Электронный ресурс] Режим доступа до ресурсу: <https://medium.com/sololearn/why-is-c-among-the-most-popular-programming-languages-in-the-world-ccf26824ffcb>
18. Бахусова Є.В. Методи підтримки прийняття рішень на основі нечіткої математики : [Електронний ресурс] / Бахусова Є.В. Режим доступа до ресурсу: <https://bit.ly/2wFPzbA>

19. Курочкін А.О. Оптимізація процесів забезпечення надійності баз даних розподілених інформаційних систем в умовах негативного зовнішнього впливу : [Електронний ресурс] / А.О. Курочкін. Режим доступу до ресурсу: <https://bit.ly/2Itylmg>
20. Саати Т. Принятие решений. Метод анализа иерархий. Москва: Радио и связь, 1993. 278 с.
21. Назаров, Д. М. Интеллектуальные системы: основы теории нечетких множин: навчальний посібник для академічного бакалавріата / Д. М. Назаров, Л. К. Конишева. : Вид. 3-є . Москва : Видавництво Юрайт, 2019. 186 с.
22. Бешелев С.Д., Гурвич Ф.Г. Математико-статистические методы экспертных оценок : 2-е изд., перераб. и доп. Москва: Статистика, 1980. 263 с.
23. Грабовецький Б. Є. Методи експертних оцінок: теорія, методологія, напрямки використання: монографія. Вінниця : ВНТУ, 2010. 171 с.
24. Гнатієнко Г. М., Снитюк В. Є. Експертні технології прийняття рішень : монографія. Київ : ТОВ «Маклаут», 2008. 444 с.
25. Федосеев В. Н. Системы поддержки принятия управленческих решений. Метод анализа иерархий: учебно-практическое пособие. Иваново : библиотека «Научная жизнь», 2011. 55 с.
26. Евланов Л. Г. Теория и практика принятия решений : Москва : Экономика, 1984, 176 с.
27. Метод парного сравнения: [Електронний ресурс]. Режим доступу до ресурсу: <https://studfiles.net/preview/5348316/page:5/>
28. Уверская Ю. У. Оценка эффективности деятельности отдельных структурных подразделений. *Понятие ранжирования, виды ранжирования и правила метода ранжирования.* [Електронний ресурс]. Режим доступу до ресурсу: <https://bit.ly/2ImGTMX>

29. Падерно, П. И. Системный администратор локальной вычислительной сети. Задачи, требования, модель, отбор [Текст] /П. И. Падерно, Ф. Э. Сатторов // Вестник академии наук Республики Таджикистан, 2009. – Т. 52, № 6. С. 437–442

30. Jensen K. Secunia Vulnerability Review 2014: Key figures and facts from a global IT-Security perspective. slideshare. Режим доступа до ресурсу: <https://bit.ly/2VOMcfc>

31. Kent S. INFOSEC Research Council Hard problem List. nitrd.gov. Режим доступа до ресурсу: <https://bit.ly/3a01kym>

## ДОДАТОК А. ПЛАНУВАННЯ РОБІТ

Продуктом дипломного проекту є моделі оцінювання організаційно-технічних заходів забезпечення надійності інформаційної системи в умовах небезпечних впливів.

Результати деталізації методом SMART розміщені у табл. А.1.

Таблиця А.1 – Деталізація мети методом SMART

|                                  |   |
|----------------------------------|---|
| Specific<br>(конкретна)          | Створити моделі оцінювання організаційно-технічних заходів забезпечення надійності інформаційної системи в умовах небезпечних впливів |
| Measurable<br>(вимірювана)       | Результатом роботи проекту є обчислені моделі оптимізації розподілу фінансових ресурсів   |
| Achievable<br>(досяжна)          | Реалізація моделей здійснюється за допомогою Microsoft Office Excel та Microsoft Office Visio   |
| Relevant<br>(реалістична)        | У наявності необхідні просунуті навички роботи з програмним забезпеченням   |
| Time-framed<br>(обмежена у часі) | Робота повинна бути виконана у час, обговорений замовником проекту.   |

## **A.1. Планування змісту структури робіт IT-проекту**

Успішне управління проектами спирається на ретельне планування, щоб визначити цілі проекту досить детально для підтримки ефективного управління. WBS забезпечує основу для визначення роботи, яка стосується цілей проекту, а також встановлює структуру управління роботою до її завершення. WBS використовується для досягнення наступного:

- Визначити обсяг роботи проекту з точки зору результатів та подальшого розкладу цих результатів на компоненти. Залежно від використовуваного методу декомпозиції, WBS може також визначати процес життєвого циклу проекту, а також результати, що відповідають проекту, програмі або будь-якому об'єкту вищого порядку. Цей декомпозиційний обсяг проекту врівноважує потребу керівництва у контролі з відповідним рівнем деталізації.

- Надання команді керівників проектів основу, на якій базуватимуться звіти про стан проекту та прогрес.

- Сприяння комунікації між керівником проекту та зацікавленими сторонами протягом усього життєвого циклу проекту. WBS може використовуватися для передачі інформації щодо обсягу проекту, залежностей та ризиків.

- Забезпечити внесок у інші процеси управління проектами та результати.

WBS чітко формулює обсяг роботи проекту і це є критичним вкладом в інші процеси управління проектами та результати, такі як: визначення діяльності, діаграми мережеских графіків проектів, графіки проектів та програм, звіти про ефективність та інструменти організації та контролю проекту. Хоча WBS є ключовим внеском у дані процеси управління проектами та кінцеві результати, WBS сама по собі не може виступати заміною або представляти будь-який з них самостійно. Для цілей цього практичного стандарту для WBS проект можливо визначити як внутрішньо

зосереджений, зосереджений зовні або обидва разом. Внутрішньо орієнтовані проекти можуть призвести до результатів для інших етапів проекту, інших осіб або інших організацій компанії, яка фінансує проект. Зовнішньо орієнтовані проекти зазвичай дають результати та результати для людей або організацій, що не входять до складу компанії, таких як замовники або спонсори проектів. Багато проектів виробляють як внутрішні, так і зовнішні результати. Незалежно від фокусу проекту, WBS слід готувати у всіх випадках. Побудована таблиця WBS зображена на рис. Б.1.

Термін організаційна структура розподілу термінів управління проектом (також званий 3-буквенною анаграмою OBS) стосується конкретно інструменту, який може бути використаний командою управління проектами або керівником групи управління проектами в ієрархічному порядку для цілей проведення та створення ретельне та чітко окреслене зображення проектної організації для цілей створення домовленості з метою встановлення взаємозв'язку між різними робочими пакетами, пов'язаними з проектом, а також між цими робочими пакетами та заздалегідь визначеними організаційними підрозділами проекту. Побудована таблиця OBS зображена на рис. А.2.

## **А.2 Побудова календарного графіку виконання ІТ-проекту**

Важливим аспектом планування проекту є додавання відпусток як неробочих днів та періодів вільного часу для ресурсів, залучених у проект. Відстеження неробочого часу є критично важливим для того, щоб переконатися, що проект дотримується свого терміну. Календарний графік визначає стандартний робочий та неробочий час, наприклад, вихідні та святкові дні для проекту. Календарі використовуються для планування проекту та визначення робочого часу, щоб визначити, коли ресурси організації, для роботи. За допомогою програми Microsoft Project Professional побудовано діаграму Ганта (рис. А.3).

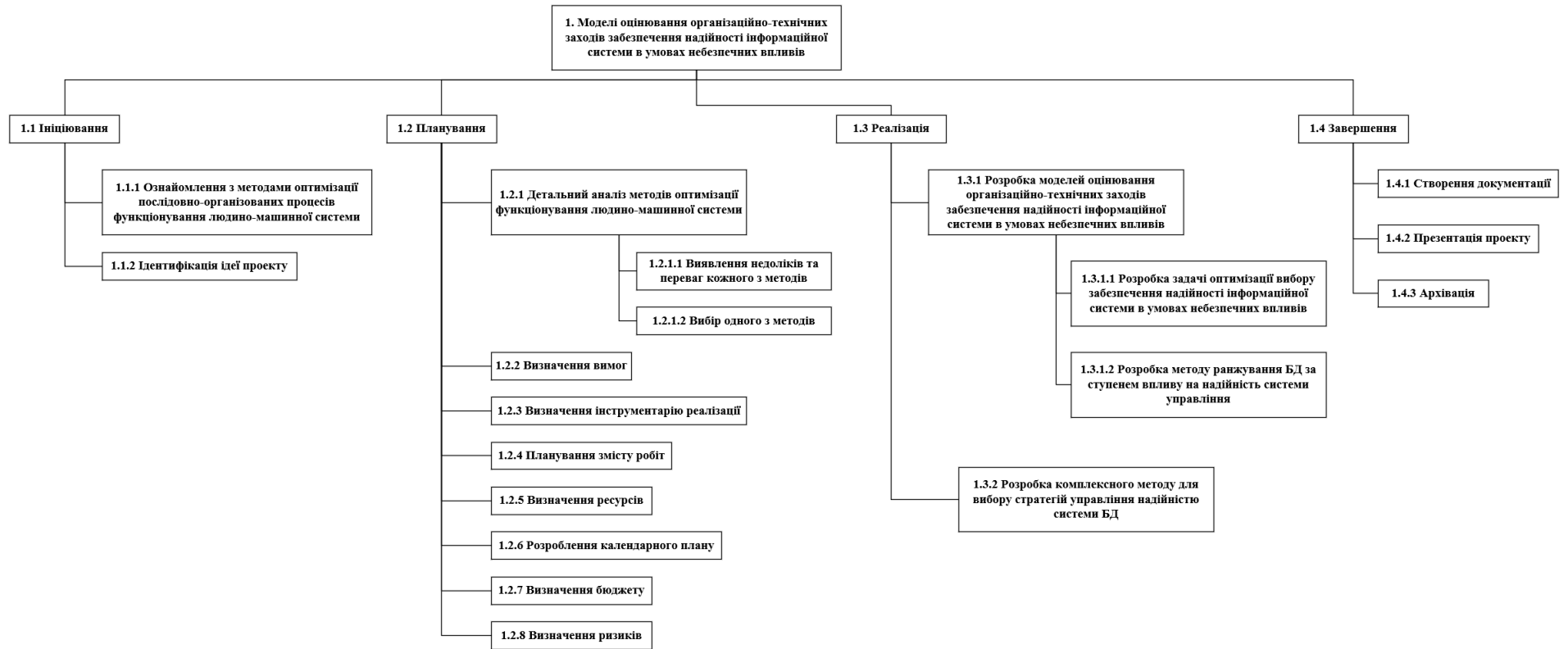


Рисунок А.1 – таблиця WBS (work breakdown structure)

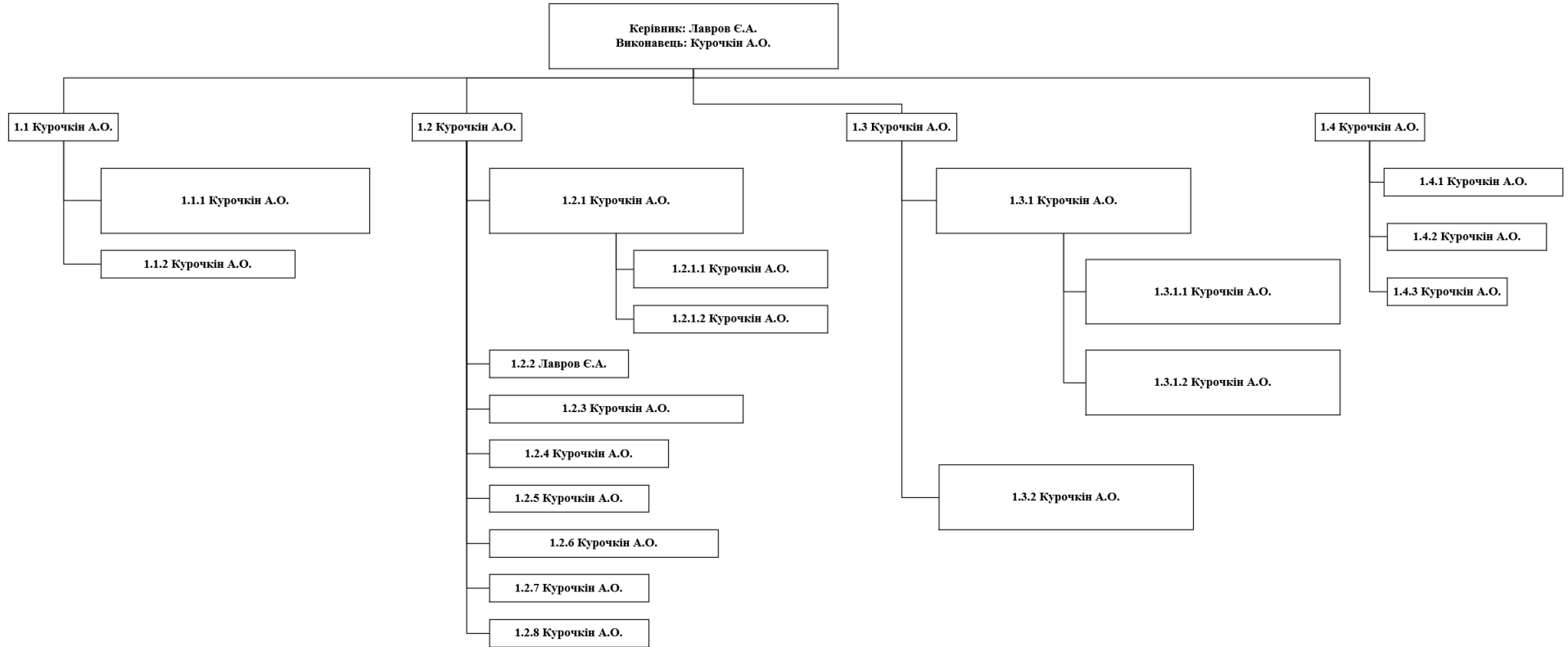


Рисунок А.2 – таблиця OBS (Organization Breakdown Structure)



|  |          |             |             |                            |
|--|----------|-------------|-------------|----------------------------|
| 1. Моделі оцінювання організаційно-технічних заходів забезпечення надійності інформаційної системи в умовах небезпечних впливів              | 175 днів | Пн 27.04.20 | Пт 25.12.20 | Курочкін А.О.; Лавров Є.А. |
| 1.1 Ініціювання  | 7 днів   | Пн 27.04.20 | Вт 05.05.20 | Курочкін А.О.              |
| 1.1.1 Ознайомлення з методами оптимізації послідовно-організованих процесів функціонування людино-машинної системи                           | 1 день   | Пн 27.04.20 | Пн 27.04.20 | Курочкін А.О.              |
| 1.1.2 Ідентифікація ідеї проекту   | 6 днів   | Вт 28.04.20 | Вт 05.05.20 | 3 Курочкін А.О.            |
| 1.2 Планування   | 10 днів  | Ср 06.05.20 | Вт 19.05.20 | 2 Курочкін А.О.            |
| 1.2.1 Детальний аналіз методів оптимізації процесів функціонування людино-машинної системи   | 3 днів   | Ср 06.05.20 | Пт 08.05.20 | Курочкін А.О.              |
| 1.2.1.1 Виявлення недоліків та переваг кожного з методів   | 2 днів   | Ср 06.05.20 | Чт 07.05.20 | Курочкін А.О.              |
| 1.2.1.2 Вибір одного з методів   | 1 день   | Пт 08.05.20 | Пт 08.05.20 | 7 Курочкін А.О.            |
| 1.2.2 Визначення вимог   | 1 день   | Пн 11.05.20 | Пн 11.05.20 | 6 Лавров Є.А.              |
| 1.2.3 Визначення інструментарію реалізації   | 1 день   | Вт 12.05.20 | Вт 12.05.20 | 9 Курочкін А.О.            |
| 1.2.4 Планування змісту робіт  | 1 день   | Ср 13.05.20 | Ср 13.05.20 | 10 Курочкін А.О.           |
| 1.2.5 Визначення ресурсів  | 1 день   | Чт 14.05.20 | Чт 14.05.20 | 11 Курочкін А.О.           |
| 1.2.6 Розроблення календарного плану   | 1 день   | Пт 15.05.20 | Пт 15.05.20 | 12 Курочкін А.О.           |
| 1.2.7 Визначення бюджету   | 1 день   | Пн 18.05.20 | Пн 18.05.20 | 13 Курочкін А.О.           |
| 1.2.8 Визначення ризиків   | 1 день   | Вт 19.05.20 | Вт 19.05.20 | 14 Курочкін А.О.           |
| 1.3 Реалізація   | 140 днів | Ср 20.05.20 | Вт 01.12.20 | 5 Курочкін А.О.            |
| 1.3.1 Розробка моделей оцінювання організаційно-технічних заходів забезпечення надійності інформаційної системи в умовах небезпечних впливів | 80 днів  | Ср 20.05.20 | Вт 08.09.20 | Курочкін А.О.              |
| 1.3.1.1 Розробка задачі оптимізації вибору забезпечення надійності інформаційної системи в умовах небезпечних впливів                        | 40 днів  | Ср 20.05.20 | Вт 14.07.20 | Курочкін А.О.              |
| 1.3.1.2 Розробка методу ранжування БД за ступенем впливу на надійність системи управління  | 40 днів  | Ср 15.07.20 | Вт 08.09.20 | 18 Курочкін А.О.           |
| 1.3.2 Розробка комплексного методу для вибору стратегій управління надійністю системи БД   | 60 днів  | Ср 09.09.20 | Вт 01.12.20 | 17 Курочкін А.О.           |
| 02.12.20   | 16 днів  | Ср 02.12.20 | Ср 23.12.20 | 16 Курочкін А.О.           |
| 1.4.1 Створення документації   | 2 днів   | Ср 02.12.20 | Чт 03.12.20 | Курочкін А.О.              |
| 1.4.2 Презентація проекту  | 2 днів   | Пн 21.12.20 | Вт 22.12.20 | Курочкін А.О.              |
| 1.4.3 Архівація  | 1 день   | Ср 23.12.20 | Ср 23.12.20 | 23 Курочкін А.О.           |

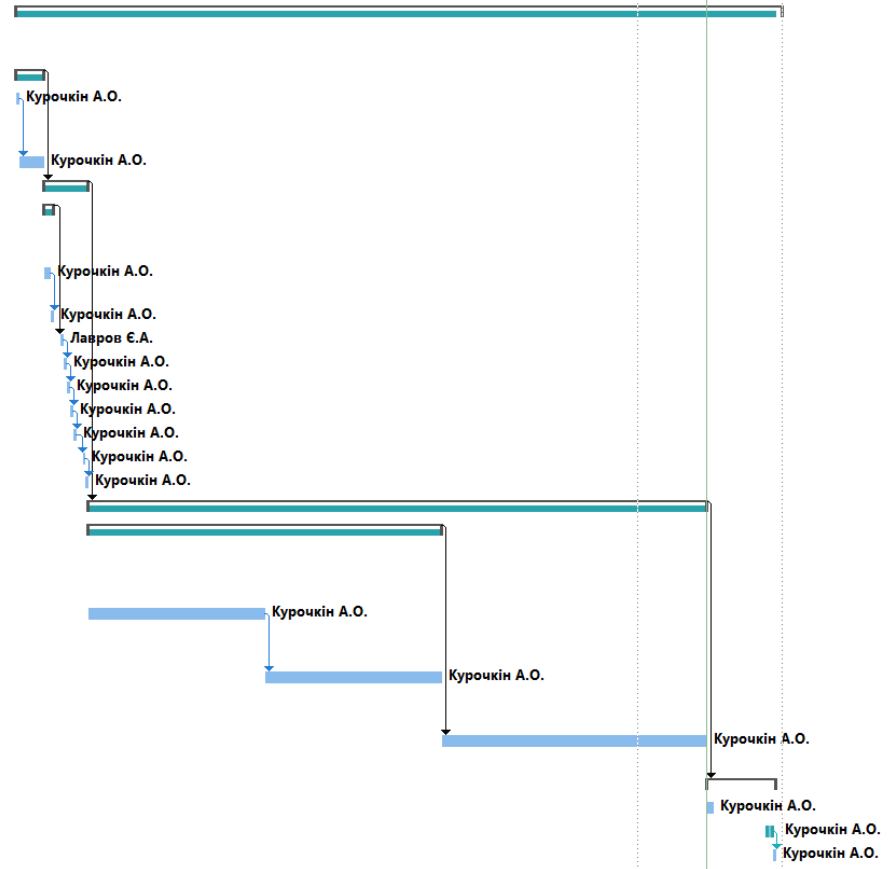


Рисунок А.3 – Діаграма Ганта

### А.3 Планування ризиків проекту

Наступним етапом, після визначення структурних та організаційні структур, розпочинається процес виявлення та аналізування потенційних ризиків, які можуть створити негативний вплив на ключові роботи дипломного проекту. Далі наведений список визначених ризиків:

R1 – зміна вимог при реалізації проекту;

R2 – невчасне виконання робіт проекту;

R3 – виявлення критичних помилок після реалізації програмного забезпечення;

R4 – розбіжність узгодженої та реалізованої версії програмного забезпечення;

R5 – хвороба розробника.

У таблиці А.2 показана ймовірність виникнення кожного ризика.

Таблиці А.2 - Ймовірність виникнення ризиків

| Ймовірність виникнення | R1 | R2 | R3 | R4 | R5 |
|------------------------|----|----|----|----|----|
| Дуже низька            |    |    |    |    |    |
| Низька                 |    |    |    |    |    |
| Середня                |    |    |    |    |    |
| Висока                 |    |    |    |    |    |
| Дуже висока            |    |    |    |    |    |

У таблиці А.3 наведено втрати, які виникнуть при появі кожного з зазначених ризиків.

Таблиця А.3 – Втрати при виникненні ризиків

|                        |    |    |    |    |    |
|------------------------|----|----|----|----|----|
| Ймовірність виникнення | R1 | R2 | R3 | R4 | R5 |
| Дуже низька            |    |    |    |    |    |
| Низька                 |    |    |    |    |    |
| Середня                |    |    |    |    |    |
| Висока                 |    |    |    |    |    |
| Дуже висока            |    |    |    |    |    |

Виходячи з цих двох, була побудована матриця впливу (ймовірностей і наслідків) ризиків. Важливість ризиків зображена у вигляді збільшення темного кольору: чим темніший колір – тим критичніше.

Таблиця А.4 – Матриця ймовірність-втрати

|             |        |    |    |
|-------------|--------|----|----|
| Ймовірність |        |    |    |
|             |        |    | R1 |
|             | R5     | R2 | R5 |
|             |        | R4 | R3 |
|             |        |    |    |
|             | Втрати |    |    |

Виходячи з цього, було визначено два критичні ризика, такі як:

R1 – зміна вимог при реалізації проекту;

R5 – хвороба розробника;

Щоб уникнути ризик зміни вимог, потрібно остаточно визначити вимоги інформаційної технології. Ризик R5 може привести до невчасного виконання робіт.

## ДОДАТОК Б. ЛІСТИНГ ПРОГРАМИ

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace System_Ranging
{
    public partial class Form1 : Form
    {
        int start = 0;

        public Form1()
        {
            InitializeComponent();
            this.input_values.RowHeadersDefaultCellStyle.Padding =
                new Padding(this.input_values.RowHeadersWidth);
            input_values.RowPostPaint +=
                new DataGridViewRowPostPaintEventHandler(dataGridView1_RowPostPaint);

            this.min_i.RowHeadersDefaultCellStyle.Padding =
                new Padding(this.min_i.RowHeadersWidth);
            min_i.RowPostPaint +=
                new DataGridViewRowPostPaintEventHandler(dataGridView1_RowPostPaint);
            value_size.Value = 5;
            value_size.Value = 4;
        }

        static double[] resource(double[,] mas)
        {
            double[] p = new double[mas.GetLength(0)];

            int div_max = 0;
            for (int i = 0; i < mas.GetLength(0); i++)
            {
                p[i] = 0;
                for (int j = 0; j < mas.GetLength(1); j++)
                {
                    p[i] += mas[i, j];
                }
                p[i] = Math.Round(p[i], 0);
                div_max += (int)p[i];
            }

            double[] answer = new double[p.Count()];

            for (int i = 0; i < answer.Count(); i++)
            {
                answer[i] = p[i] / div_max;
                answer[i] = Math.Round(answer[i], 4);
            }

            return answer;
        }
    }
}

```

```

static void RT2(double[,] mas)
{
    double[,] temp = mas;
    for (int i = 0; i < mas.GetLength(0); i++)
        for (int j = 0; j < mas.GetLength(1); j++)
            {
                double multiplier = Math.Pow(10, Convert.ToDouble(2));
                mas[i, j] = Math.Round(mas[i, j], 2);
            }
}

static bool array_check(int[,] mas)
{
    int check = 0;
    for (int i = 0; i < mas.GetLength(0); i++)
        {
            for (int j = 0; j < mas.GetLength(1); j++)
                {
                    if (i == j)
                        {
                            check++;
                        }
                    else if (mas[i, j] >= 1 && mas[i, j] <= 9 && mas[i, j].GetType() == typeof(int))
                        {
                            check++;
                        }
                }
        }
    if (check == Math.Pow(mas.GetLength(0), 2))
        return true;
    else
        return false;
}

static object[,] Class_Evaluation(double[,] mas)
{
    System.Collections.ArrayList unique = new System.Collections.ArrayList();

    for (int i = 0; i < mas.GetLength(0); i++)
        for (int j = 0; j < mas.GetLength(1); j++)
            if (unique.IndexOf(mas[i, j]) == -1)
                unique.Add(mas[i, j]);
    unique.Sort();
    double[] temp = new double[unique.Count];
    for (int i = 0; i < unique.Count; i++)
        {
            temp[i] = Convert.ToDouble(unique[i]);
        }
    string[] elements_classes = new string[temp.GetLength(0)];
    string bonus;
    for (int class_count = 0; class_count < elements_classes.GetLength(0); class_count++)
        {
            bonus = "";
            elements_classes[class_count] = "{";

            for (int i = 0; i < mas.GetLength(0); i++)
                {

                    int help_count = 0;

```

```

    for (int j = 0; j < mas.GetLength(1); j++)
    {
        if (temp[class_count] == mas[i, j])
            help_count++;
    }

    if (help_count > 0)
    {
        elements_classes[class_count] += "x" + (i + 1) + ", ";
    }
    else
        bonus += ", {x" + (i + 1) + "}";
}
elements_classes[class_count] = elements_classes[class_count].Remove(elements_classes[class_count].Length -
2);
elements_classes[class_count] += "}" + bonus;
}

int last = elements_classes.Count() - 1;
elements_classes[last] = "";
for (int i = 0; i < mas.GetLength(0); i++)
{
    if (i != mas.GetLength(0) - 1)
        elements_classes[last] += "{x" + (i + 1) + "}, ";
    else
        elements_classes[last] += "{x" + (i + 1) + "}";
}
char search = Convert.ToChar("{}");
int[] result_count = new int[elements_classes.Count()];
for (int i = 0; i < result_count.Count(); i++)
    for (int j = 0; j < elements_classes[i].Length; j++)
        if (elements_classes[i][j] == search)
            result_count[i]++;

result_count[result_count.Count() - 1] = mas.GetLength(0);
object[,] result = new object[elements_classes.GetLength(0), 3];
for (int i = 0; i < 3; i++)
{
    for (int j = 0; j < elements_classes.GetLength(0); j++)
    {
        switch (i)
        {
            case 0:
                result[j, i] = unique[j];
                break;

            case 1:
                result[j, i] = result_count[j];
                break;

            case 2:
                result[j, i] = elements_classes[j];
                break;
        }
    }
}

return result;
}

```

```

static double[,] Compos(double[,] mas)
{
    double[,] norm_mas = new double[mas.GetLength(0), mas.GetLength(1)];
    for (int i = 0; i < mas.GetLength(0); i++)
    {
        for (int j = 0; j < mas.GetLength(1); j++)
        {
            norm_mas[i, j] = 0;
        }
    }

    double[] min_value = new double[mas.GetLength(1)];
    double max_value = 0;
    for (int i = 0; i < mas.GetLength(0); i++)
    {
        for (int j = 0; j < mas.GetLength(1); j++)
        {
            if (i != j)
            {
                for (int k = 0; k < mas.GetLength(0); k++)
                {
                    if (mas[i, k] < mas[k, j])
                    {
                        min_value[k] = mas[i, k];
                    }
                    else
                    {
                        min_value[k] = mas[k, j];
                    }
                }
                max_value = min_value[0];
                for (int k = 0; k < mas.GetLength(0) - 1; k++)
                {
                    if (max_value < min_value[k + 1])
                    {
                        max_value = min_value[k + 1];
                    }
                }
                norm_mas[i, j] = max_value;
            }
            else
                norm_mas[i, j] = 1;
        }
    }
    return norm_mas;
}

```

```

static double[,] Normalization(double[,] mas)
{
    double[] max_value = new double[mas.GetLength(0)];
    double[,] result = new double[mas.GetLength(0), mas.GetLength(1)];
    for (int i = 0; i < mas.GetLength(0); i++)
    {
        max_value[i] = mas[i, 0];
        for (int j = 1; j < mas.GetLength(1); j++)
        {
            if (max_value[i] < mas[i, j])

```

```

        max_value[i] = mas[i, j];
    }
}

for (int i = 0; i < mas.GetLength(0); i++)
{
    for (int j = 0; j < mas.GetLength(1); j++)
    {
        result[i, j] = mas[i, j] / max_value[i];
    }
}
RT2(result);
return result;
}

static double[,] fuzzy_sim(double[,] mas)
{
    double[,] result = new double[mas.GetLength(0), mas.GetLength(1)];
    double sum = 0;
    for (int i = 0; i < mas.GetLength(0); i++)
    {
        for (int j = 0; j < mas.GetLength(1); j++)
        {
            if (i != j)
            {
                for (int k = 0; k < mas.GetLength(0); k++)
                {
                    sum += Math.Abs(mas[i, k] - mas[j, k]);
                }
                sum = sum / mas.GetLength(0);
                sum = 1 - sum;
                result[i, j] = sum;
                sum = 0;
            }
            else
                result[i, j] = 1;
        }
    }
    RT2(result);

    return result;
}

private void value_size_ValueChanged(object sender, EventArgs e)
{
    start++;
    input_values.Rows.Clear();
    input_values.Columns.Clear();
    min_i.Rows.Clear();
    for (int i = 0; i < value_size.Value; i++)
    {
        DataGridViewTextBoxColumn col = new DataGridViewTextBoxColumn();
        col.Width = 30;
        col.Name = "x" + (i + 1);
        input_values.Columns.Add(col);
        input_values.Rows.Add();
        input_values.Rows[i].HeaderCell.Value = "x" + (i + 1);
        min_i.Rows.Add();
    }
}

```



```

        min_i.Rows[i].HeaderCell.Value = "x" + (i + 1);
    }

    for (int i = 0; i < input_values.Columns.Count; i++)
    {
        input_values.Rows[i].Cells[i].Value = 0;
        input_values.Rows[i].Cells[i].ReadOnly = true;
    }
}

void dataGridView1_RowPostPaint(object sender, DataGridViewRowPostPaintEventArgs e)
{
    object o = input_values.Rows[e.RowIndex].HeaderCell.Value;

    e.Graphics.DrawString(
        o != null ? o.ToString() : "",
        input_values.Font,
        Brushes.Black,
        new PointF((float)e.RowBounds.Left + 2, (float)e.RowBounds.Top + 4));
}

private void button1_Click(object sender, EventArgs e)
{
    Result_Table.Rows.Clear();
    int[,] input_array = new int[input_values.Rows.Count, input_values.Columns.Count];

    for (int x = 0; x < input_array.GetLength(0); x++)
        for (int i = 0; i < input_array.GetLength(1); i++)
            {
                input_array[x, i] = Convert.ToInt32(input_values.Rows[x].Cells[i].Value);
            }

    if (!array_check(input_array))
    {
        MessageBox.Show("Проверьте массив со входными данными!\n(Значения должны быть целочисленными и в диапазон [1;9])");
    }
    else
    {
        int[] min_inf = new int[min_i.Rows.Count];

        for (int i = 0; i < min_inf.Length; i++)
            {
                min_inf[i] = Convert.ToInt32(min_i.Rows[i].Cells[0].Value);
            }

        double[,] fuz_rel = new double[input_array.GetLength(0), input_array.GetLength(1)];

        double sum = 0;
        int helper = 0;
        for (int i = 0; i < input_array.GetLength(0); i++)
            {
                for (int j = 0; j < input_array.GetLength(1); j++)
                    {
                        if (i != j)
                            {
                                if (j == min_inf[i] && helper == 0)
                                    {

```

```

        for (int k = 0; k < input_array.GetLength(0); k++)
        {
            sum += input_array[i, k] / input_array[i, j];
        }
        sum = Math.Pow(sum, -1);
        fuz_rel[i, j] = sum;
        j = 0;
        helper++;
    }
    else
        fuz_rel[i, j] = sum * input_array[i, j];
    }
}
helper--;
}

double[,] norm_ratio = Normalization(fuz_rel);
RT2(norm_ratio);

double[,] ham_dist = fuzzy_sim(norm_ratio);
RT2(ham_dist);
double[,] fuzzy_tranz;

double[,] temp = ham_dist;

int check, iter = 0;
do
{
    check = 0;
    fuzzy_tranz = Compos(temp);

    for (int i = 0; i < fuzzy_tranz.GetLength(0); i++)
        for (int j = 0; j < fuzzy_tranz.GetLength(1); j++)
            if (fuzzy_tranz[i, j] != temp[i, j])
                check++;
    temp = fuzzy_tranz;
    iter++;
    if (iter > 1000)
    {
        MessageBox.Show("Внимание! Количество транзитивных замыканий превысило 1000.\nПроверьте значения входных значений");
        return;
    }
}
while (check != 0);

object[,] result = Class_Evaluation(fuzzy_tranz);

for (int i = 0; i < result.GetLength(0); i++)
{
    Result_Table.Rows.Add();
    for (int j = 0; j < result.GetLength(1); j++)
    {
        Result_Table[j, i].Value = result[i, j];
    }
}

double[] results = resource(fuzzy_tranz);

int C0_Value=Convert.ToInt32(C0.Text);

```



## ДОДАТОК В. КОМП'ЮТЕРНІ ЕКСПЕРИМЕНТИ

Ранжирование элементов

Размерность массива: 4

Очистить Расчитать

| №  | x1 | x2 | x3 | x4 |   |
|----|----|----|----|----|---|
| x1 | 2  | 0  | 1  | 3  | 1 |
| x2 | 3  | 1  | 0  | 1  | 8 |
| x3 | 2  | 1  | 1  | 0  | 8 |
| x4 | 1  | 1  | 3  | 9  | 0 |

| Уровень | К-во классов | Классы элементов       |
|---------|--------------|------------------------|
| 0.5     | 1            | {x1, x2, x3, x4}       |
| 0.89    | 3            | {x1, x4}, {x2}, {x3}   |
| 0.91    | 3            | {x2, x3}, {x1}, {x4}   |
| 1       | 4            | {x1}, {x2}, {x3}, {x4} |

Допустимые затраты: 1000

| № | Кoeffициент распределения затрат | Допустимые затраты | Распределение затрат |
|---|----------------------------------|--------------------|----------------------|
| 1 | 0.25                             | 1000               | 250                  |
| 2 | 0.25                             | 1000               | 250                  |
| 3 | 0.25                             | 1000               | 250                  |
| 4 | 0.25                             | 1000               | 250                  |

(с) Курочкин Андрей Олегович

Рисунок В.1 – Комп'ютерні експерименти (кількість БД – 4)

Ранжирование элементов

Размерность массива: 5

Очистить Расчитать

| №  | x1 | x2 | x3 | x4 | x5 |   |
|----|----|----|----|----|----|---|
| x1 | 2  | 0  | 1  | 5  | 3  | 1 |
| x2 | 1  | 1  | 0  | 9  | 3  | 1 |
| x3 | 5  | 9  | 3  | 0  | 2  | 1 |
| x4 | 2  | 1  | 1  | 7  | 0  | 3 |
| x5 | 1  | 1  | 5  | 9  | 7  | 0 |

| Уровень | К-во классов | Классы элементов             |
|---------|--------------|------------------------------|
| 0.53    | 1            | {x1, x2, x3, x4, x5}         |
| 0.83    | 2            | {x1, x2, x4, x5}, {x3}       |
| 0.84    | 3            | {x1, x2, x4}, {x3}, {x5}     |
| 0.87    | 4            | {x1, x2}, {x3}, {x4}, {x5}   |
| 1       | 5            | {x1}, {x2}, {x3}, {x4}, {x5} |

Допустимые затраты: 1000

| № | Кoeffициент распределения затрат | Допустимые затраты | Распределение затрат |
|---|----------------------------------|--------------------|----------------------|
| 1 | 0.2105                           | 1000               | 210.5                |
| 2 | 0.2105                           | 1000               | 210.5                |
| 3 | 0.1579                           | 1000               | 157.9                |
| 4 | 0.2105                           | 1000               | 210.5                |
| 5 | 0.2105                           | 1000               | 210.5                |

(с) Курочкин Андрей Олегович

Рисунок В.2 - Комп'ютерні експерименти (кількість БД – 5)

Ранжирование элементов

Размерность массива: 6

Очистить Расчитать

|    | № | x1 | x2 | x3 | x4 | x5 | x6 |
|----|---|----|----|----|----|----|----|
| x1 | 5 | 0  | 2  | 3  | 4  | 1  | 1  |
| x2 | 3 | 7  | 0  | 1  | 7  | 3  | 2  |
| x3 | 6 | 1  | 3  | 0  | 8  | 2  | 1  |
| x4 | 5 | 1  | 2  | 1  | 0  | 1  | 6  |
| x5 | 2 | 5  | 1  | 3  | 3  | 0  | 5  |
| x6 | 1 | 1  | 2  | 4  | 3  | 1  | 0  |

| Уровень | К-во классов | Классы элементов                   |
|---------|--------------|------------------------------------|
| 0,63    | 1            | {x1, x2, x3, x4, x5, x6}           |
| 0,64    | 5            | {x4, x5}, {x1}, {x2}, {x3}, {x6}   |
| 0,81    | 3            | {x1, x2, x3, x6}, {x4}, {x5}       |
| 0,83    | 5            | {x1, x6}, {x2}, {x3}, {x4}, {x5}   |
| 0,84    | 5            | {x2, x3}, {x1}, {x4}, {x5}, {x6}   |
| 1       | 6            | {x1}, {x2}, {x3}, {x4}, {x5}, {x6} |

Допустимые затраты: 1000

| № | Коэффициент распределения затрат | Допустимые затраты | Распределение затрат |
|---|----------------------------------|--------------------|----------------------|
| 1 | 0,1786                           | 1000               | 178,6                |
| 2 | 0,1786                           | 1000               | 178,6                |
| 3 | 0,1786                           | 1000               | 178,6                |
| 4 | 0,1429                           | 1000               | 142,9                |
| 5 | 0,1429                           | 1000               | 142,9                |

(с) Курочкин Андрей Олегович

Рисунок В.3 - Комп'ютерні експерименти (кількість БД – 6)

Ранжирование элементов

Размерность массива: 7

Очистить Расчитать

|    | № | x1 | x2 | x3 | x4 | x5 | x6 | x7 |
|----|---|----|----|----|----|----|----|----|
| x1 | 2 | 0  | 1  | 6  | 1  | 2  | 9  | 4  |
| x2 | 1 | 1  | 0  | 6  | 7  | 5  | 5  | 2  |
| x3 | 5 | 6  | 3  | 0  | 2  | 1  | 6  | 3  |
| x4 | 2 | 6  | 1  | 3  | 0  | 4  | 3  | 2  |
| x5 | 3 | 1  | 5  | 1  | 6  | 0  | 4  | 1  |
| x6 | 1 | 1  | 4  | 5  | 2  | 1  | 0  | 5  |
| x7 | 2 | 2  | 1  | 1  | 2  | 7  | 3  | 0  |

| Уровень | К-во классов | Классы элементов                         |
|---------|--------------|--|
| 0,56    | 1            | {x1, x2, x3, x4, x5, x6, x7}             |
| 0,65    | 2            | {x1, x2, x3, x4, x5, x7}, {x6}           |
| 0,66    | 3            | {x1, x2, x3, x4, x7}, {x5}, {x6}         |
| 0,67    | 5            | {x3, x4, x7}, {x1}, {x2}, {x5}, {x6}     |
| 0,68    | 6            | {x1, x2}, {x3}, {x4}, {x5}, {x6}, {x7}   |
| 0,7     | 6            | {x4, x7}, {x1}, {x2}, {x3}, {x5}, {x6}   |
| 1       | 7            | {x1}, {x2}, {x3}, {x4}, {x5}, {x6}, {x7} |

Допустимые затраты: 1000

| № | Коэффициент распределения затрат | Допустимые затраты | Распределение затрат |
|---|----------------------------------|--------------------|----------------------|
| 4 | 0,1471                           | 1000               | 147,1                |
| 5 | 0,1471                           | 1000               | 147,1                |
| 6 | 0,1176                           | 1000               | 117,6                |
| 7 | 0,1471                           | 1000               | 147,1                |

(с) Курочкин Андрей Олегович

Рисунок В.4 - Комп'ютерні експерименти (кількість БД – 7)

Ранжирование элементов

Размерность массива: 8

Очистить Расчитать

| №  | x1 | x2 | x3 | x4 | x5 | x6 | x7 | x8 |   |   |
|----|----|----|----|----|----|----|----|----|---|---|
| x1 | 5  | 0  | 1  | 2  | 4  | 1  | 3  | 2  | 6 |   |
| x2 | 3  | 2  | 0  | 1  | 1  | 4  | 3  | 1  | 8 |   |
| x3 | 8  | 6  | 1  | 0  | 2  | 3  | 4  | 2  | 1 |   |
| x4 | 1  | 4  | 1  | 5  | 3  | 0  | 2  | 1  | 4 | 3 |
| x5 | 2  | 5  | 8  | 1  | 2  | 1  | 0  | 3  | 4 | 7 |
| x6 | 3  | 6  | 3  | 2  | 1  | 8  | 5  | 0  | 2 | 4 |
| x7 | 1  | 7  | 1  | 6  | 7  | 8  | 6  | 5  | 0 | 3 |
| x8 | 6  | 8  | 5  | 6  | 7  | 8  | 7  | 1  | 9 | 0 |

| Уровень | К-во классов | Классы элементов                               |
|---------|--------------|--|
| 0.62    | 1            | {x1, x2, x3, x4, x5, x6, x7, x8}               |
| 0.67    | 2            | {x1, x2, x3, x5, x6, x7, x8}, {x4}             |
| 0.68    | 4            | {x1, x2, x3, x5, x6}, {x4}, {x7}, {x8}         |
| 0.72    | 5            | {x1, x2, x3, x5}, {x4}, {x6}, {x7}, {x8}       |
| 0.77    | 6            | {x2, x3, x5}, {x1}, {x4}, {x6}, {x7}, {x8}     |
| 0.84    | 7            | {x2, x5}, {x1}, {x3}, {x4}, {x6}, {x7}, {x8}   |
| 1       | 8            | {x1}, {x2}, {x3}, {x4}, {x5}, {x6}, {x7}, {x8} |

Допустимые затраты: 1000

| № | Коэффициент распределения затрат | Допустимые затраты | Распределение затрат |
|---|----------------------------------|--------------------|----------------------|
| 1 | 0,1277                           | 1000               | 127,7                |
| 2 | 0,1277                           | 1000               | 127,7                |
| 3 | 0,1277                           | 1000               | 127,7                |
| 4 | 0,1064                           | 1000               | 106,4                |
| 5 | 0,1277                           | 1000               | 127,7                |

(с) Курочкин Андрей Олегович

Рисунок В.5 - Комп'ютерні експерименти (кількість БД – 8)

Ранжирование элементов

Размерность массива: 9

Очистить Расчитать

| №  | x1 | x2 | x3 | x4 | x5 | x6 | x7 | x8 | x9 |   |   |
|----|----|----|----|----|----|----|----|----|----|---|---|
| x1 | 5  | 0  | 2  | 4  | 2  | 1  | 7  | 3  | 8  | 9 |   |
| x2 | 3  | 3  | 0  | 1  | 2  | 6  | 5  | 4  | 5  | 7 |   |
| x3 | 2  | 4  | 1  | 0  | 6  | 1  | 5  | 3  | 4  | 4 |   |
| x4 | 1  | 3  | 4  | 1  | 6  | 9  | 0  | 2  | 4  | 1 | 3 |
| x5 | 1  | 5  | 6  | 1  | 0  | 5  | 6  | 8  | 7  |   |   |
| x6 | 9  | 4  | 1  | 1  | 2  | 3  | 0  | 8  | 9  | 1 |   |
| x7 | 8  | 5  | 4  | 8  | 9  | 5  | 7  | 0  | 1  | 4 |   |
| x8 | 7  | 4  | 6  | 2  | 2  | 1  | 4  | 1  | 0  | 3 |   |
| x9 | 6  | 9  | 4  | 3  | 2  | 1  | 1  | 6  | 8  | 0 |   |

| Уровень | К-во классов | Классы элементов                                     |
|---------|--------------|--|
| 0.68    | 1            | {x1, x2, x3, x4, x5, x6, x7, x8, x9}                 |
| 0.69    | 1            | {x1, x2, x3, x4, x5, x6, x7, x8, x9}                 |
| 0.7     | 7            | {x4, x7, x8}, {x1}, {x2}, {x3}, {x5}, {x6}, {x9}     |
| 0.72    | 8            | {x4, x8}, {x1}, {x2}, {x3}, {x5}, {x6}, {x7}, {x9}   |
| 0.79    | 6            | {x1, x5, x6, x9}, {x2}, {x3}, {x4}, {x7}, {x8}       |
| 1       | 9            | {x1}, {x2}, {x3}, {x4}, {x5}, {x6}, {x7}, {x8}, {x9} |

Допустимые затраты: 1000

| № | Коэффициент распределения затрат | Допустимые затраты | Распределение затрат |
|---|----------------------------------|--------------------|----------------------|
| 1 | 0,1148                           | 1000               | 114,8                |
| 2 | 0,0984                           | 1000               | 98,4                 |
| 3 | 0,0984                           | 1000               | 98,4                 |
| 4 | 0,1148                           | 1000               | 114,8                |
| 5 | 0,1148                           | 1000               | 114,8                |

(с) Курочкин Андрей Олегович

Рисунок В.6 - Комп'ютерні експерименти (кількість БД – 9)

Ранжирование элементов

Размерность массива: 10

Очистить      **Расчитать**

| №   | x1 | x2 | x3 | x4 | x5 | x6 | x7 | x8 | x9 |
|-----|----|----|----|----|----|----|----|----|----|
| x1  | 2  | 0  | 1  | 5  | 8  | 7  | 7  | 9  | 8  |
| x2  | 3  | 2  | 0  | 1  | 2  | 5  | 8  | 6  | 7  |
| x3  | 4  | 1  | 6  | 0  | 1  | 5  | 6  | 5  | 1  |
| x4  | 6  | 6  | 5  | 1  | 0  | 2  | 1  | 1  | 4  |
| x5  | 8  | 5  | 1  | 1  | 2  | 0  | 1  | 3  | 1  |
| x6  | 7  | 1  | 4  | 2  | 7  | 1  | 0  | 1  | 8  |
| x7  | 1  | 1  | 7  | 2  | 3  | 1  | 9  | 0  | 4  |
| x8  | 2  | 6  | 1  | 1  | 6  | 8  | 8  | 4  | 0  |
| x9  | 2  | 8  | 1  | 6  | 5  | 4  | 3  | 2  | 3  |
| x10 | 1  |    |    |    |    |    |    |    |    |

| Уровень | К-во классов | Классы элементов                                    |
|---------|--------------|---|
| 0.59    | 1            | {x1, x2, x3, x4, x5, x6, x7, x8, x9, x10}           |
| 0.68    | 3            | {x2, x3, x4, x5, x6, x7, x8, x9}, {x1}, {x...}      |
| 0.72    | 8            | {x3, x5, x6}, {x1}, {x2}, {x4}, {x7}, {x8}, ...     |
| 0.78    | 7            | {x1, x3, x5, x10}, {x2}, {x4}, {x6}, {x7}, ...      |
| 0.93    | 9            | {x4, x8}, {x1}, {x2}, {x3}, {x5}, {x6}, {x7}, ...   |
| 1       | 10           | {x1}, {x2}, {x3}, {x4}, {x5}, {x6}, {x7}, {x8}, ... |

Допустимые затраты: 1000

| № | Коэффициент распределения затрат | Допустимые затраты | Распределение затрат |
|---|----------------------------------|--------------------|----------------------|
| 1 | 0.0882                           | 1000               | 88,2                 |
| 2 | 0.1029                           | 1000               | 102,9                |
| 3 | 0.1029                           | 1000               | 102,9                |
| 4 | 0.1029                           | 1000               | 102,9                |
| 5 | 0.1029                           | 1000               | 102,9                |

(с) Курочкин Андрей Олегович

Рисунок В.7 - Комп'ютерні експерименти (кількість БД – 10)

Ранжирование элементов

Размерность массива: 11

Очистить      **Расчитать**

| №   | x1 | x2 | x3 | x4 | x5 | x6 | x7 | x8 | x9 | x10 | x11 |
|-----|----|----|----|----|----|----|----|----|----|-----|-----|
| x3  | 6  | 0  | 1  | 1  | 5  | 2  | 3  | 3  | 5  | 4   |     |
| x4  | 1  | 3  | 0  | 6  | 1  | 2  | 2  | 8  | 7  | 8   |     |
| x5  | 2  | 8  | 8  | 0  | 4  | 6  | 1  | 6  | 4  | 5   |     |
| x6  | 3  | 1  | 4  | 5  | 0  | 2  | 5  | 7  | 7  | 3   |     |
| x7  | 5  | 8  | 1  | 5  | 1  | 0  | 7  | 1  | 7  | 4   |     |
| x8  | 7  | 1  | 4  | 1  | 6  | 6  | 0  | 4  | 8  | 4   |     |
| x9  | 6  | 1  | 2  | 2  | 2  | 1  | 6  | 0  | 1  | 2   |     |
| x10 | 2  | 2  | 1  | 7  | 6  | 6  | 3  | 1  | 0  | 8   |     |
| x11 | 2  | 5  | 4  | 2  | 2  | 8  | 1  | 3  | 3  | 0   |     |

| Уровень | К-во классов | Классы элементов                                    |
|---------|--------------|---|
| 0.66    | 1            | {x1, x2, x3, x4, x5, x6, x7, x8, x9, x10, ...}      |
| 0.69    | 2            | {x1, x2, x3, x4, x5, x6, x7, x9, x10, x11, ...}     |
| 0.7     | 3            | {x1, x2, x3, x4, x5, x6, x7, x9, x11}, {x8}, ...    |
| 0.71    | 6            | {x1, x2, x3, x4, x6, x9}, {x5}, {x7}, {x8}, ...     |
| 0.72    | 7            | {x1, x2, x4, x6, x9}, {x3}, {x5}, {x7}, {x8}, ...   |
| 0.73    | 10           | {x1, x6}, {x2}, {x3}, {x4}, {x5}, {x7}, {x8}, ...   |
| 0.74    | 10           | {x2, x4}, {x1}, {x3}, {x5}, {x6}, {x7}, {x8}, ...   |
| 0.76    | 10           | {x5, x7}, {x1}, {x2}, {x3}, {x4}, {x6}, {x8}, ...   |
| 1       | 11           | {x1}, {x2}, {x3}, {x4}, {x5}, {x6}, {x7}, {x8}, ... |

Допустимые затраты: 1000

| № | Коэффициент распределения затрат | Допустимые затраты | Распределение затрат |
|---|----------------------------------|--------------------|----------------------|
| 1 | 0.0909                           | 1000               | 90,9                 |
| 2 | 0.0909                           | 1000               | 90,9                 |
| 3 | 0.0909                           | 1000               | 90,9                 |
| 4 | 0.0909                           | 1000               | 90,9                 |
| 5 | 0.0909                           | 1000               | 90,9                 |

(с) Курочкин Андрей Олегович

Рисунок В.8 - Комп'ютерні експерименти (кількість БД – 11)

Ранжирование элементов

Размерность массива: 12

Очистить      Рассчитать

| №:    | x3   | x4 | x5 | x6 | x7 | x8 | x9 | x10 | x11 | x12 |
|-------|------|----|----|----|----|----|----|-----|-----|-----|
| x4 6  | x1 1 | 7  | 7  | 6  | 6  | 3  | 6  | 5   | 1   |     |
| x5 7  | x2 8 | 1  | 2  | 2  | 3  | 4  | 6  | 5   | 2   |     |
| x6 2  | x3 0 | 6  | 5  | 1  | 9  | 9  | 2  | 3   | 5   |     |
| x7 3  | x4 5 | 0  | 1  | 1  | 3  | 8  | 8  | 7   | 2   |     |
| x8 4  | x5 2 | 6  | 0  | 7  | 1  | 6  | 6  | 1   | 2   |     |
| x9 7  | x6 6 | 3  | 4  | 0  | 1  | 8  | 1  | 6   | 8   |     |
| x10 8 | x7 1 | 7  | 4  | 6  | 0  | 8  | 8  | 1   | 7   |     |
| x11 1 | x8 3 | 1  | 4  | 6  | 7  | 0  | 5  | 4   | 2   |     |
| x12 2 | x9 2 | 6  | 3  | 1  | 1  | 1  | 0  | 1   | 4   |     |

| Уровень | К-во классов | Классы элементов   |
|---------|--------------|--|
| 0.63    | 1            | {x1, x2, x3, x4, x5, x6, x7, x8, x9, x10, ...}                                 |
| 0.67    | 3            | {x1, x2, x3, x5, x6, x7, x8, x10, x11, x12, ...}                               |
| 0.69    | 5            | {x1, x2, x5, x6, x7, x10, x11, x12}, {x3}, {x4}                                |
| 0.7     | 6            | {x1, x2, x5, x6, x7, x11, x12}, {x3}, {x4}                                     |
| 0.71    | 8            | {x1, x2, x5, x7, x12}, {x3}, {x4}, {x6}, {x8}                                  |
| 0.74    | 11           | {x5, x7}, {x1}, {x2}, {x3}, {x4}, {x6}, {x8}, {x9}, {x10}, {x11}, {x12}        |
| 0.76    | 10           | {x1, x2, x12}, {x3}, {x4}, {x5}, {x6}, {x7}, {x8}, {x9}, {x10}, {x11}, {x12}   |
| 0.77    | 11           | {x1, x12}, {x2}, {x3}, {x4}, {x5}, {x6}, {x7}, {x8}, {x9}, {x10}, {x11}, {x12} |
| 0.83    | 11           | {x4, x9}, {x1}, {x2}, {x3}, {x5}, {x6}, {x7}, {x8}, {x10}, {x11}, {x12}        |

Допустимые затраты: 1000

| №: | Коэффициент распределения затрат | Допустимые затраты | Распределение затрат |
|----|----------------------------------|--------------------|----------------------|
| 1  | 0.0891                           | 1000               | 89.1                 |
| 2  | 0.0891                           | 1000               | 89.1                 |
| 3  | 0.0792                           | 1000               | 79.2                 |
| 4  | 0.0792                           | 1000               | 79.2                 |
| 5  | 0.0891                           | 1000               | 89.1                 |

(с) Курочкин Андрей Олегович

Рисунок В.9 - Комп'ютерні експерименти (кількість БД – 12)

Ранжирование элементов

Размерность массива: 13

Очистить      Рассчитать

| №:     | x6   | x7 | x8 | x9 | x10 | x11 | x12 | x13 |
|--------|------|----|----|----|-----|-----|-----|-----|
| x5 2   | x1 6 | 7  | 2  | 3  | 9   | 2   | 6   | 8   |
| x6 3   | x2 8 | 7  | 6  | 3  | 6   | 4   | 8   | 6   |
| x7 6   | x3 4 | 4  | 7  | 5  | 2   | 8   | 6   | 7   |
| x8 9   | x4 1 | 1  | 7  | 8  | 8   | 7   | 6   | 2   |
| x9 10  | x5 4 | 8  | 7  | 2  | 6   | 6   | 4   | 7   |
| x10 11 | x6 0 | 5  | 9  | 8  | 7   | 5   | 2   | 4   |
| x11 12 | x7 1 | 0  | 9  | 6  | 6   | 3   | 1   | 6   |
| x12 13 | x8 1 | 3  | 0  | 1  | 5   | 6   | 6   | 2   |
| x13 10 | x9 7 | 7  | 8  | 0  | 1   | 3   | 2   | 4   |

| Уровень | К-во классов | Классы элементов  |
|---------|--------------|---|
| 0.66    | 1            | {x1, x2, x3, x4, x5, x6, x7, x8, x9, x10, ...}                                    |
| 0.69    | 3            | {x1, x2, x3, x4, x5, x6, x7, x8, x11, x12, ...}                                   |
| 0.7     | 12           | {x9, x10}, {x1}, {x2}, {x3}, {x4}, {x5}, {x6}, {x7}, {x8}, {x11}, {x12}, {x13}    |
| 0.73    | 4            | {x1, x2, x4, x5, x6, x7, x8, x11, x12, x13}, {x3}, {x9}, {x10}                    |
| 0.75    | 10           | {x6, x8, x12, x13}, {x1}, {x2}, {x3}, {x4}, {x5}, {x7}, {x9}, {x10}, {x11}, {x12} |
| 0.76    | 12           | {x1, x2}, {x3}, {x4}, {x5}, {x6}, {x7}, {x8}, {x9}, {x10}, {x11}, {x12}, {x13}    |
| 0.78    | 12           | {x8, x13}, {x1}, {x2}, {x3}, {x4}, {x5}, {x6}, {x7}, {x9}, {x10}, {x11}, {x12}    |
| 0.8     | 12           | {x6, x12}, {x1}, {x2}, {x3}, {x4}, {x5}, {x7}, {x8}, {x9}, {x10}, {x11}, {x13}    |
| 1       | 13           | {x1}, {x2}, {x3}, {x4}, {x5}, {x6}, {x7}, {x8}, {x9}, {x10}, {x11}, {x12}, {x13}  |

Допустимые затраты: 1000

| №: | Коэффициент распределения затрат | Допустимые затраты | Распределение затрат |
|----|----------------------------------|--------------------|----------------------|
| 10 | 0.0744                           | 1000               | 74.4                 |
| 11 | 0.0744                           | 1000               | 74.4                 |
| 12 | 0.0826                           | 1000               | 82.6                 |
| 13 | 0.0826                           | 1000               | 82.6                 |

(с) Курочкин Андрей Олегович

Рисунок В.10 - Комп'ютерні експерименти (кількість БД – 13)



Ранжирование элементов

Размерность массива: 14

Очистить

| №   | x7 | x8 | x9 | x10 | x11 | x12 | x13 | x14 |
|-----|----|----|----|-----|-----|-----|-----|-----|
| x1  | 2  | 7  | 8  | 6   | 8   | 2   | 7   | 7   |
| x2  | 8  | 4  | 6  | 8   | 4   | 1   | 3   | 5   |
| x3  | 2  | 8  | 6  | 2   | 4   | 2   | 2   | 5   |
| x4  | 7  | 3  | 7  | 7   | 1   | 4   | 3   | 4   |
| x5  | 1  | 2  | 7  | 6   | 6   | 7   | 4   | 3   |
| x6  | 3  | 1  | 5  | 3   | 1   | 2   | 8   | 2   |
| x7  | 0  | 2  | 1  | 2   | 5   | 2   | 2   | 3   |
| x8  | 1  | 0  | 5  | 2   | 6   | 5   | 2   | 5   |
| x9  | 5  | 4  | 0  | 6   | 8   | 6   | 6   | 3   |
| x10 | 2  |    |    |     |     |     |     |     |

| Уровень | Кво классов | Классы элементов                                |
|---------|-------------|---|
| 0.65    | 1           | {x1, x2, x3, x4, x5, x6, x7, x8, x9, x10, ...   |
| 0.69    | 2           | {x1, x3, x4, x5, x6, x7, x8, x9, x10, x11, ...  |
| 0.7     | 6           | {x1, x3, x5, x7, x8, x9, x11, x12, x13}, {...   |
| 0.71    | 6           | {x1, x3, x5, x7, x8, x9, x11, x12, x13}, {...   |
| 0.73    | 13          | {x4, x14}, {x1}, {x2}, {x3}, {x5}, {x6}, {x...} |
| 0.75    | 12          | {x3, x7, x13}, {x1}, {x2}, {x4}, {x5}, {x6...}  |
| 0.79    | 13          | {x5, x9}, {x1}, {x2}, {x3}, {x4}, {x6}, {x7...} |
| 1       | 14          | {x1}, {x2}, {x3}, {x4}, {x5}, {x6}, {x7}, {...  |

Допустимые затраты: 1000

| № | Коэффициент распределения затрат | Допустимые затраты | Распределение затрат |
|---|----------------------------------|--------------------|----------------------|
| 1 | 0.0719                           | 1000               | 71,9                 |
| 2 | 0.0647                           | 1000               | 64,7                 |
| 3 | 0.0719                           | 1000               | 71,9                 |
| 4 | 0.0719                           | 1000               | 71,9                 |
| 5 | 0.0719                           | 1000               | 71,9                 |

(с) Курочкин Андрей Олегович

Рисунок В.11 - Комп'ютерні експерименти (кількість БД – 14)

Ранжирование элементов

Размерность массива: 15

Очистить

| №   | x1 | x2 | x3 | x4 | x5 | x6 | x7 | x8 | x1 |
|-----|----|----|----|----|----|----|----|----|----|
| x7  | 0  | 1  | 4  | 1  | 1  | 7  | 8  | 1  | 8  |
| x8  | 1  | 0  | 2  | 3  | 8  | 8  | 8  | 2  | 1  |
| x9  | 5  | 8  | 0  | 5  | 7  | 6  | 6  | 8  | 8  |
| x10 | 9  | 5  | 7  | 0  | 1  | 5  | 6  | 3  | 2  |
| x11 | 5  | 4  | 9  | 2  | 0  | 3  | 1  | 7  | 5  |
| x12 | 2  | 1  | 9  | 8  | 5  | 0  | 4  | 8  | 7  |
| x13 | 4  | 4  | 8  | 5  | 7  | 2  | 0  | 7  | 1  |
| x14 | 3  | 1  | 2  | 7  | 5  | 3  | 2  | 0  | 2  |
| x15 | 1  | 3  | 2  | 5  | 1  | 2  | 1  | 2  | 0  |

| Уровень | Кво классов | Классы элементов                                |
|---------|-------------|---|
| 0.7     | 1           | {x1, x2, x3, x4, x5, x6, x7, x8, x9, x10, ...   |
| 0.71    | 2           | {x1, x2, x3, x4, x5, x6, x7, x8, x9, x10, ...   |
| 0.72    | 7           | {x1, x2, x4, x5, x6, x7, x8, x9, x12}, {x3...}  |
| 0.74    | 13          | {x4, x5, x12}, {x1}, {x2}, {x3}, {x6}, {x7...}  |
| 0.77    | 14          | {x5, x12}, {x1}, {x2}, {x3}, {x4}, {x6}, {x...} |
| 1       | 15          | {x1}, {x2}, {x3}, {x4}, {x5}, {x6}, {x7}, {...  |

Допустимые затраты: 1000

| № | Коэффициент распределения затрат | Допустимые затраты | Распределение затрат |
|---|----------------------------------|--------------------|----------------------|
| 1 | 0.0667                           | 1000               | 66,7                 |
| 2 | 0.0667                           | 1000               | 66,7                 |
| 3 | 0.0667                           | 1000               | 66,7                 |
| 4 | 0.0667                           | 1000               | 66,7                 |
| 5 | 0.0667                           | 1000               | 66,7                 |

(с) Курочкин Андрей Олегович

Рисунок В.12 - Комп'ютерні експерименти(кількість БД – 15)

## ДОДАТОК Г. ДИПЛОМИ ТА ГРАМОТИ

Дипломи та грамоти, отримані у всеукраїнських конкурсах  
студентських наукових робіт



МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ  
УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ



# ДИПЛОМ

НАГОРОДЖУЄТЬСЯ

Курочкін Андрій Олегович

Сумський державний університет

за II місце

у II турі Всеукраїнського конкурсу студентських  
наукових робіт 2018/2019 навчального року  
зі спеціальності «Комп'ютерні науки»

Голова галузевої  
конкурсної комісії  
25 квітня 2019 р.



І.В. Рубан





НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

# ДИПЛОМ

III СТУПЕНЯ

нагороджується

**Курочкін Андрій Олегович**

студент Сумського державного університету

**ПЕРЕМОЖЕЦЬ**

другого туру Всеукраїнського конкурсу студентських наукових  
робіт з галузей знань і спеціальностей у 2018/2019  
навчальному році за спеціальністю «Кібербезпека»

Голова галузевої конкурсної комісії,  
проректор з науково-педагогічної роботи



О.М. Новіков  
05 квітня 2019 р.

