

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

**КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА**

**на тему:** «Моделі вибору технологій забезпечення цілісності баз даних в умовах конфліктних взаємодій»

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

**Виконавець роботи:** студент групи ІТ-51 Курочкін Андрій Олегович

**Кваліфікаційна робота бакалавра  
захищена на засіданні ЕК  
з оцінкою**

\_\_\_\_\_ «\_\_» \_\_\_\_\_ 2019 р.

Науковий керівник

\_\_\_\_\_  
(підпис)

професор Лавров Є.А.  
(науковий ступінь, вчене звання, прізвище та ініціали)

Голова комісії

\_\_\_\_\_  
(підпис)

Шифрін Д. М.  
(науковий ступінь, вчене звання, прізвище та ініціали)

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

Студент \_\_\_\_\_  
(підпис)

Суми-2019

**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  
«\_\_\_\_\_» \_\_\_\_ 2019 р.

**З А В Д А Н Н Я**  
**НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА СТУДЕНТУ**

Курочкіну Андрію Олеговичу

**1 Тема роботи** Моделі вибору технологій забезпечення цілісності баз даних в умовах конфліктних взаємодій

керівник роботи Лавров Євгеній Анатолійович, професор,

затверджені наказом по університету від «17» травня 2019 р. № 0834-III

**2 Строк подання студентом роботи** «3» червня 2019 р.

**3 Вхідні дані до роботи**

Організаційно-технічні заходи, система управління залізничним транспортом, методи ранжування, вразливості інформаційної системи, система взаємопов'язаних баз даних

**4 Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)**

вступ, аналіз предметної області, моделювання процесів розробки методу вибору технологій забезпечення цілісності баз даних інформаційної системи в умовах конфліктної взаємодії, розробка моделі вибору технологій забезпечення цілісності баз даних інформаційної системи в умовах конфліктної взаємодії, висновки, список використаної літератури

**5 Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)**

Втрати від кібератак, проблема ненадійності, система забезпечення цілісності бд, типові задачі забезпечення цілісності, моделювання розробки методу, задача, приклад реальної системи управління залізницею, ранжування елементів інформаційної системи, задача 2, модель функціонування інформаційної системи, напівмарківський процес прийняття рішень, приклад визначення коефіцієнтів впливу порушень, інформаційна технологія, впровадження, апробація, висновки

**6. Консультанти розділів роботи:**

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання \_\_\_\_\_ 3.09.2018 \_\_\_\_\_

**КАЛЕНДАРНИЙ ПЛАН**

№ п/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1.	Ознайомлення з поточним станом втрат від кібератак та негативних впливів на Україні в цілому	01.04.19-02.04.19	
2.	Визначення проблеми ненадійності	03.04.19-03.04.19	
3.	Ідентифікація мети та задач	04.04.19-04.04.19	
4.	Аналіз документації	05.04.19-10.04.19	
5.	Визначення інструментарію	11.04.19-10.04.19	
6.	Розроблення методу ранжування	19.04.19-23.04.19	
7.	Реалізація інформаційної технології ранжування	24.04.19-26.04.19	
8.	Розроблення системи виявлення і усунення негативного впливу бази даних	29.05.19-06.05.19	
9.	Аналіз результатів	07.05.19-15.05.19	
10.	Оформлення документації	16.04.19-30.05.19	

Студент

\_\_\_\_\_

(підпис)

Курочкін А.О.

Керівник роботи

\_\_\_\_\_

(підпис)

д.т.н., професор Лавров Є.А.

## РЕФЕРАТ

Тема бакалаврської роботи: «Моделі вибору технологій забезпечення цілісності баз даних в умовах конфліктної взаємодії».

Пояснювальна записка містить вступ, 3 розділи, висновки, додатки та список літератури, включає 83 сторінки, 13 таблиць, 37 ілюстрації, 28 джерел.

В першому розділі обґрунтована актуальність дослідження. Проводиться аналіз вразливостей та їх взаємодії з інформаційним забезпеченням, аналіз сучасних методів ранжування елементів системи та формулюється мета, задачі та вибір засобів реалізації.

Другий розділ присвячений моделюванню процесів розробки методу забезпечення цілісності, яка включає в себе структурно-функціональний аналіз проекту, а також варіанти використання даної моделі.

Третій розділ присвячений розробці моделі вибору технологій, яка включає в себе як задачу лінійного програмування, так і інформаційну технологію запропонованого методу ранжування. Описується поетапна реалізація та приводиться приклад роботи програмного забезпечення.

Результатом проведеної роботи є розроблені моделі вибору технологій забезпечення цілісності баз даних в умовах конфліктної взаємодії.

Ключові слова: ранжування, негативний вплив, цілісність баз даних, конфліктні взаємодії, напівмарківські процеси, забезпечення цілісності, транзитивне замикання.

# ЗМІСТ

ВСТУП .....	7
1. АНАЛІЗ ПРОБЛЕМ ЗАБЕЗПЕЧЕННЯ ЦІЛІСНОСТІ ІНФОРМАЦІЙНОЇ СИСТЕМИ В УМОВАХ КОНФЛІКТНОЇ ВЗАЄМОДІЇ.....	8
1.1 Характеристик сучасних проблем забезпечення цілісності інформаційних систем .....	8
1.2. Ранжування елементів системи для задач забезпечення цілісності інформаційних систем .....	10
1.3. Постановка задачі .....	14
1.3.1. Мета та задачі.....	14
1.3.2. Вибір засобів реалізації.....	14
2. МОДЕЛЮВАННЯ ПРОЦЕСІВ РОЗРОБКИ МЕТОДУ ВИБОРУ ТЕХНОЛОГІЙ ЗАБЕЗПЕЧЕННЯ ЦІЛІСНОСТІ БАЗ ДАНИХ ІНФОРМАЦІЙНОЇ СИСТЕМИ В УМОВАХ КОНФЛІКТНОЇ ВЗАЄМОДІЇ	16
2.1. Структурно функціональне моделювання процесів розробки методу вибору технологій забезпечення цілісності баз даних інформаційної системи в умовах конфліктної взаємодії .....	16
2.2. Моделювання варіантів використання .....	22
3. РОЗРОБКА МОДЕЛІ ВИБОРУ ТЕХНОЛОГІЙ ЗАБЕЗПЕЧЕННЯ ЦІЛІСНОСТІ БАЗ ДАНИХ ІНФОРМАЦІЙНОЇ СИСТЕМИ В УМОВАХ КОНФЛІКТНОЇ ВЗАЄМОДІЇ.....	24
3.1. Модель ранжування на основі транзитивного замикання нечіткого відношення подібності .....	24
3.2. Приклад визначення коефіцієнтів впливу порушень цілісності баз даних на ефективність інформаційної системи.....	28

3.3. Інформаційна технологія методу ранжування на основі нечіткого відношення подібності .....	34
3.4. Розподіл обмежених ресурсів для вибору стратегій технічного обслуговування баз даних .....	44
ВИСНОВКИ .....	51
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ.....	52
Додаток А. Технічне завдання.....	55
Додаток Б. Планування робіт .....	57
Додаток В. Лістинг програми .....	63
Додаток Г. Комп'ютерні експерименти.....	73
Додаток І. Акт впровадження.....	79
Додаток Д. Дипломи та грамоти .....	81

## ВСТУП

**Актуальність.** Останнім часом збільшують збитки від кібератак та негативного впливу на інформаційні системи. Не дивлячись на кількість наукових досліджень, задача оптимізації заходів з питань забезпечення цілісності баз даних вирішена не до кінця.

**Предмет дослідження.** Технології забезпечення цілісності баз даних в умовах конфліктних взаємодій.

**Об'єкт дослідження.** Цілісність баз даних інформаційної системи.

**Мета дослідження.** Розробити моделі і інформаційну технологію вибору організаційно-технічних заходів забезпечення цілісності баз даних інформаційної системи в умовах конфліктних взаємодій.

**Наукова новизна.** Розроблений метод використовує моделі виникнення і усунення пошкоджень, побудовану з використанням апарату напівмарківських процесів, і технологію ранжування баз даних.

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

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

**Публікації.** Результати дослідження опубліковано на порталі студентського конкурсу наукових робіт [19].

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

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

# **1. Аналіз проблем забезпечення цілісності інформаційної системи в умовах конфліктної взаємодії**

## **1.1 Характеристик сучасних проблем забезпечення цілісності інформаційних систем**

Ускладнення завдань, що виконуються сучасними інформаційними системами (ІС), розвиток використовуваних в них інформаційних технологій, а також виникнення умов функціонування, пов'язаних з можливостями навмисних і ненавмисних негативних впливів (НВ) вимагає нових підходів до аналізу та прогнозування цілісності ІС.

Під цілісністю ІС розуміється властивість інформаційної системи зберігати свої характеристики в даних умовах експлуатації [1]. Під інформаційними технологіями в даній роботі розуміються технології використання комп'ютерних систем і телекомунікаційного устаткування для зберігання, обробки, передачі та управління даними [2].

Зокрема, технології зберігання, обробки, передачі та управління даними (інформаційні технології) реалізуються в програмному забезпеченні (ПЗ) ІС (тобто програмне забезпечення здійснює всі вищенаведені функції). Таким чином, питання про надійність використання ПЗ ІС, що розглядається в даній роботі, є окремим випадком більш широкого питання про надійність використання ІТ.

Одним з найважливіших аспектів досліджень в області аналізу і синтезу ІС є конфліктний аспект цілісності використання ПЗ ІС, який передбачає облік конфліктного характеру інформаційних взаємодій. Тут під конфліктним взаємодією мається на увазі можливість джерел негативних впливів (ДНВ), як навмисних, так і ненавмисних, вплинути на ІС, а, з іншого боку, робота системного



адміністратора (персоналу, що обслуговує інформаційну систему), спрямована на збереження працездатності ПО ІС і запобігання наслідків цього впливу. При цьому наявність двох сторін з різними інтересами і цілями, різними можливостями і умовами їх реалізації накладає особливі, часто детерміновані, правила розвитку конфліктної взаємодії [3,4,5].

Як правило, зовнішні НВ на ІС здійснюються за рахунок використання одного з класів дефектів ПЗ [6] (помилки ПО [7,8]), які в науковій літературі називаються вразливостями [3,4,5,6]. Згідно [9], використовуючи уразливості в ПЗ ІС, можна порушити конфіденційність, доступність і/або цілісність інформації. При цьому порушення доступності і/або цілісності інформації, зазвичай призводить до відмов [10] в роботі ІС, відповідно, можна стверджувати, що використовуються при цьому уразливості впливають на надійність [10] ІС.

Кількість вразливостей, які виявляються в ПЗ, з кожним роком зростає [11-14]. Зростає також середній час їх усунення [11-14], і росте число експлоїтів (програм або скриптів, які використовують вразливості для НВ на ІС) [11-14]. Дана ситуація підвищує можливості успішного НВ на ІС, що відбивається зростанням числа спроб НВ на ІС [15]. Більш того, статистика показує, що якщо раніше в основному навмисне негативний вплив виявлялося на ІС великих компаній, то зараз НВ все більше і більше піддаються ІС середніх і малих компаній [15]. Відповідно, зростають сукупні матеріальні, репутаційні та інші втрати. У зв'язку з цим користувачі і розробники ПЗ потребують вироблення адекватних методів оцінки надійності використання ПЗ в ІС в умовах внутрішніх дефектів (вразливостей) і конфліктних взаємодій.

## 1.2. Ранжування елементів системи для задач забезпечення цілісності інформаційних систем

На ранніх етапах проектування системи виникає необхідність оцінки рангів її елементів. Ранг - це кількісна характеристика важливості елементу, яка використовується при вирішенні наступних завдань:

- формування вимог до цілісності елементів, виходячи із заданих вимог до надійності системи в цілому;

- розподіл коштів на підвищення цілісності системи між її елементами;

Класичний підхід до обчислення рангів використовує чутливість функції надійності системи до зміни надійності її елементів.

Альтернативою цьому підходу є експертні оцінки, які формалізуються засобами нечіткої (fuzzy) математики [23,24].

Ранжування являє собою процедуру упорядкування об'єктів і виконується особою або експертом, яка займається ранжуванням. За допомогою знань і досліджень особа чи експерт розпоряджують об'єкти у порядку переваги, керуючись показниками порівняння. В залежності від категорії відносин між об'єктами існують різні види порядку об'єктів.

Наприклад: нехай серед об'єктів є лише різні по порівнянню показники, тобто немає еквівалентних об'єктів. Тоді між об'єктами створюється тільки відношення строгого порядку. У результаті порівняння об'єктів по антирефлексивному відношенню складається ланцюгова послідовність, де об'єкт з першим номером є найкращим [21]. Об'єкт з другим номером менш кращий першому об'єкту, але краще всіх інших об'єктів і таким чином до останнього об'єкту дослідження.

Види ранжирування можна виявити на підставі розрізнення цілей дослідження (ранжування установок, цінностей, стимулів, способів оцінювання і т. д.) і предмету дослідження.

З огляду на кількість процедур ранжування, можна розрізнити просте (разове) і складне (n-разове) ранжування. Особливо цікаві результати складного ранжування, так як одержані в результаті порядкові шкали при зіставленні і застосуванні спеціальних прийомів вибіркового аналізу дозволяють робити висновки щодо заходів погодження думок, цінностей, позицій опитуваних, тощо. Повторне ранжування елементів аналізу з певним часовим інтервалом (панельне ранжування) дозволяє виявити динаміку як окремої особистості, так і цілої соціальної групи.

Моніторингове (багатопланове, регулярно проводиться) ранжування дозволяє не тільки визначити тенденції зміни досліджуваного об'єкта, а й вибудовувати прогнози.

За способом оцінювання можна розрізнити висхідне і низхідне ранжування (основа – напрямок оцінювання). З огляду на спосіб порівняння, розрізняють одночасне ранжування та поетапне ранжування. Поетапне ранжування найчастіше зустрічається в двох видах: ранжування за допомогою парних порівнянь і ранжування за допомогою блокових порівнянь. Іноді застосовується методика ранжування за допомогою множинних порівнянь [28].

Поетапне ранжування дозволяє розбити на фрагменти і, тим самим, спростити процедуру ранжирування. Особливий інтерес цей метод викликає у дослідників, які прагнуть виявити приховані (в тому числі свідомо приховувані) переваги, протиріччя в думках, оцінках, системах цінностей особистості і соціальних груп і в ряді інших випадків (наприклад, коли дослідник припускає, що

опитуваним важко буде впоратися з одночасним ранжируванням пропонованих елементів аналізу).

Дещо детальніше слід зупинитися на описі методики парних порівнянь, яка може бути рекомендована для використання при проведенні навчальних емпіричних досліджень надійності ІС. За основу методики парних порівнянь взята процедура поетапного попарного співвіднесення всіх елементів аналізу. При цьому число парних порівнянь, які увійшли в шкалу аналізу, буде дорівнює числу всіх можливих поєднань елементів аналізу [27].

Метод аналізу ієрархій (МАІ) , запропонований професором Томасом Л. Сааті, заснований на методиці парних порівняннях альтернативних варіантів, використовуючи різні критерії з використанням 9-бальної шкали Сааті і подальшим ранжуванням низки альтернатив, які відповідають критеріям і цілям [20]. Значення відношень між критеріями враховуються за допомогою ієрархії критеріїв і застосуванням методики парних порівнянь для кількісного визначення важливості критеріїв [25].

До основних процедур методу аналізу ієрархій відносяться наступні:

- генерація безлічі альтернативних варіантів;
- створення множини критеріїв для оцінки альтернатив і графічного представлення у вигляді ієрархії;
- визначення переваг експертами на множині альтернатив за допомогою порівняння критеріїв;
- визначення важливості критеріїв на спільну мету ранжування і інші альтернативи;
- отримання альтернатив за всіма критеріями і цілям, які були створені за допомогою ранжування.

Всі оцінки визначаються експертами. На першому етапі групою експертів визначається множина допустимих альтернатив, серед яких необхідно обрати кращу альтернативу або виконати їх упорядкування. Вершиною ієрархії зазвичай є глобальна ціль або мета дослідження, на наступних рівнях - відображені критерії, обрані групою експертів і на нижчому рівні – множина альтернатив альтернативи.

Ієрархічна структура критеріїв і цілей є моделлю знань конкретної предметної області, яка змінюється і уточнюється з плином часу [22].

Другим етапом елементи одного рівня ієрархії порівнюються по величині їх впливу на елементи більш високого рівня. Результати обчислень заносяться в матрицю попарних порівнянь. При порівнянні елемента із собою маємо рівну значущість «1», тобто головна діагональ матриці складається з одиниць.

Основною перевагою методу аналізу ієрархій є висока універсальність - метод застосовується для вирішення завдань різного типу складності: аналіз всіх можливих варіантів розвитку тієї чи іншої ситуації, упорядкування ресурсів, визначення рейтингу групи працівників організації, допомога з прийняттям рішень про працевлаштування та ін.

У свою чергу, основним недоліком даного методу є необхідність отримання дуже великого обсягу інформації від експертів. Для оптимального використання методу, його використовують для випадків, коли основна частина даних заснована на перевагах групи експертів, яка приймає рішення, під час процесу обрання найкращої альтернативи з множини вже існуючих альтернатив .

## 1.3. Постановка задачі

### 1.3.1. Мета та задачі

У зв'язку з величезними втратами від кібер-атак, негативного впливу і невирішеністю в цілому питання захисту від конфліктної взаємодії, переді мною було поставлено задачу розробки методів забезпечення цілісності баз даних інформаційних систем в умовах конфліктних взаємодій.

Аналіз предметної області дав змогу сформулювати наступні задачі проекту:

- планування робіт проекту;
- аналіз існуючих методів ранжування;
- розроблення інформаційної технології ранжування елементів інформаційної системи;
- розроблення узагальненої моделі виявлення і усунення негативного впливу на бази даних;
- комп'ютерне моделювання процесів виникнення і усунення негативного впливу;

### 1.3.2. Вибір засобів реалізації

Для розроблення моделей забезпечення цілісностей були застосовані програма Microsoft Excel 2016, мова програмування C# та програмне середовище Visual Studio 2015.

Microsoft Excel - програмне забезпечення, створене компанією Microsoft, яке використовує електронні таблиці для організації чисел і даних за допомогою

формул і функцій. Аналіз Excel є повсюдним у всьому світі і використовується компаніями будь-якого розміру для здійснення складних математичних обчислень.

C# - об'єктно-орієнтована мова програмування, яка використовує платформу .NET як безпечну типізацію.

Будучи потужною, гнучкою і добре підтримуваною, C# швидко стала однією з найпопулярніших доступних мов програмування. Сьогодні це четверта за популярністю мова програмування, причому близько 31% всіх розробників використовують її регулярно. Це також третє найбільше співтовариство на форумі StackOverflow (яке було побудовано з використанням C#) з більш ніж 1,1 мільйонами записів [18].

У свою чергу, для розроблення ПЗ використано середовище Visual Studio 2015, основними перевагами якої є:

- повнофункціональна інтегроване середовище розробки для будь-якої операційної системи;
- доступ до великої бібліотеки програмного забезпечення Майкрософт актуальних і минулих версій;
- Технічна та професійна підтримка світового класу

## **2. Моделювання процесів розробки методу вибору технологій забезпечення цілісності баз даних інформаційної системи в умовах конфліктної взаємодії**

Визначені структури робіт, ризику, команди проекту та матриці відповідальності дають можливість виконати структурно-функціональне моделювання процесу моделей вибору організаційно-технічних заходів. Для оптимального опису об'єкту дослідження використовується методологія графічного і функціонального моделювання процесів IDEF0. Техніка IDEF0 використовує діаграми для представлення функцій у процесі і показує взаємозв'язок з дочірніми системами.

### **2.1. Структурно функціональне моделювання процесів розробки методу вибору технологій забезпечення цілісності баз даних інформаційної системи в умовах конфліктної взаємодії**

Для розуміння процесу моделі вибору технологій та заходів структура самої моделі складається з 3 частин функціонування та тестування всього процесу. На рис. 2.1 зображена контекстна діаграма - представляє вид деревовидної структури діаграм, показує призначення системи і її взаємодії з іншими механізмами. В окремій моделі знаходиться лише одна контекстна діаграма. Після описання основних функцій виконується функціональна декомпозиція, визначаються функції, які є основними. В IDEF0 розрізняють наступні види стрілок:



Вхід – категорія об’єктів, що використовуються для отримання результату і потрібні для виконання тієї чи іншої роботи.

Управління - інформація, керуюча діями роботи. Зазвичай керуючі стрілки несуть інформацію, яка вказує, що повинна виконувати робота. Механізм - ресурси, які необхідні для виконання тієї чи іншої роботи. Ресурси можуть бути представлені як у вигляді матеріалу, так і у вигляді команди, яка виконує роботу.

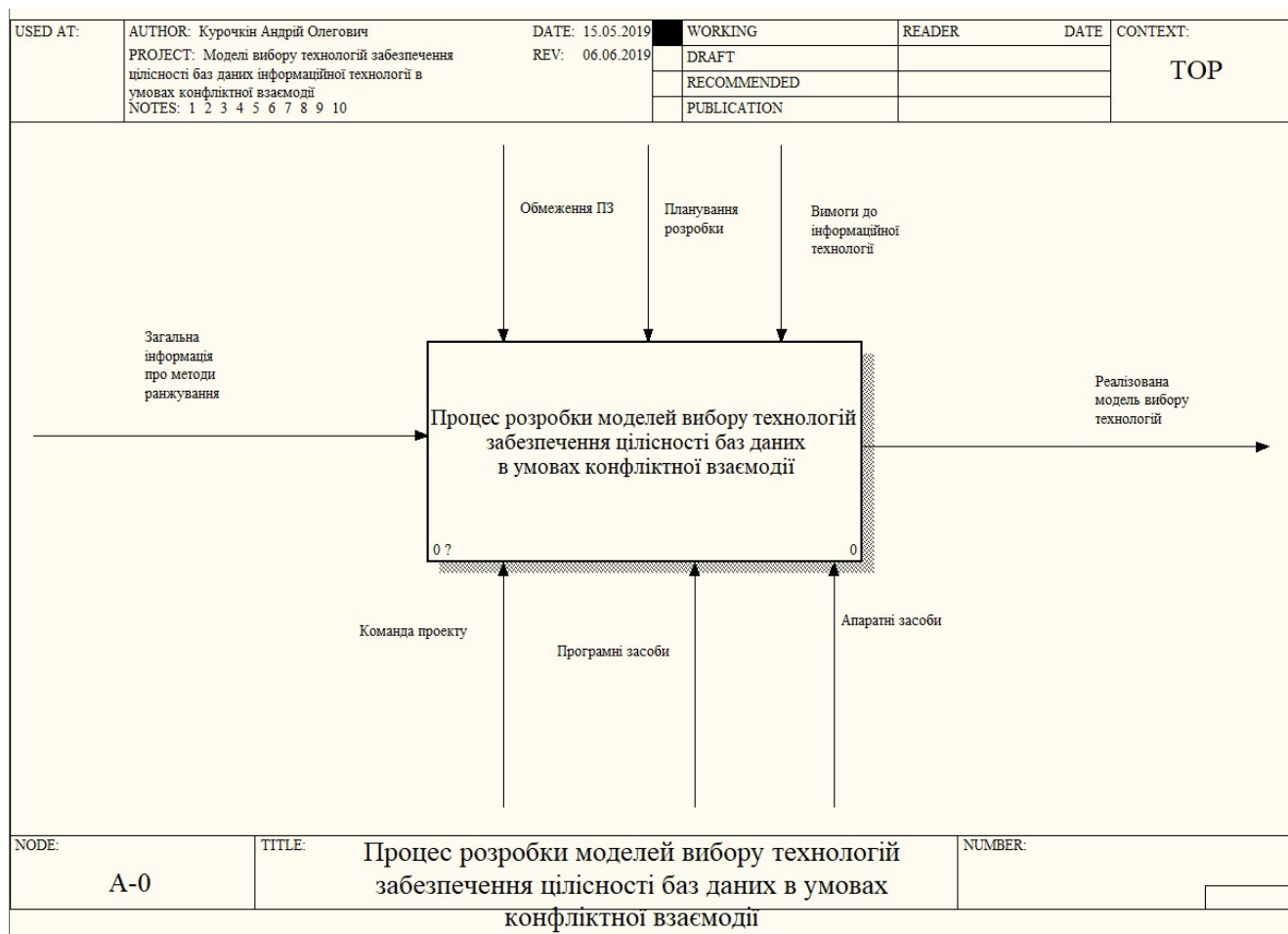


Рисунок 2.1 – Контекстна діаграма IDEF0

На вході маємо технічне завдання, яке визначає основне завдання для створення підмоделей та інформаційної технології. Відповідно на виході отримуємо вже модель вибору. Механізмами виступають команда проекту, до якої входить розробник, програмні засоби (програмне середовище, тощо), апаратні засоби. Усі процеси моделі регулюються обмеженнями ПЗ, плануванням робіт та

вимогами до інформаційної технології, які визначають, як саме повинна виглядати програма.

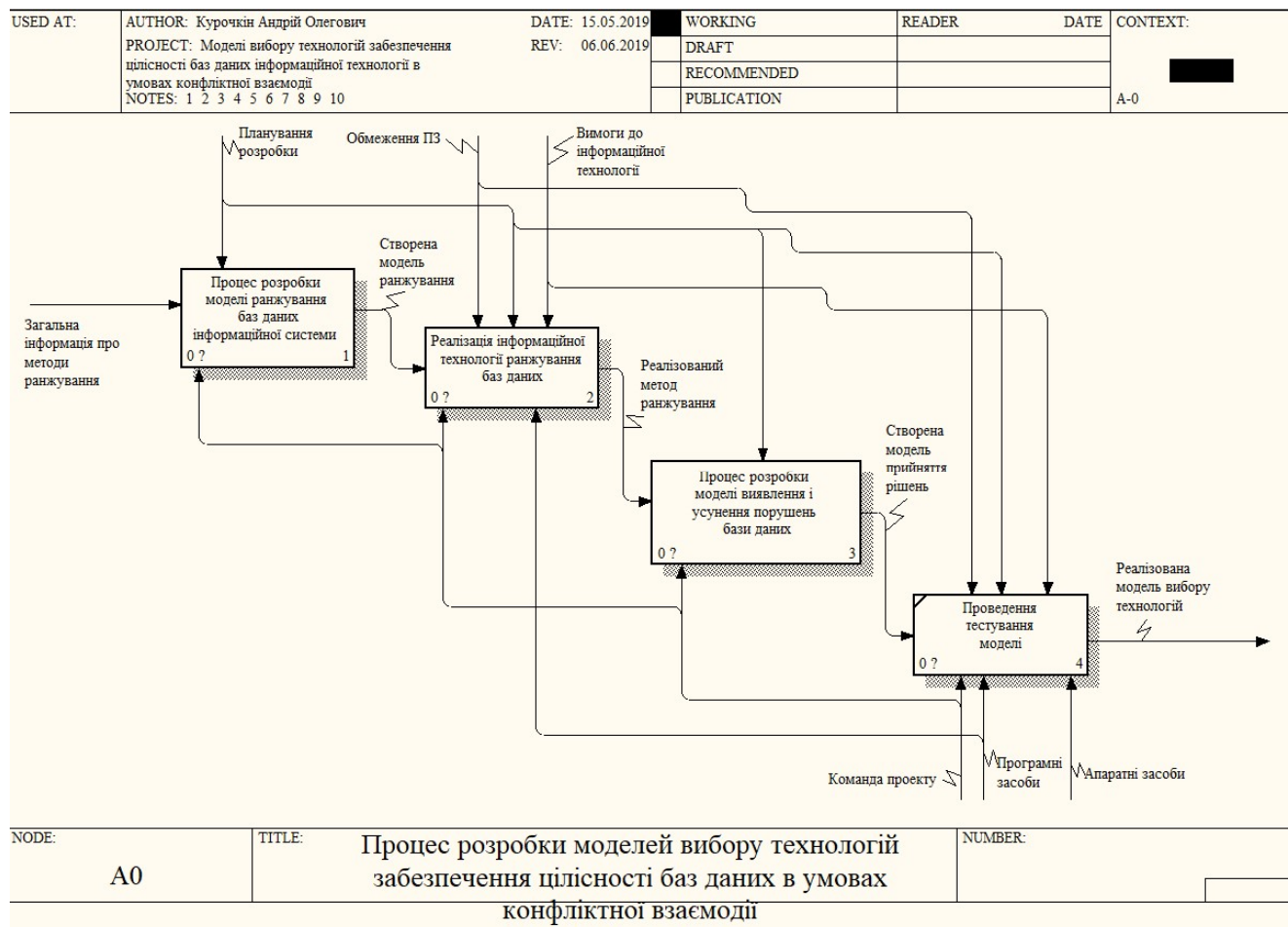


Рисунок 2.2 – Діаграма IDEF0

Контексна діаграма А-0 була розбита на 4 підфункції, такі як: модель ранжування баз даних інформаційної системи, інформаційна технологія ранжування баз даних, модель виявлення і усунення порушень бази даних та тестування моделі. Це обумовлено тим, що алгоритм поділяється на 3 етапи:

- перший етап: визначення коефіцієнтів важливості баз даних;
- другий етап: розроблення інформаційної технології ранжування баз даних;

- третій етап: створення графу подій напівмарківського процесу прийняття рішень;

- четвертий етап: тестування моделей та інформаційних технологій.

Опис першого етапу:

- вхідні дані: загальна інформація про методи ранжування;
- вихідні дані: реалізована модель вибору технологій забезпечення цілісності баз даних інформаційної системи в умовах конфліктної взаємодії;

- управління: планування розробки

- механізми: команда проекту.

Опис другого етапу:

- вхідні дані: створена модель ранжування;
- вихідні дані: реалізований метод ранжування;
- управління: обмеження ПЗ, планування розробки, вимоги до інформаційної технології;

- механізми: команда проекту, програмні засоби, апаратні засоби.

Опис третього етапу:

- вхідні дані: реалізований метод ранжування;
- вихідні дані: створена модель прийняття рішень;
- управління: планування розробки;
- механізми: команда проекту.

Опис четвертого етапу:

- вхідні дані: створена модель прийняття рішень;
- вихідні дані: реалізована модель вибору технологій;
- управління: обмеження ПЗ, вимоги до інформаційної технології, планування розробки;

- механізми: команда проекту, програмні засоби, апаратні засоби.

Діаграма другого рівня представлена на рис.2.2.

Після декомпозиції контекстної діаграми на чотири підфункції було виконано розбиття кожної з них на певний перелік робіт, результат кожної з якої є вхідними даними для наступної.

Модель ранжування баз даних інформаційної системи містить в собі такі етапи, а саме: визначення оцінок впливу баз даних, обчислення ступеней впливу баз даних, нормалізація матриці ступеней впливу, обчислення нечіткого відношення подібності, обчислення транзитивного замикання, визначення коефіцієнтів важливості баз даних.

Для визначення оцінок впливу баз даних визначені такі дані:

- Вхідні дані: технічне завдання.
- Вихідні дані: вхідні дані визначені.
- Управління: команда проекту.
- Механізми: обмеження ПЗ.

Для наступних етапів розроблення методу визначення коефіцієнтів важливості дані управління та механізмів не змінюються. Декомпозиція етапу модель ранжування зображене на рис. 2.3, декомпозиція інформаційної технології – на рис. 2.4, а модель виявлення і усунення порушень бази даних – на рис. 2.5.

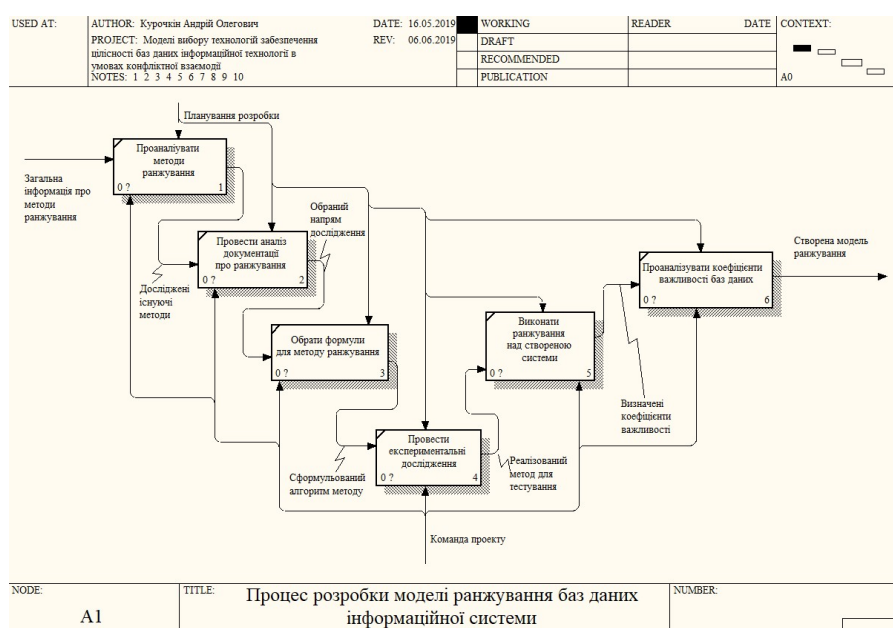


Рисунок 2.3 - Діаграма декомпозиції першого етапу

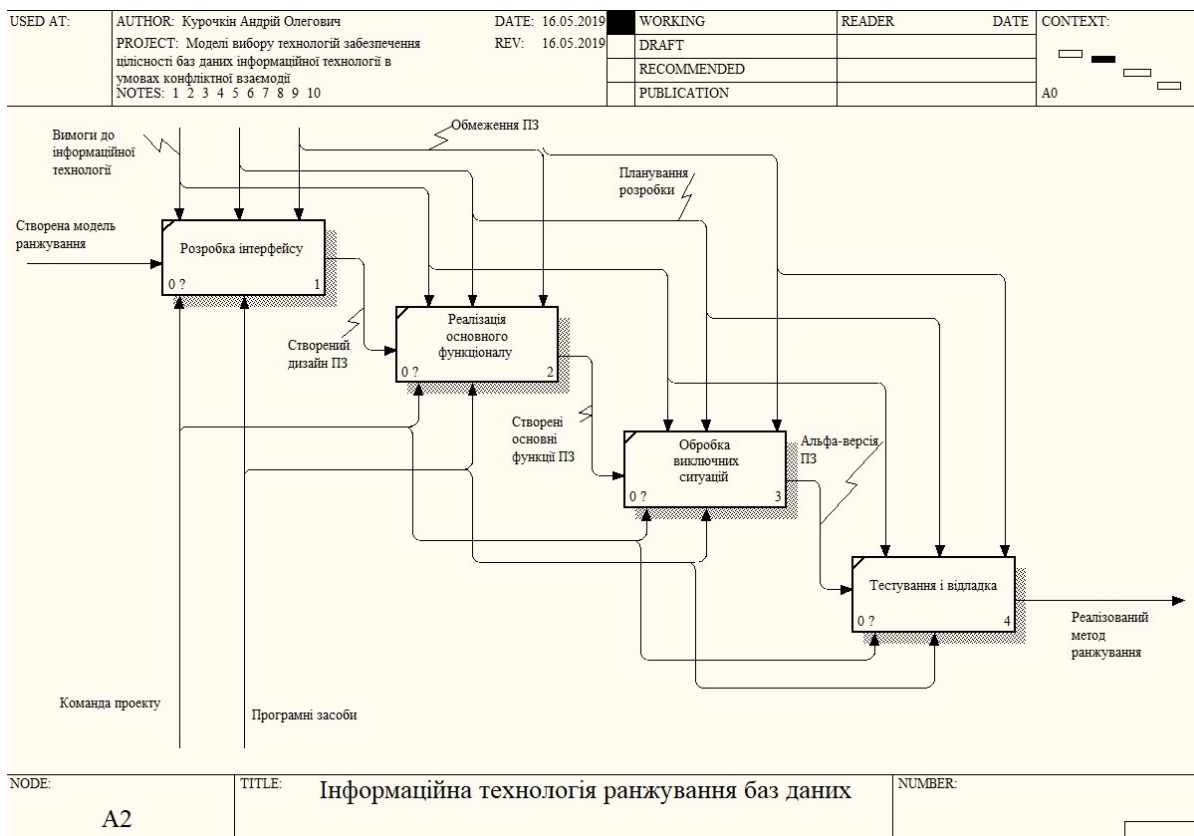


Рисунок 2.4 - Діаграма декомпозиції другого етапу

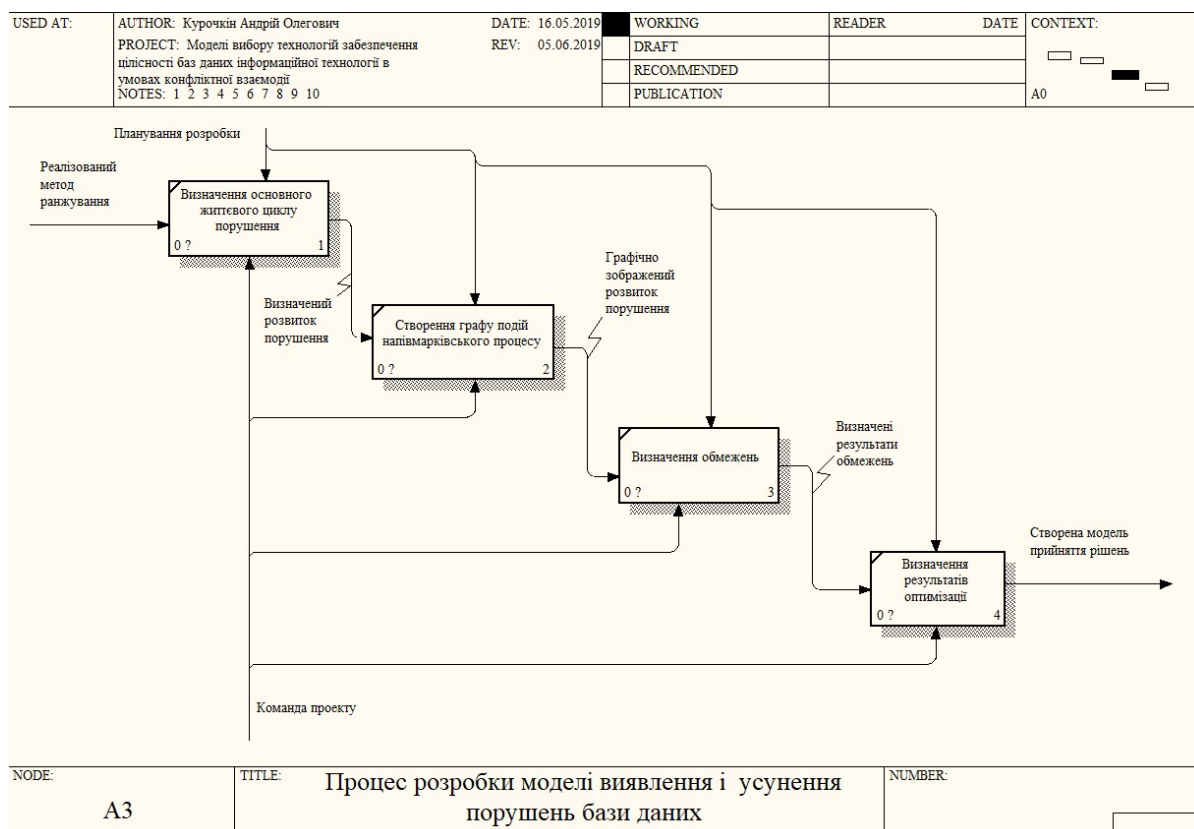


Рисунок 2.5 - Діаграма декомпозиції третього етапу

## 2.2. Моделювання варіантів використання

Мова уніфікованої моделювання (UML) - стандартизована мова моделювання, що дозволяє розробникам вказувати, візуалізувати, конструювати та документувати артефакти програмної системи.

Дана мова була створена для графічного зображення моделі систем з використанням об'єктно-орієнтованого підходу, а також для оптимізації процесу реалізації інформаційних систем, що дозволяє отримати перевагу у вигляді збільшення ефективності реалізації у кілька разів і тим самим поліпшити якість кінцевого ПЗ.

Діаграма варіантів використання є динамічною діаграмою або діаграмою поведінки в UML. Діаграми моделюють функціональність системи за допомогою акторів і варіантів використання. Варіанти використання (use cases) - це набір дій, служб і функцій, які система повинна виконувати. У цьому контексті "система" – модель вибору технологій забезпечення цілісності. "Суб'єкти" - це люди або суб'єкти, що діють у визначених ролях в системі.

Для розробки діаграми варіантів використання були визначені такі актори:

Model – модель в цілому, яка включає в себе модель ранжування баз даних інформаційної системи, її реалізацію та модель виявлення і усунення порушень бази даних.

User- користувач, який використовує модель у власних цілях.

Після того, як були визначені всі актори, які будуть взаємодіяти з системою, необхідно сформулювати перелік варіантів використання. Вони відповідають вимогам, яким повинна відповідати інформаційна технологія, що зазначені у Додатку А:

- визначення коефіцієнтів важливості;

- вибір кількості баз даних у корпоративній мережі;
- визначення кількості організаційно-технічних заходів;
- визначення життєвого циклу порушення;
- визначення рангів баз даних;
- визначення кількості циклів транзитивного замикання

Виходячи з сформованих даних про акторів та всі можливі варіанти моделі вибору технологій забезпечення цілісності, розроблена модель використання діаграма, яка представлена на рис.2.6.

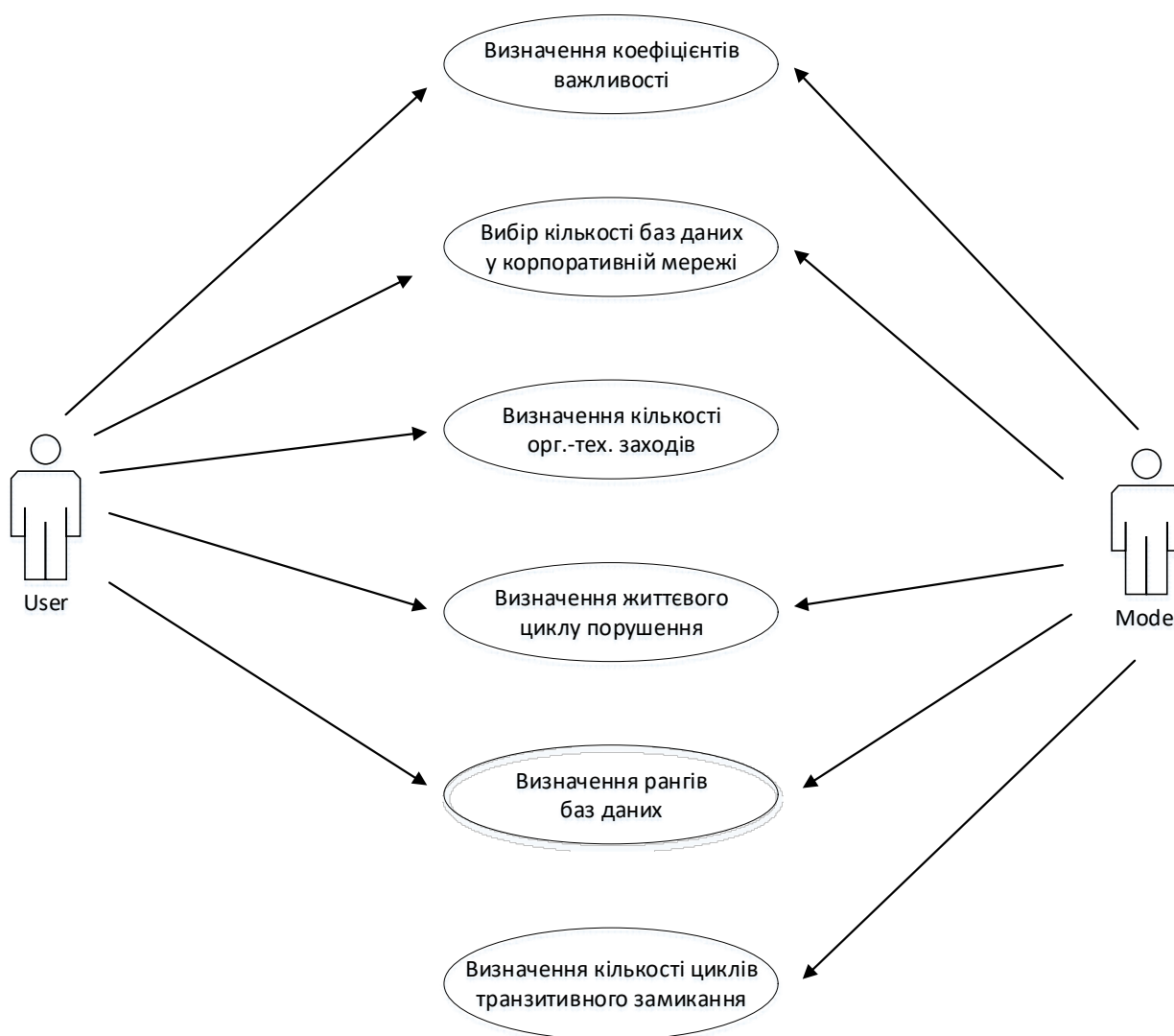


Рисунок 2.6 – Діаграма варіантів використання

### 3. Розробка моделі вибору технологій забезпечення цілісності баз даних інформаційної системи в умовах конфліктної взаємодії

#### 3.1. Модель ранжування на основі транзитивного замикання нечіткого відношення подібності

Задачу ранжування зведено до автоматичного упорядкування на основі нечіткого відношення подібності та транзитивного замикання.

Відомі підходи до ранжування елементів системи розділяють на два класи, зовнішній і внутрішній, спираючись на категорію причинно-спадкових зв'язків між відмовами.

Зовнішній підхід використовує чутливість цілісності системи до зміни цілісності її елементів. Розглянемо функцію цілісності:

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

Дана функція пов'язує ймовірності безвідмовної роботи системи ( $P_s$ ) і її елементів ( $P_i$ ). Уявімо цю функцію у вигляді ряду:

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

Коефіцієнти якого мають сенс часткових похідних:

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

Коефіцієнт  $b_i$  в (3) відповідає індексу важливості  $i$ -го елемента (reliability importance index). Коефіцієнт  $b_{ij}$  в (3) відповідає індексу важливості спільного впливу  $i$ -го і  $j$ -го елементів (Joint reliability importance). Для систем, цілісність яких



моделюється методом Монте-Карло. Обчислення важливості елементів системи безпосередньо на основі логічної (або структурної) функції має наступний вигляд:

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

де  $\alpha_s$  ( $\alpha_i$ ) = 1 (0), якщо система (і-й елемент) працює (не працює),  $f_L$  - булева функція.

Внутрішній підхід пов'язаний з оцінкою значимості елементів на основі теорії відносин і графів.

Внутрішній підхід не вимагає побудови структурної функції (4) і функції цілісності (1). Він спирається на інформацію про структуру системи, тобто склад її елементів і зв'язки між ними. При цьому можуть використовуватися знання про вплив порушень в одних елементах на виникнення порушень в інших елементах. Наприклад, «догляд параметрів і-го елемента призводить до відтоку параметрів j-го елемента, що в свою чергу призводить до відмови k-го елемента, і т.д.». Таким чином, може враховуватися «ефект доміно». Носієм інформації для обчислення рангів в служить матриця зв'язків.

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

В якій  $a_{ij} = 1$  (0), якщо і-й елемент пов'язаний (не пов'язаний) з j-м елементом. Ранг і-го елемента обчислюється як сума елементів і-го рядка матриці

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

Яка враховує однокрокові і двокрокові впливи порушень в і-м елементі системи.

Нижче пропонується рішення задачі ранжування елементів на основі нечіткого транзитивного замикання і спеціальних процедур побудови нечітких відносин впливу і подібності.

Нехай  $\{x_1, x_2, \dots, x_n\}$  – множина елементів системи. Вплив елемента  $x_i \in X$  на інші елементи задаємо нечіткою множиною:

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

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

Будемо припускати, що вплив елемента  $x_i \in X$  на самого себе відсутній, тобто:

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

Сукупність ступенів впливу  $\mu_{ij}$  з (7) для всіх елементів  $x_i \in X$  утворює нечітке відношення впливу  $I$ , яке задане 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 ставиться у відповідність кожній парі елементів  $(x_i, x_j)$ , може здаватися експертно, або методом найменшого впливу, який пропонується нижче. Зауважимо, що подібна процедура обчислення ступенів належності використовувалася раніше в роботі [22].

Нехай  $f_{ij}$  - сила впливу елемента  $x_i \in X$  на елемент  $x_j \in X$ , причому виконується умова: «чим більше сила  $f_{ij}$ , тим більше ступінь впливу  $\mu_{ij}$ », тобто має місце

співвідношення:

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

Передбачається, що відповідно до (8):

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

Нехай  $x_l$  - елемент, на який елемент  $x_i$  має найменший вплив. З (10) маємо:

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

Підставляючи (12) у вимогу:

$$\mu_{i1} + \mu_{i2} + \dots + \mu_{in} = 1, i = 1, 2, \dots, n,$$

Отримуємо найменший ступінь впливу елемента  $x_i \in X$  в системі:

$$\mu_{i1} = \left( \frac{f_{i1}}{f_{i1}} + \frac{f_{i1}}{f_{i2}} + \dots + \frac{f_{i1}}{f_{in}} \right)^{-1} \quad (13)$$

Співвідношення (13) і (12) дозволяють обчислювати ступеня впливу в нечіткому відношенні (9) шляхом порівняння сил впливів  $f_{ij}$  з найменшою силою впливу  $f_{il}$  для кожного елемента  $x_i \in X$ . Для цього використовується 9-бальна шкала 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 однаковий, 3 - вплив трохи більший, 5 - вплив більший, 7 - вплив значно більший, 9 - вплив абсолютно більший (можливі проміжні оцінки: 2, 4, 6, 8).

Міру подібності за ступенем впливу між елементами  $x_i \in X$  і  $x_j \in X$  визначимо величиною

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

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  $d_{ij}$  можуть використовуватися відносні відстані по Хеммінгу ( $d_{ij}^{(h)}$ ) або по Евкліду ( $d_{ij}^{(e)}$ ):

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

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

Сукупність величин  $r_{ij}$  для всіх пар  $(x_i, x_j) \in X \times X$  утворює нечітке відношення подібності  $R \subset X \times X$ :

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

яке має властивості:

(А) рефлексивність, тобто  $r_{ij} = 1$ , для всіх  $x_i \in X$ ,

(Б) симетричність, тобто  $r_{ij} = r_{ji}$ , для всіх  $x_i \in X, x_j \in X$ .

Для розбиття множини  $X$  на непересічні класи елементів, подібних за ступенем впливу, необхідно надати вихідного нетранзитивному відношенню подібності  $R$  властивість транзитивності. Дане перетворення забезпечується операцією транзитивного замикання нечіткого відношення.

### **3.2. Приклад визначення коефіцієнтів впливу порушень цілісності баз даних на ефективність інформаційної системи**

Для кращого розуміння методу ранжування перед розробленням інформаційної технології пропонується розглянути приклад визначення коефіцієнтів важливості.

Нехай стоїть задача розподілу обмежених щоденних ресурсів на підтримку цілісності баз даних інформаційної системи керування залізничним транспортом. Структура обраної системи зображена на рисунку 3.2.1.

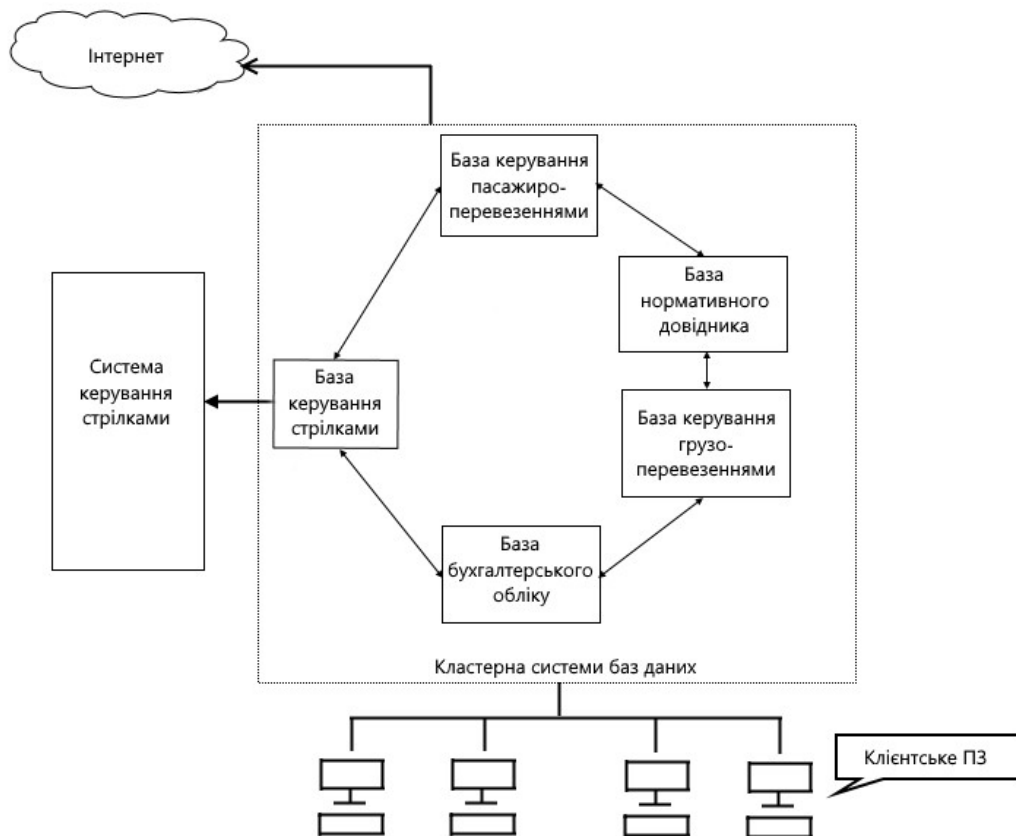


Рисунок 3.2.2 – Спроекована модель ІС залізничного транспорту

Таким чином розглядається інформаційна система, яка складається з 5 баз даних, тобто  $X = \{x_1, x_2, x_3, x_4, x_5\}$ . Експертна інформація, необхідна для обчислення відношення методом найменшого впливу, представлена у таблиці 3.2.1.

Таблиця 3.2.1 – Вихідні дані для методу найменшого впливу

$x_i$	$x_l$	ij/ il				
		$x_1$	$x_2$	$x_3$	$x_4$	$x_5$
$x_1$	$x_2$	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

Другий стовпець таблиці 3.2.1 ( $x_l$ ) містить елементи, на які відповідні елементи першого стовпчика ( $x_i$ ) мають найменший вплив:  $x_1$  найменш впливає на  $x_2$ ,  $x_2$  найменш впливає на  $x_2$ , ...,  $x_5$  найменш впливає на  $x_1$ . Джерелом цієї інформації являється експерт.

Користуючись даними з таблиці 3.2.1 і формулами, обчислимо ступені впливу  $\mu_{ij}$  для відношення. Тут  $\mu_{ii} = 0$ .

Для елемента  $x_1$  ( $i = 1, \ell = 2$ ) маємо:

$$\mu_{11} = 0$$

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

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

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

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

Аналогічно обчислюємо інші ступені приналежності, які утворюють нечітке відношення впливу:

Таблиця 3.2.2 – 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

Для нормалізації відношення розділимо елементи кожного рядка на максимальне значення і запишемо результати у таблицю 3.2.3.

Таблиця 3.2.3 – 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

Нечітке відношення подібності має вигляд таблиці 3.2.4.

Таблиця 3.2.4 – 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

Ступені приналежності із використанням відстані по Хеммінгу. Наприклад,  $r_{12} = 1 - d_{12}$ , де

$$d_{12} = \frac{1}{5} [(0,0,2,1,0,6,0,2) - (0,11,0,1,0,33,0,1)] =$$

$$\begin{aligned}
&= \frac{1}{5} [|0 - 0,11| + |0,2 - 0| + |1 - 1| + |0,6 - 0,33| + |0,2 - 0,1|] \\
&= \frac{1}{5} [0,11 + 0,1 + 0 + 0,27 + 0,09] = 0,13
\end{aligned}$$

Для отримання транзитивного замикання відношення подібності використовуємо нечітку композицію (таблиця 3.2.5):

Таблиця 3.2.5 – 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

У результаті бачимо, що транзитивне замикання в нашому випадку має вигляд:

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

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

тобто збігається з відношенням. Підсумовуючи значення рядків матриці, отримуємо кількісні значення рангів елементів:

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

$$p_2 = 4,06, p_3 = 3,12, p_4 = 4,02, p_5 = 4,02$$



Нечітке відношення (26) можна розкласти по  $\alpha$ -рівням наступним чином:

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

Для стислості елемент  $x_i$  позначається символом  $i$ ,  $i=1,2,\dots, 5$ . Чіткі відносини  $\alpha$ -рівня утворюють класи елементів, еквівалентних за важливістю (таблиця. 3.2.7).

Таблиця 3.2.6 – 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	$\{x_1, x_2\}, \{x_3\}, \{x_4\}, \{x_5\}$
$\alpha = 0,1$	5	$\{x_1\}, \{x_2\}, \{x_3\}, \{x_4\}, \{x_5\}$

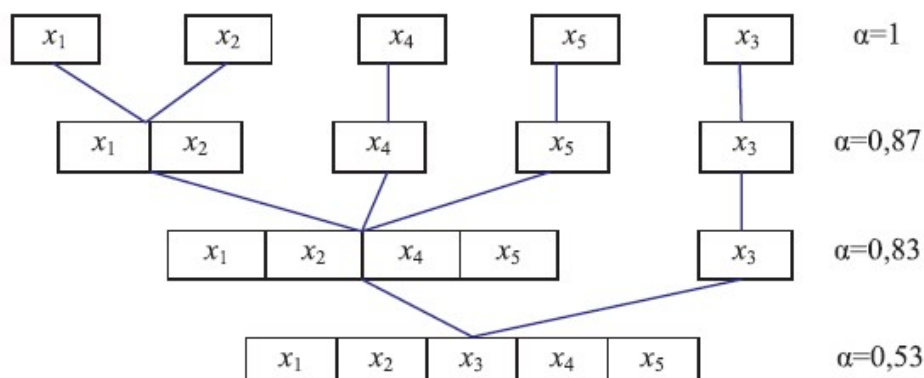


Рисунок 3.2.3 – Дерево декомпозиції на класи еквівалентності

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  $\alpha$ .

З рис. 3.2.3 видно, що при максимальній визначеності ( $\alpha = 1$ ) кожен з елементів  $x_i$  являє собою унікальний клас важливості. Однак на рівні  $\alpha = 0,53$  всі

елементи системи не помітні по рангах. З урахуванням кількісних оцінок для практичних розрахунків можна вибрати рівень визначеності  $\alpha = 0,83$ , на якому:

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

Якщо  $C_0$  - допустимі витрати на забезпечення цілісності системи, то з урахуванням рангів елементів ці витрати повинні розподілятися так:

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

### 3.3. Інформаційна технологія методу ранжування на основі нечіткого відношення подібності

Першим етапом створення інформаційної технології є узгодження функціональних вимог та дизайну. Функціональні вимоги наступні:

- визначення кількості баз даних у інформаційній системі, корпоративній мережі, тощо;
- виведення результатів методу ранжування у вигляді коефіцієнтів важливості, допустимих затрат та розподілених затрат в залежності від значення самого коефіцієнту;
- очищення даних для повторного обчислення коефіцієнтів;
- вибір допустимих затрат для забезпечення цілісності інформаційної системи;
- обробка усіх можливих виключних ситуацій введення некоректних вхідних даних;
- реалізувати залежність таблиці мінімальних впливів і матриці оцінок впливу за шкалою Сааті;
- розмістити текст допомоги використання на головному вікні;

Після узгодження функціональних та нефункціональних вимог, розробник проекту приступає до виконання прототипу головного вікна програмного забезпечення. Прототипом являється ескізне зображення вікна, яке виконане за допомогою MS Visio. Прототип вікна зображений на рис. 3.3.1.

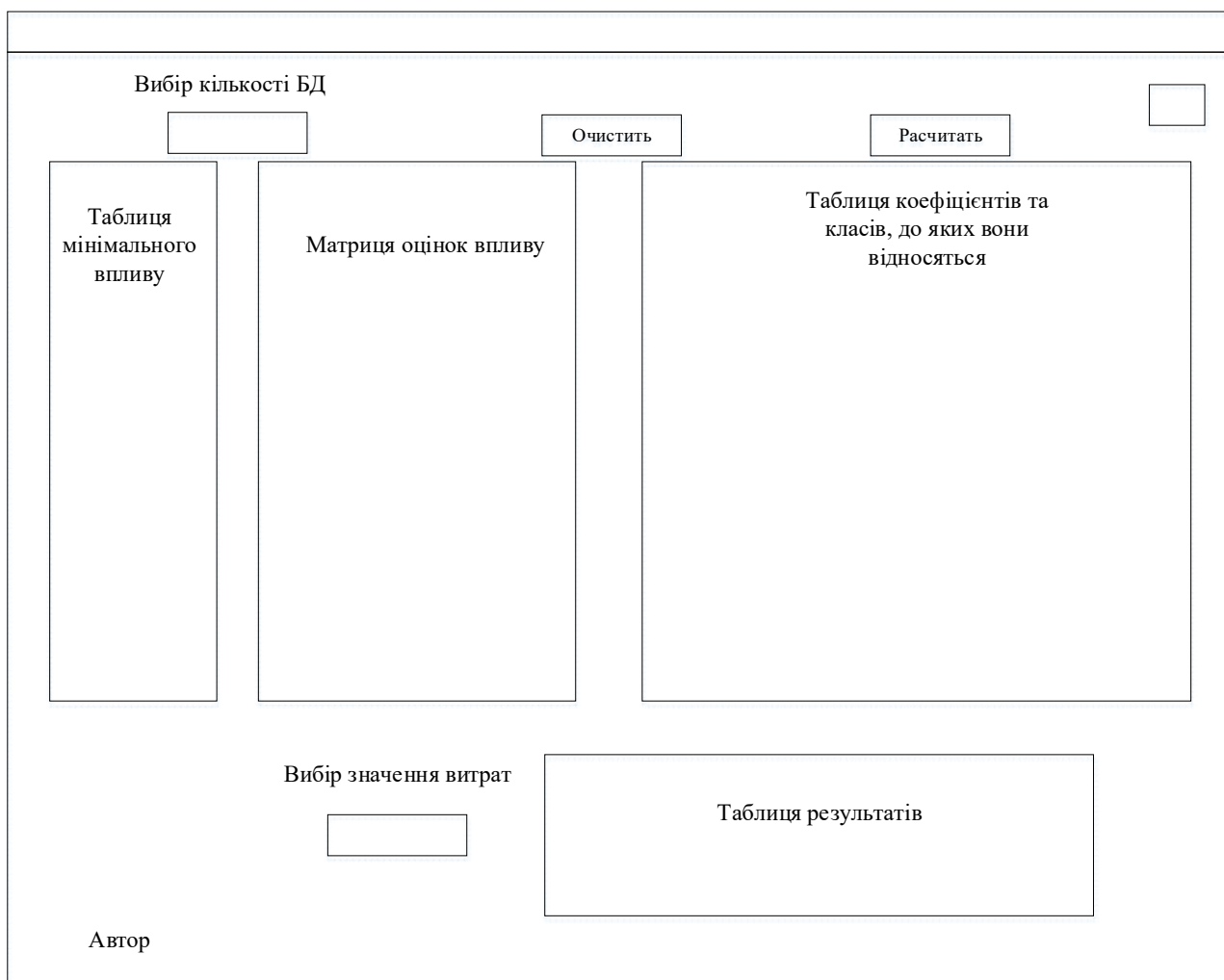


Рисунок 3.3.1 – Прототип інформаційної технології

Наступним кроком потрібно визначити, які саме елементи відповідають кожному компоненту ескізу вікна. Так як реалізація проводиться за допомогою мови програмування C#, обираємо лише елементи, які вже присутні у стандартній бібліотеці компонентів Windows Forms. Одним з головних елементів являється DataGridView – компонент, який дозволяє відображати дані у вигляді сітки, яку

можна модифікувати. Список усіх використаних елементів наведено у таблиці 3.3.1.

Таблиця 3.3.1 – Таблиця використаних елементів

№	Назва елемента	Опис елемента
1.	Label	Блок статичного тексту з прозорим фоном
2.	Button	Кнопка, яка виконує конкретні дії після натискання на неї
3.	NumericUpDown	Поле введення лише числових значень з допоміжним функціоналом у вигляді двох стрілок, які зменшують або збільшують значення на обраний крок
4.	DataGridView	компонент, який дозволяє відображати дані у вигляді сітки

Далі розробник з команди проекту починає будувати реальний прототип у програмному середовищі Visual Studio за допомогою раніше визначених компонентів. Реалізація проходила за допомогою створення проекту «додаток Windows Forms», і у конструкторі проекту розробник маніпулював елементами для здійснення очікуваного результату. Робочий прототип вікна зображений на рис. 3.3.2.

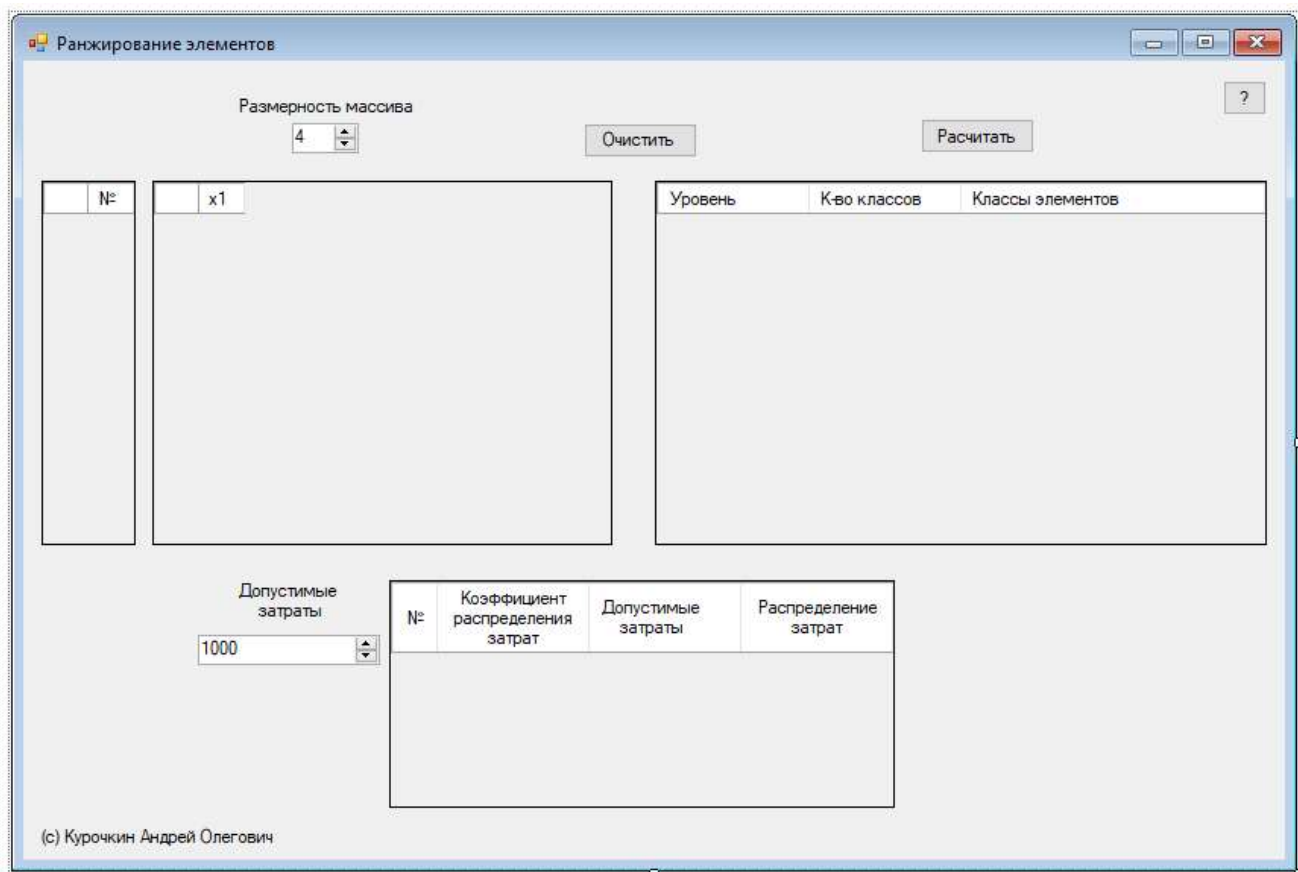


Рисунок 3.3.2 – Прототип вікна, створений у Visual Studio

Наступним етапом являється реалізація основних функцій із захистом від некоректного вводу та обробка виключних ситуацій обчислення згідно узгодженого алгоритму ранжування. Лістинг файлів коду реалізацій приведено у додатку С. Опис всіх створених модулів приведено у табл. 3.3.2.

Таблица 3.3.2. – Опис усіх використаних функцій

№	Назва функції/модулю	Опис
1.	Form1()	Відповідає за ініціалізацію компонентів та побудову стилів заголовків DataGridView

## Продовження таблиці 3.3.2.

2.	finances()	Визначення коефіцієнтів важливості баз даних інформаційної системи
3.	RoundTo2	Модифікація функції округлення десяткових чисел таким чином, щоб округлення виконувалося для всіх чисел окремого масиву чисел
4.	Array_check	Перевірка заповнення таблиць вхідних даних на коректне значення
5.	Find_Classes	Знаходження класів елементів в залежності від обчислень рангів баз даних
6.	Composition	Обчислення транзитивного замикання
7.	Normalization	Нормалізація матриці ступенів впливу баз даних
8.	fuzzy_sim	Обчислення ступенів приналежності баз даних за допомогою відстані Хеммінга
9.	value_size_ValueChanged	Відповідає за зміну розмірності таблиць вхідних значень при зміні числа кількості баз даних

## Продовження таблиці 3.3.2.

10.	datagridView_RowPostPaint	Виконує створення нового стилю для заголовку кожного рядка, щоб прибрати стрілку, яка зображується, коли обраний поточний рядок
11.	button1_Click	Виконує обчислення відповідно алгоритму ранжування
12.	button2_Click	Відображає текстове повідомлення з інформацією про використання ПЗ
13.	Min_i_CellContentClick	Реалізація залежності двох таблиць вхідних даних
14.	Button3_Click	Очищує дані з усіх таблиць
15.	numericUpDown1_ValueChanged	Змінює значення розподілених витрат при зміні значення допустимих витрат
16.	Button4_Click	Кнопка, яка відповідає адмін-панелі – заповнення таблиць вхідних чисел випадковими значеннями за шкалою Сааті

Створена інформаційна технологія відповідає усім поставленим вимогам та проведена серія комп'ютерних експериментів довела конструктивність методу. Далі наведено приклад роботи з програмою та повідомлення при некоректному введенні вхідних даних. Серія тестування наведена у додатку D.

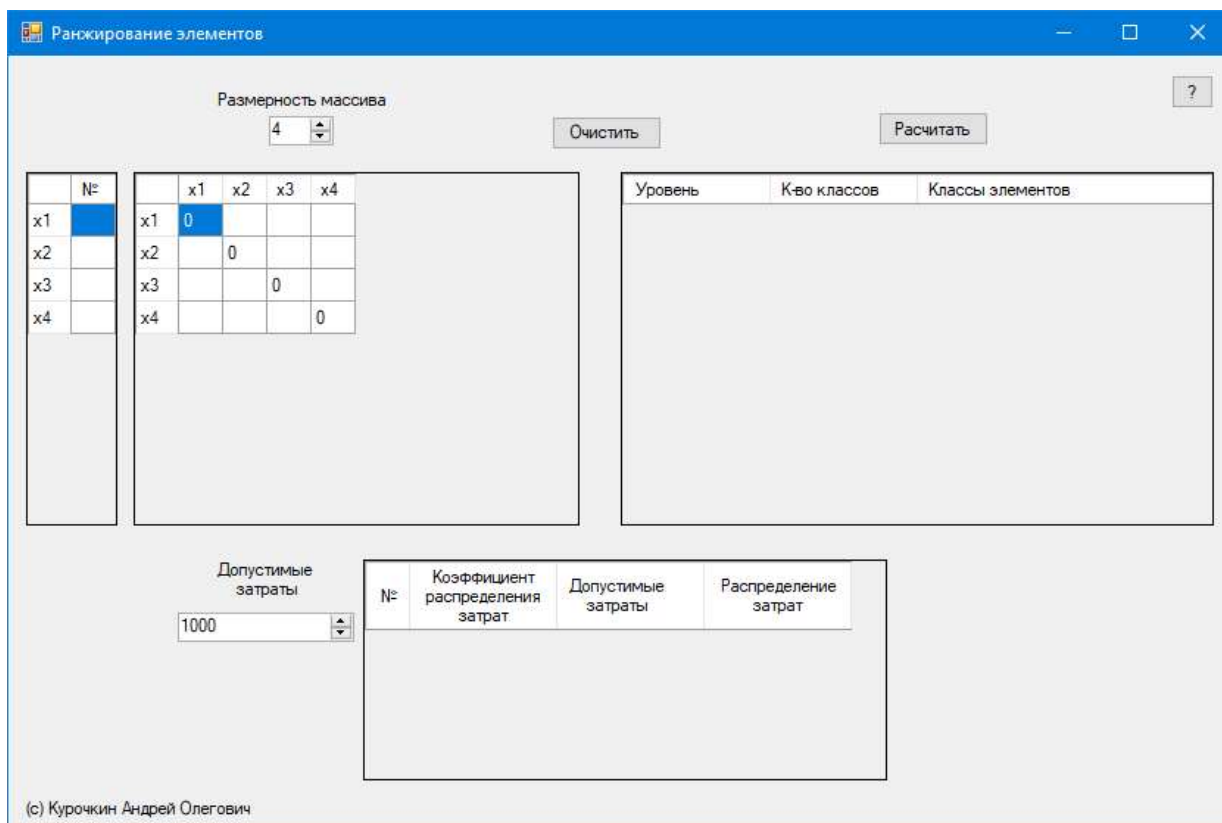


Рисунок 3.3.3 – Головне вікно інформаційної технології

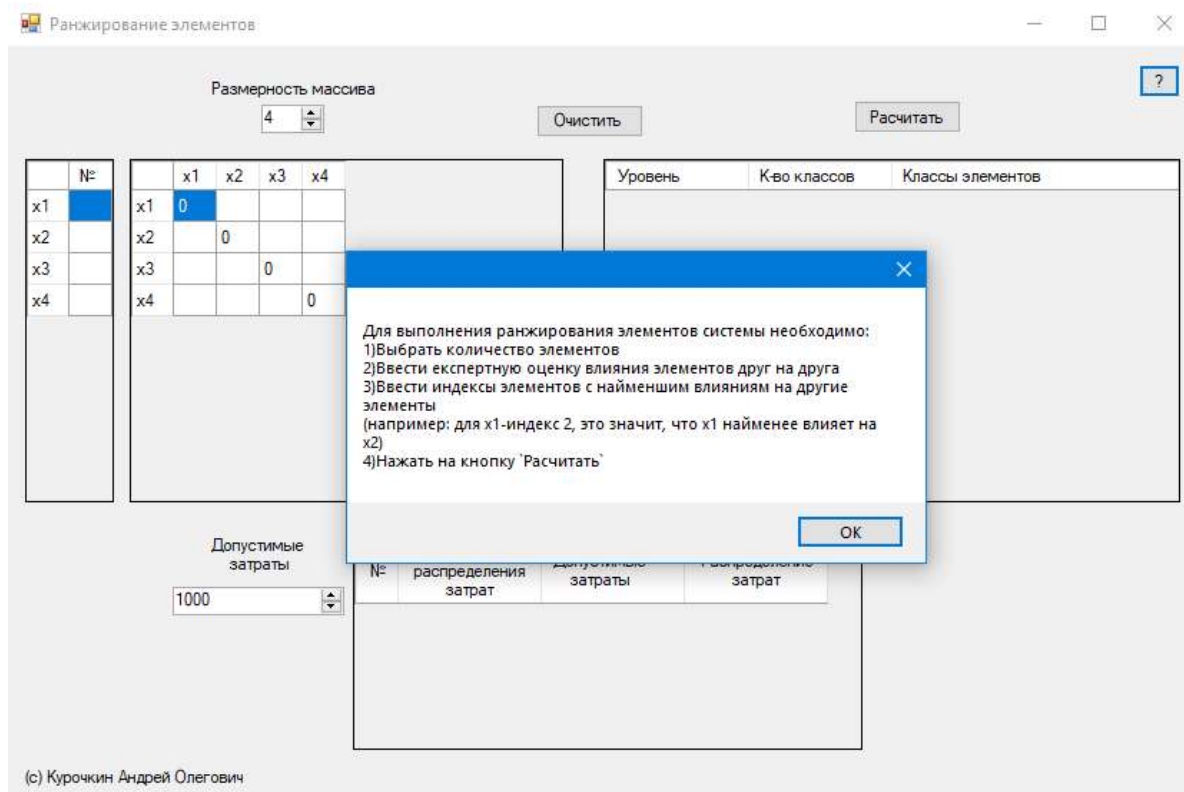


Рисунок 3.3.4 – Допоміжна інформація про користування ПЗ



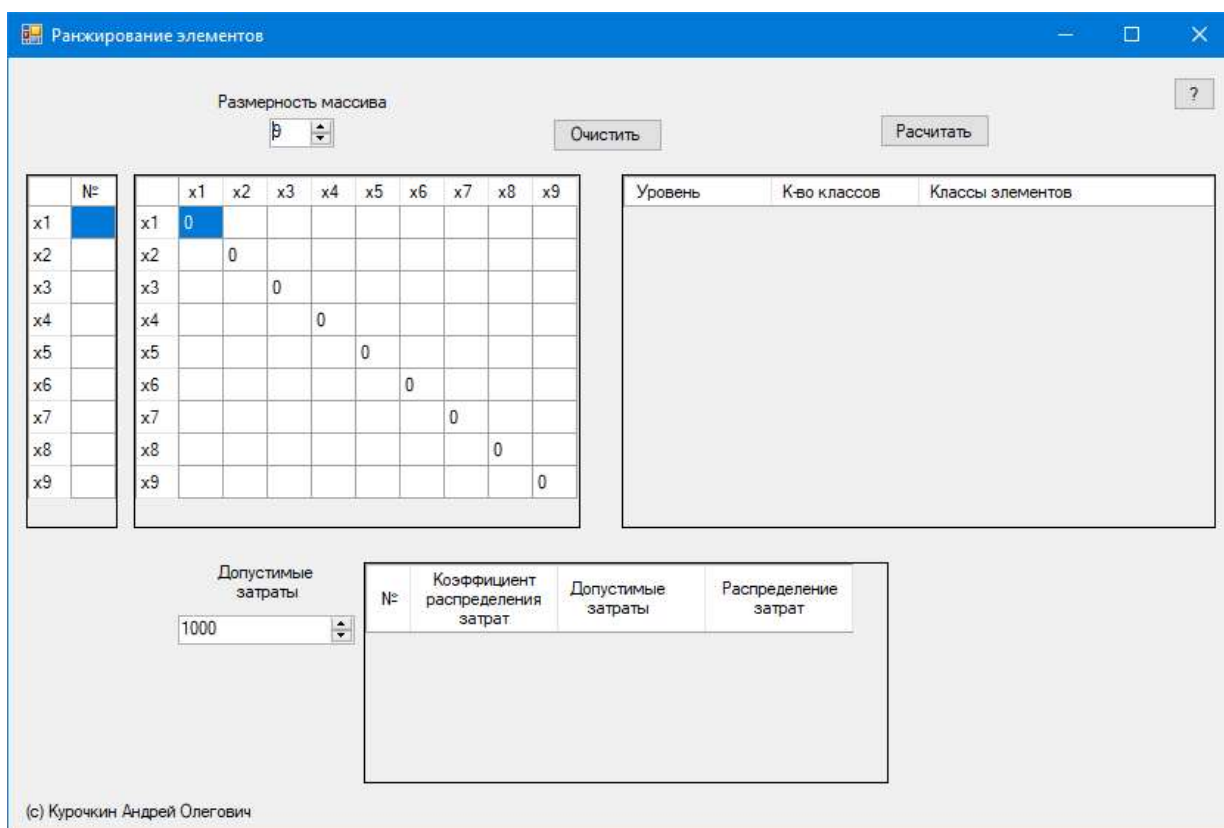


Рисунок 3.3.5 – Зміна кількості баз даних

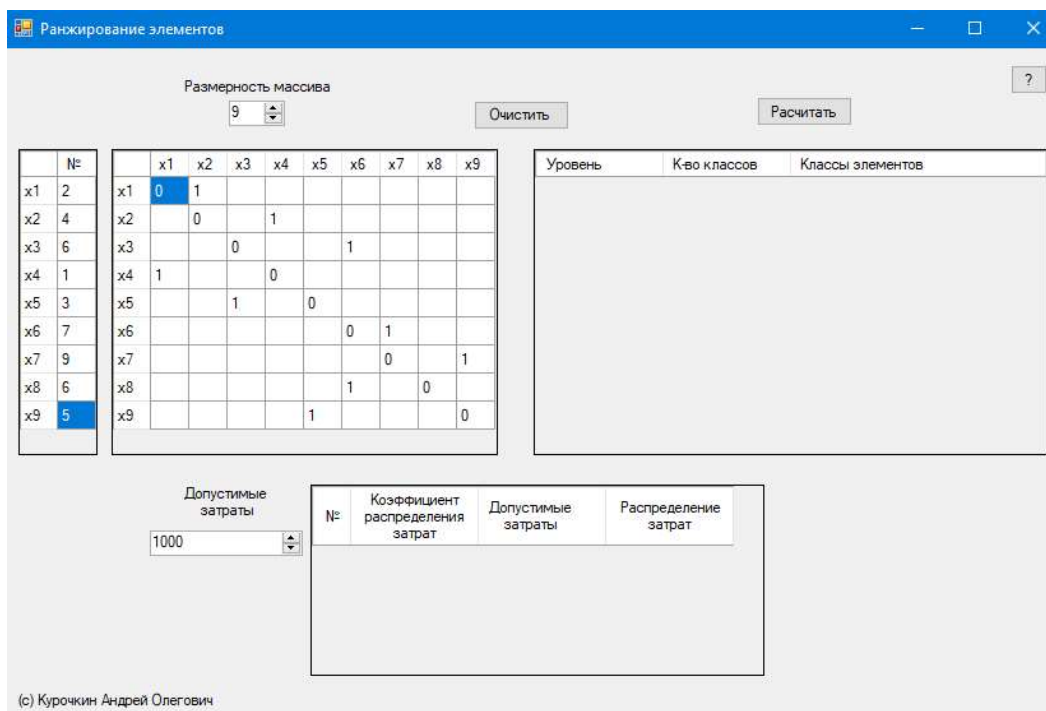


Рисунок 3.3.6 – Заповнена таблиця мінімального впливу

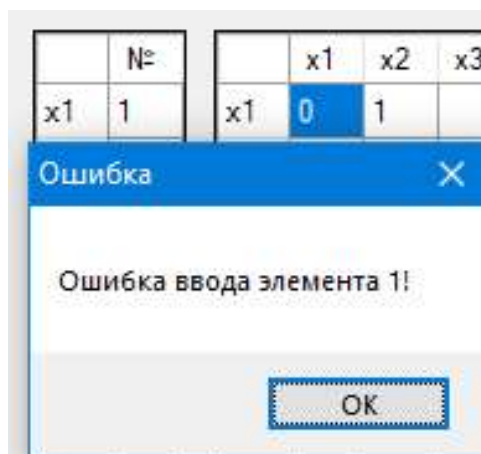


Рисунок 3.3.7 – Пример введения некорректного числа до таблицы минимального влияния

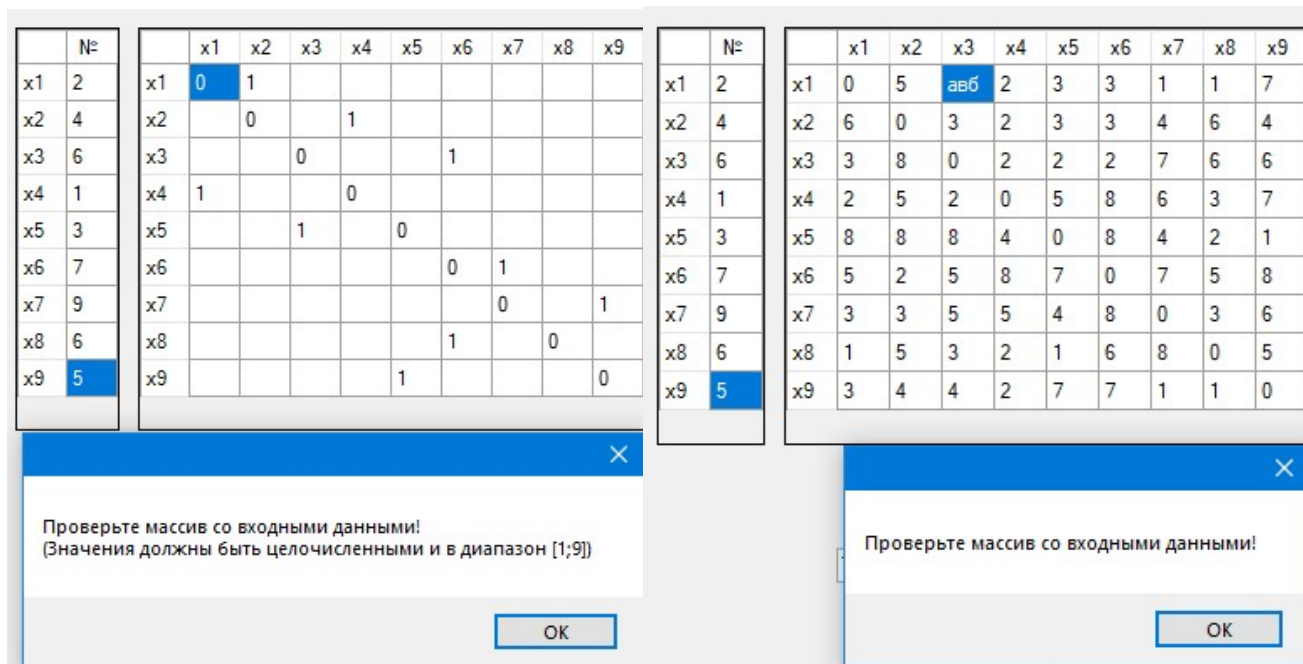


Рисунок 3.3.8 – Нажатие кнопки «Расчитать» при некорректных данных

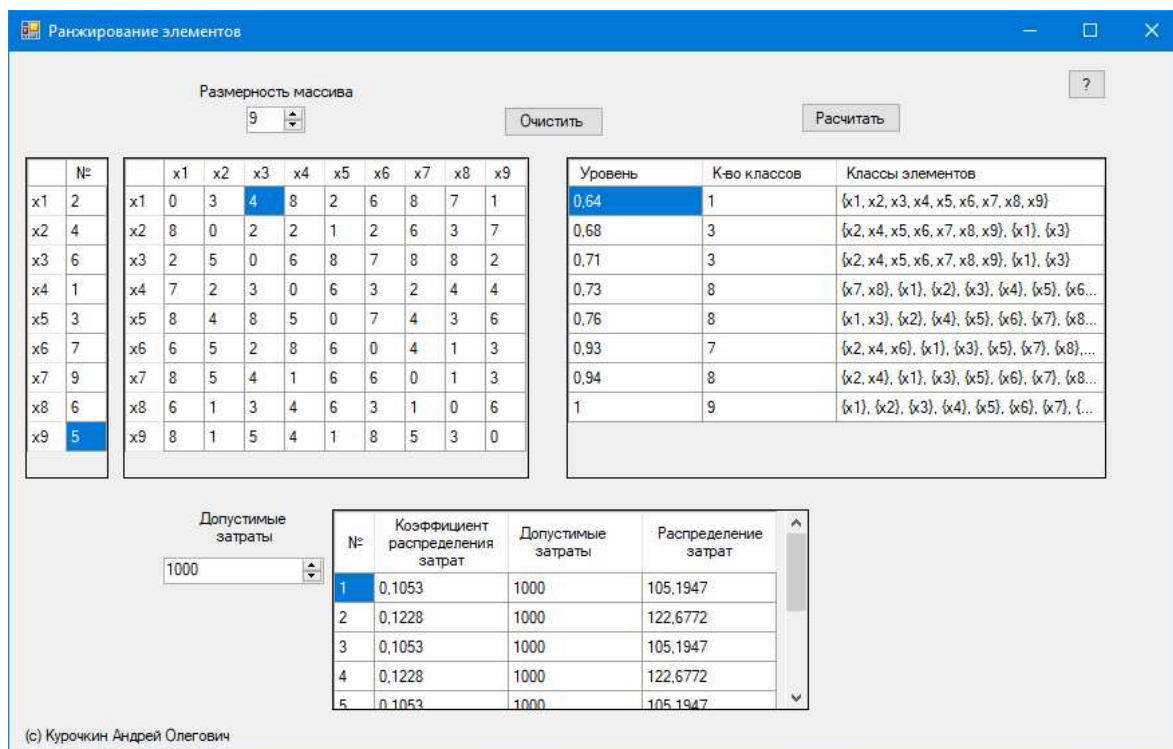


Рисунок 3.3.9 – Приклад обчислення розподіленних витрат

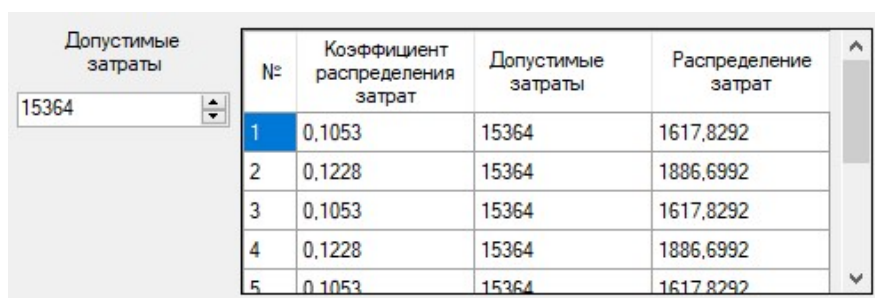


Рисунок 3.3.10 – Зміна допустимих витрат

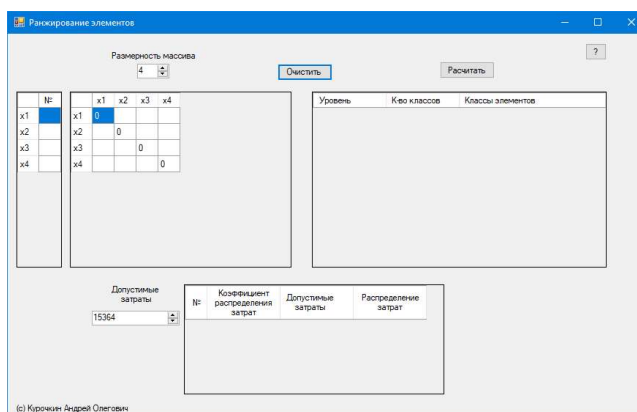


Рисунок 3.3.11 – Очищення даних

### 3.4. Розподіл обмежених ресурсів для вибору стратегій технічного обслуговування баз даних

Розв'язком задачі, що поставлено, є значення фінансових витрат на забезпечення цілісності конкретної бази даних. Якщо  $V_0$  - допустимі витрати на забезпечення цілісності системи, то з урахуванням рангів елементів ці витрати повинні розподілятися наступним чином:

$$V_i = \lambda V_0,$$

де ( $i = 1, n$ ) – кількість баз даних;

$\lambda$  – коефіцієнт розподілення ресурсів.

Для попереднього прикладу:

$$\sum_{i=1}^5 V_i = V_0, \quad V_1 = V_2 = V_4 = V_5 = 0,21V_0, \quad V_3 = 0,15V_0$$

Дані витрати визначають параметри:

$k_1$  – кількість основного і допоміжного персоналу;

$t_v$  – технологію виявлення порушень;

$t_y$  – технологію усунення наслідків НВ, створеного ДНВ;

Параметри  $k_1$ ,  $t_v$ ,  $t_y$  фактично визначають характеристики моделей ІС (рис 2.2.1). Це дозволяє вирішити задачу розподілу обмеженого ресурсу на забезпечення цілісності баз даних в умовах конфліктних взаємодій.

Для вирішення завдання вибору оптимальних стратегій процес усунення негативного впливу на базу даних представимо у вигляді напівмарковських процесу (НМП) – рисунок 3.4.1.

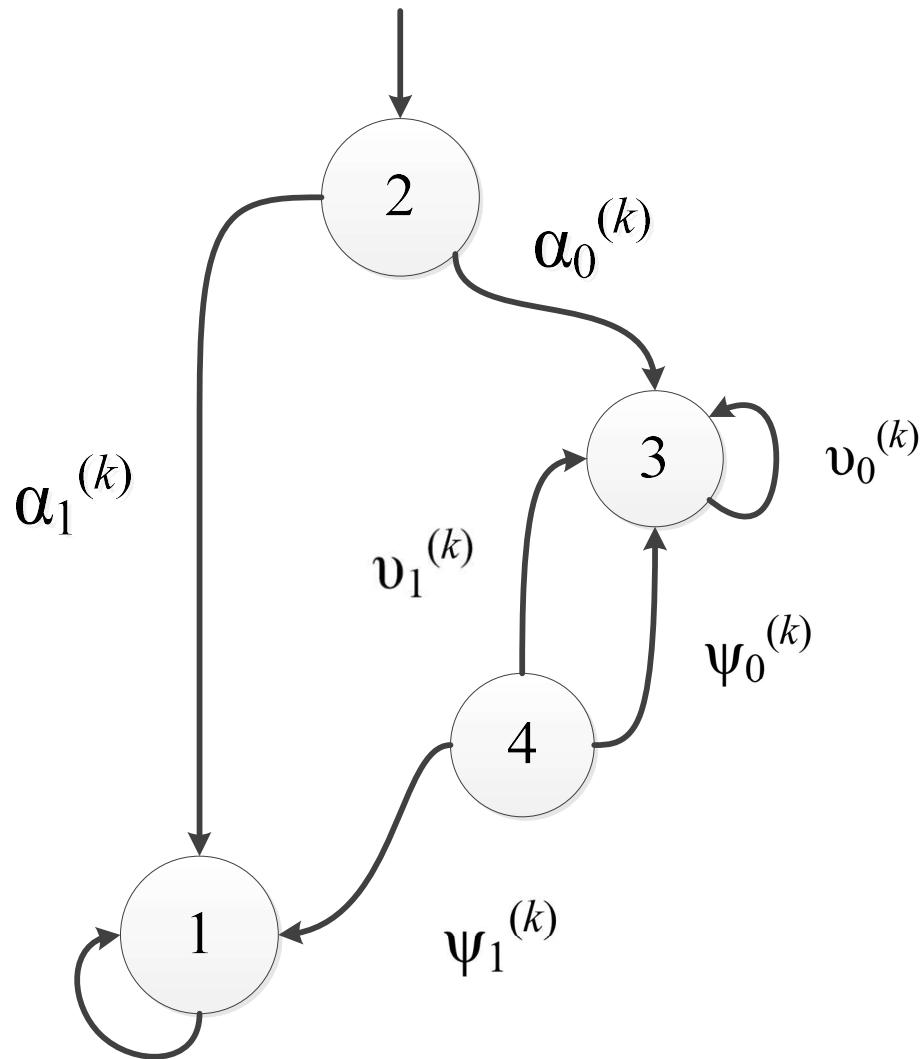


Рис. 3.4.1. - Граф подій процесу усунення негативного впливу на базу даних

Де  $S_1$  - працездатний стан

$S_2$  - стан, який відповідає початку негативного впливу;

$S_3$  - стан порушення в базі даних і невиявлений фактом порушення;

$S_4$  - стан порушення в базі даних і виявлений фактом порушення;

У даному документі введено такі показники якості виконання операцій по усуненню негативного впливу

$\alpha_1^{(k)}$  – ймовірність автоматичного усунення негативного впливу при застосуванні  $k$ -ї стратегії,  $k \in K_a$ ,  $K_a$  – множина стратегій для автоматичного усунення негативного впливу;

$\alpha_0^{(k)}$  – ймовірність порушення ( $\alpha_0^{(k)}=1-\alpha_1^{(k)}$ ) при застосуванні  $k$ -ї стратегії,  $k \in K_a$ ;

$\nu_1^{(k)}$  – ймовірність виявлення порушення ДНВ при застосуванні  $k$ -ї стратегії  $k \in K_p$

$K_p$  – безліч стратегій виявлення порушення системним адміністратором;

$\nu_0^{(k)}$  – ймовірність не виявлення порушення системним адміністратором;

$\psi_1^{(k)}$  – ймовірність безпомилкового усунення порушення системним адміністратором при застосуванні  $k$ -ї стратегії  $k \in K_p$ ,

$K_p$  – множина стратегій виявлення порушення системним адміністратором;

$\psi_0^{(k)}$  – ймовірність усунення порушення вендором з помилкою  $k$ -ї стратегії  $k \in K_p$ .

Матриця НМП в позначеннях показників якості виконання операцій задачі оптимізації прийняття рішення про організацію процесу усунення негативного впливу на базу даних має вигляд:

$$[P_{ij}^k] = \begin{pmatrix} 1 & 0 & 0 & 0 \\ \alpha_1^{(k)} & 0 & \alpha_0^{(k)} & 0 \\ 0 & 0 & \nu_0^{(k)} & \nu_1^{(k)} \\ \psi_1^{(k)} & 0 & \psi_0^{(k)} & 0 \end{pmatrix}$$

Вектор початкових ймовірностей:

$$a = (a_2, a_3, a_4) = (1, 0, 0);$$

$K$  – безліч всіх стратегій для усунення негативного впливу на базу даних;

$$K = K_a \cup K_p, \text{ де}$$

$K_a$  – безліч стратегій для автоматичного усунення негативного впливу на базу даних;

$K_p$  – безліч стратегій виявлення порушення людиною;

Кожному варіанту закінчення функціонування процесу усунення негативного впливу на базу даних на графі подій ставимо у відповідність

поглинаючий стан, наприклад, «працевдатний стан». Поглинаючу вершину нумеруємо першим  $s$  натуральним числом ( $s=1$ ). Для початкових вершин, які нумеруються числами з числової послідовності після першої  $s$  поглинає вершини, задаємо вектор початкових ймовірностей:

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

де  $N$  – загальна кількість вершин графа, з яких перша  $s$ - поглинаюча.

У кожній вершині  $i$  може бути  $K_i$  альтернатив або рішень,  $K_i \in K$ . Кожному рішенню відповідає свій набір переходів, який характеризується ймовірністю переходу з вершини  $i$  в вершину  $j$  при виборі  $k$ -го рішення,  $k \in K_i$ .

В умовах задачі прийняття рішення про організацію усунення негативного впливу на базу даних  $k$ -е рішення означає застосування стратегії  $k \in K_i \in K$  на етап технологічного процесу, який моделюється вершиною  $i$  графа подій.

$P_{ij}^{(k)}$  – ймовірність переходу системи з вершини  $i$  в вершину  $j$  при виборі  $k$ -го рішення. Причому:

$$\sum_j P_{ij}^{(k)} = 1 \quad \text{при всіх } i \text{ и при всіх } k \in K_i.$$

$T_{ij}^{(k)}$  – середній час  $i$ -ї роботи при  $k$ -му рішенні при переході в вершину  $j$ . В цьому випадку середнє очікуване час  $i$ -ї роботи при  $k$ -й стратегії  $T_i^{(k)}$  обчислюються за наступною формулою:

$$T_i^{(k)} = \sum_j P_{ij}^{(k)} T_{ij}^{(k)}$$

$r_{ij}^{(k)}$  – середня вартість  $i$ -ї роботи по усуненню негативного впливу на базу даних при  $k$ -й стратегії при переході в вершину  $j$ . У цьому випадку середня очікувана вартість  $i$ -ї роботи при  $k$ -й стратегії обчислюються за наступною формулою:

$$r_i^{(k)} = \sum_j P_{ij}^{(k)} r_{ij}^{(k)}$$

Під оптимізацією системи розуміється вибір в кожній вершині такого рішення (альтернативи), щоб цільовій функції доставлявся екстремум.

В якості цільової функції беремо  $T$  - середній час процесу усунення негативного впливу на базу даних до поглинання (вершина  $s=I$  на графі подій):

$$T = \sum_{i=2}^N \sum_{k \in K_i} P_{i1}^{(k)} T_{i1}^{(k)} x_i^{(k)}$$

У формулі  $x_i^{(k)}$  – змінна, яка характеризує вибір рішення:

$x_i^{(k)} > 0$  у тому випадку, якщо в  $i$ -й вершині для виконання роботи по усуненню негативного впливу на базу даних обрано  $k$ -е рішення та  $x_i^{(k)} = 0$ , у протилежному випадку.

У задачі оптимізації враховуємо той факт, що середня вартість роботи по усуненню негативного впливу на базу даних  $V$  не повинна перевищувати заданого значення  $V_0$ .

$$V = \sum_{i=2}^N \sum_{k \in K_i} P_{i1}^{(k)} r_{i1}^{(k)}$$

Для обліку обмеження за вартістю, вводяться булеві змінні  $\delta_i^{(k)}$  ( для застосування на етапі технологічного процесу, який моделюється вершиною  $i$  графа подій,  $k$ -ї стратегії) і додаються нові обмеження.

Результатами оптимізації є:

Значення цільової функції  $T_i$  – значення середньої вартості роботи по усуненню негативного впливу на базу даних  $V$ ;

Номер стратегії в кожній вершині, який доставляє екстремум цільової функції.



Таким чином, завдання оптимізації прийняття рішення про організацію процесу усунення негативного впливу на базу даних з урахуванням обмеження на вартість виконання операцій  $V$ , має вигляд:

$$T = \sum_{i=2}^N \sum_{k \in K_i} P_{i1}^{(k)} T_{i1}^{(k)} x_i^{(k)} \rightarrow \min \quad (19)$$

$$\sum_{i=2}^N \sum_{k \in K_i} P_{i1}^{(k)} r_{i1}^{(k)} x_i^{(k)} \leq V_0 \quad (20)$$

$$\sum_{k \in K_i} x_j^{(k)} - \sum_{i=r+1}^N \sum_{k \in K_i} P_{ij}^{(k)} x_i^{(k)} = a_j, j = 2, 3, \dots, N \quad (21)$$

$$\sum_{i=2}^N \sum_{k \in K_i} P_{i1}^{(k)} x_i^{(k)} = 1 \quad (22)$$

$$x_i^{(k)} \geq 0 \text{ для всіх } i \text{ та } k \in K_i \quad (23)$$

$$\sum_{k \in K_i} \delta_i^{(k)} = 1, \text{ для всіх } i \text{ та } k \in K_i \quad (24)$$

$$x_i^{(k)} - M * \delta_i^{(k)} \leq 0 \text{ для всіх } i \text{ та } k \in K_i \quad (25)$$

$$x_i^{(k)} - m * \delta_i^{(k)} \geq 0 \text{ для всіх } i \text{ та } k \in K_i \quad (26)$$

$$\delta_i^{(k)} \in \{0,1\} \text{ для всіх } i \text{ та } k \in K_i \quad (27)$$

де  $m$  і  $M$  - досить мале і досить велике число.

Рішення даної задачі лінійного програмування (19) - (27) має таку властивість, що для кожного  $i$  лише один  $x_i^{(k)}$  відмінний від нуля. Це означає, що в вершині  $i$  для виконання роботи приймається  $k$ -е рішення,  $k \in K_i$ , тобто застосовується стратегія  $k \in K_i \in K$ .

Сенс зазначених обмежень наступний:

Обмеження (22) є нормованою умовою, що вимагає, щоб з ймовірністю 1 процес поглинувся.

Обмеження (24) для булевої змінної  $\delta_i^{(k)}$  вимагає, щоб для кожного  $i$  лише одне  $\delta_i^{(k)}$  було рівне одиниці. Це означає, що в кожній вершині  $i$  для виконання роботи приймається тільки одне  $k$ -е рішення,  $k \in K_i$ ;

Обмеження (25) вимагає, щоб при кожному  $i$  не більше ніж одне  $x_i^{(k)}$  було відмінно від нуля (спільно з обмеженням (5));

Обмеження (27) вимагає, щоб при кожному  $i$  одне або більше  $x_i^{(k)}$  було відмінно від нуля. Спільно з цим, обмеження (25) і (26) вимагають, щоб лише одне  $x_i^{(k)}$  було відмінне від нуля.

Зв'язок між кожним із завдань оптимізації здійснений через обмеження типу (20), в якому  $V_0$  – розв'язок задачі ранжування важливості баз даних.

## ВИСНОВКИ

Останнім часом збільшується кількість кібератак та негативних впливів на інформаційні системи. На жаль, не дивлячись на кількість наукових досліджень, задача оптимізації організаційно-технічних заходів з питань забезпечення цілісності баз даних вирішена не до кінця.

Розподіл обмежених ресурсів на заходи по забезпеченню цілісності між окремими елементами системи вимагає чисельних значень, які характеризують важливість цих елементів. Запропонована математична модель і інформаційна технологія ранжування, засновану на методі нечіткого відношення подібності та транзитивного замикання.

Для моделювання процесу виникнення та усунення негативних впливів на бази даних інформаційної системи зручно використовувати апарат напівмарківських процесів.

Побудована напівмарківська модель процесу прийняття рішення про вибір заходів по забезпеченню цілісності баз даних дозволяє обирати стратегії виявлення атак і помилок, а також усунення їх наслідків.

Побудована відповідна модель зведена до задачі лінійного програмування, що є передумовою вирішення її в будь-яких програмних середовищах.

На відміну від існуючих методів забезпечення цілісності баз даних інформаційної системи, орієнтованих, як правило, на використання інтуїтивних підходів, розроблений метод використовує моделі виникнення і усунення пошкоджень, побудовану з використанням апарату напівмарківських процесів, і технологію ранжування баз даних.

Результати можуть бути покладені в основу системи забезпечення цілісності баз даних для підприємств з розподіленим інформаційним забезпеченням.

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Уткин, Л. В. Методы и модели анализа надежности и безопасности информационных систем при неполной информации: автореф. дис. на соиск. учен. степ. докт. техн. наук : 05.13.18 / Уткин Лев Владимирович. : С.Пб. : 2001. 305 с.
2. Daintith, J. "IT" / J. Daintith // A Dictionary of Physics. : Oxford University Press, 2012. 592 p.
3. Ethical Hacking and Countermeasures: Attack Phases / M. Bellegarde, M. Orvis, S. Helba. – EC-Council Press, 2010. 384 p.
4. Скембрей, Дж. Секреты хакеров. Безопасность сетей – готовые решения / Дж. Скембрей, Ст. Мак-Клар, Дж. Курц. Москва : Вильямс: 2001. 656 с.
5. Шелухин, О. И. Обнаружение вторжений в компьютерные сети [сетевые аномалии] / О. И. Шелухин, Д. Ж. Сакалема, А. С. Филинова. М.: Горячая линия – Телеком, 2013. 220 с.
6. Alhazmi, O.H. Modeling the Vulnerability Discovery Process / O.H. Alhazmi, Y.K. Malaiya. // Proc. Int. Symp. Software Reliability Eng, Nov. 2005, pp. 129-138.
7. ГОСТ 28195-89 Оценка качества программных средств. : Изд-во стандартов, 1989. 38 с.
8. ГОСТ Р ИСО/МЭК 9126-93 Оценка программной продукции. Характеристики качества и руководства по их применению. : Изд-во стандартов, 1994. 15 с.
9. A Complete Guide to the Common Vulnerability Scoring System Version 2.0: [Электронный ресурс]. URL: <http://www.first.org/cvss/cvssguide>.

10. ГОСТ 27.002-89 Надежность в технике. Основные понятия. Термины и определения. : Издательство стандартов, 1990. 37 с.
11. The Laws of Vulnerabilities: Six Axioms for Understanding Risk : [Электронный ресурс]. URL: [http://www.qualys.com/docs/Laws\\_2.0.pdf](http://www.qualys.com/docs/Laws_2.0.pdf).
12. Microsoft Security Intelligence Report v14 : [Электронный ресурс]. URL: <http://www.microsoft.com/security/sir/>.
13. Microsoft Security Intelligence Report v15 : [Электронный ресурс]. URL: <http://www.microsoft.com/security/sir/>.
14. Internet Security Threat Report 2011 Trends : [Электронный ресурс]. URL: [http://owasp.com/images/7/70/Symantec\\_ISTR\\_17.pdf](http://owasp.com/images/7/70/Symantec_ISTR_17.pdf).
15. Rotshtein A., Shnaider E., Schneider M. and Kandel A. Fuzzy multicriterial selection of alternatives : The worst-case method, International journal of intelligent systems: 01/2010 pp. 948-957
16. Ротштейн А.П. Ранжирование элементов системы на основе нечетких отношений: метод наименьшего влияния : [Электронный ресурс]. URL: <https://www.dependability.ru/jour/article/view/100/273>
17. Why Is C# Among The Most Popular Programming Languages in The World? : [Электронный ресурс]. URL: <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 : [Электронный ресурс]. URL: <https://bit.ly/2wFPzbA>
19. Курочкин А.О. Оптимізація процесів забезпечення надійності баз даних розподілених інформаційних систем в умовах негативного зовнішнього впливу : [Электронный ресурс]. URL: <https://bit.ly/2ItylNg>
20. Саати Т. Принятие решений. Метод анализа иерархий. Москва: Радио и связь, 1993. 278 с.

21. Назаров, Д. М. Интеллектуальные системы: основы теории нечетких множеств: учебный пособие для академического бакалавриата / Д. М. Назаров, Л. К. Конишева. : Вид. 3-е . Москва : Издательство 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. Грабовецкий Б. Є. Методи експертних оцінок: теорія, методологія, напрямки використання: монографія. Вінниця : ВНТУ, 2010. 171 с.
24. Гнатієнко Г. М., Снітюк 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: [Электронный ресурс]. URL: <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.* [Электронный ресурс]. URL: <https://bit.ly/2ImGTMX>

## **ДОДАТОК А. ТЕХНІЧНЕ ЗАВДАННЯ**

**ТЕХНІЧНЕ ЗАВДАННЯ**  
**на розробку моделей і інформаційної технології**  
**«Моделі вибору технологій забезпечення цілісності баз даних інформаційної**  
**системи в умовах конфліктних взаємодій»**

**Суми 2019**

## **Формування вимог до моделей і інформаційної забезпечення цілісності баз даних інформаційної системи в умовах конфліктних взаємодій**

**1 Найменування:** Моделі і інформаційна технологія забезпечення цілісності баз даних інформаційної системи в умовах конфліктних взаємодій

**2 Терміни виконання:** 3 червня 2019 року.

**3 Призначення:** розроблені моделі і інформаційна технологія для забезпечення цілісності баз даних інформаційної системи в умовах конфліктної взаємодії, повинні оптимізувати процес виявлення і усунення негативного впливу.

**4 Мета:** розробити моделі і інформаційну технологію вибору заходів забезпечення цілісності баз даних інформаційної системи в умовах конфліктних взаємодій.

**5 Основні завдання:** задача визначення коефіцієнтів важливості баз даних інформаційної системи та оптимізація процесу виявлення і усунення наслідків порушень.

**6 Вхідні дані:** експерта оцінка взаємовпливу баз даних

**7 Вихідні дані:** коефіцієнти важливості баз даних, значення середньої вартості усунення негативного впливу, список організаційно-технічних заходів

**8 Програмне забезпечення:** Microsoft Excel 2016, Visual Studio 2015

**9 Апаратне забезпечення:** склад апаратного забезпечення повинен забезпечувати роботу програмного забезпечення, зазначеного у п. 8.

**10 Рівень кваліфікації:** користувач повинен мати навички роботи з ПК.



## ДОДАТОК Б. ПЛАНУВАННЯ РОБІТ

**Деталізація мети методом SMART.** Продуктом дипломного проекту є моделі функціонування інформаційної системи в умовах конфліктної взаємодії. Результати деталізації методом SMART розміщені у табл. Б.1.

Таблиця Б.1 – Деталізація мети методом SMART

Specific (конкретна)	Створити моделі функціонування інформаційної системи в умовах конфліктної взаємодії
Measurable (вимірювана)	Результатом роботи проекту є програмне забезпечення оптимального розподілу фінансових ресурсів
Achievable (досяжна)	Реалізація моделей та інформаційної технології здійснюється за допомогою Microsoft Excel та програмного середовища Visual Studio
Relevant (реалістична)	У наявності необхідні навички та програмне забезпечення
Time-framed (обмежена у часі)	Робота повинна бути виконана у час, обговорений замовником проекту.

**Планування змісту структури робіт.** Структурна декомпозиція робіт (work breakdown structure, WBS) - структурна схема, що розкладає роботи за рівнем відповідно до внутрішньої структури або процесу реалізації. Вона розбиває проект на декілька відносно незалежних робочих одиниць. Велика структура розбиття на роботу призведе до кращої продуктивності команди та комплексного управління проектами. Схема структури робіт зображена на рис Б.1.

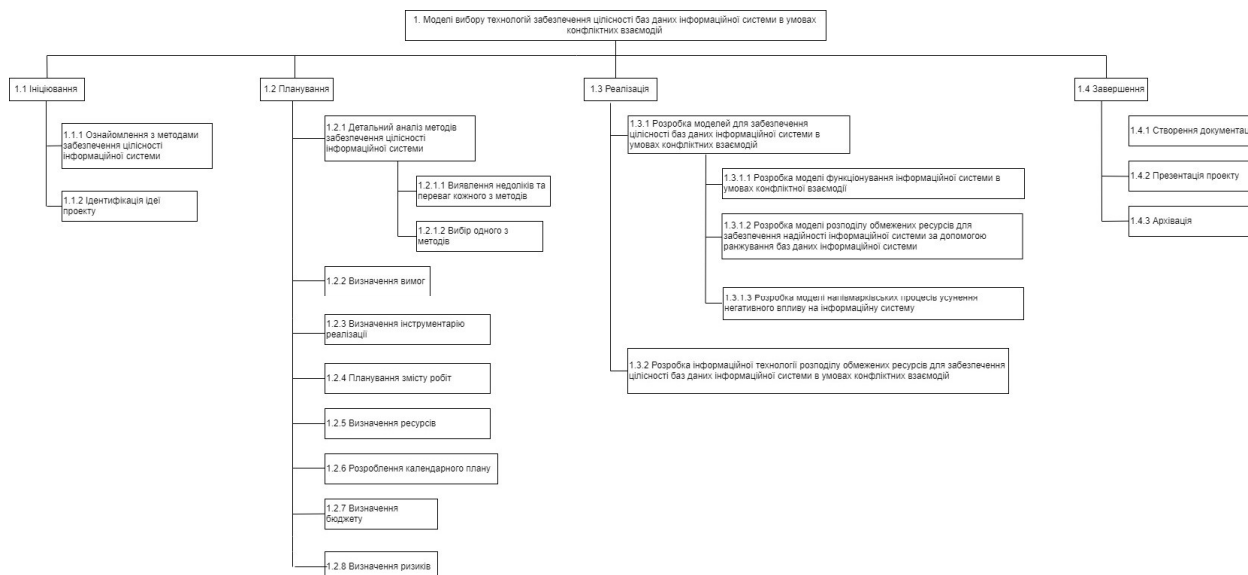


Рисунок Б.1 – WBS структура проекту

**Планування структури організації, для впровадження готового проекту (OBS).** Наступним плануванням структури робіт є розроблення організаційної структури проекту, яка зображує відповідність керівника та виконавця до робіт проекту. Структура організації проекту зображена на рис. Б.2.

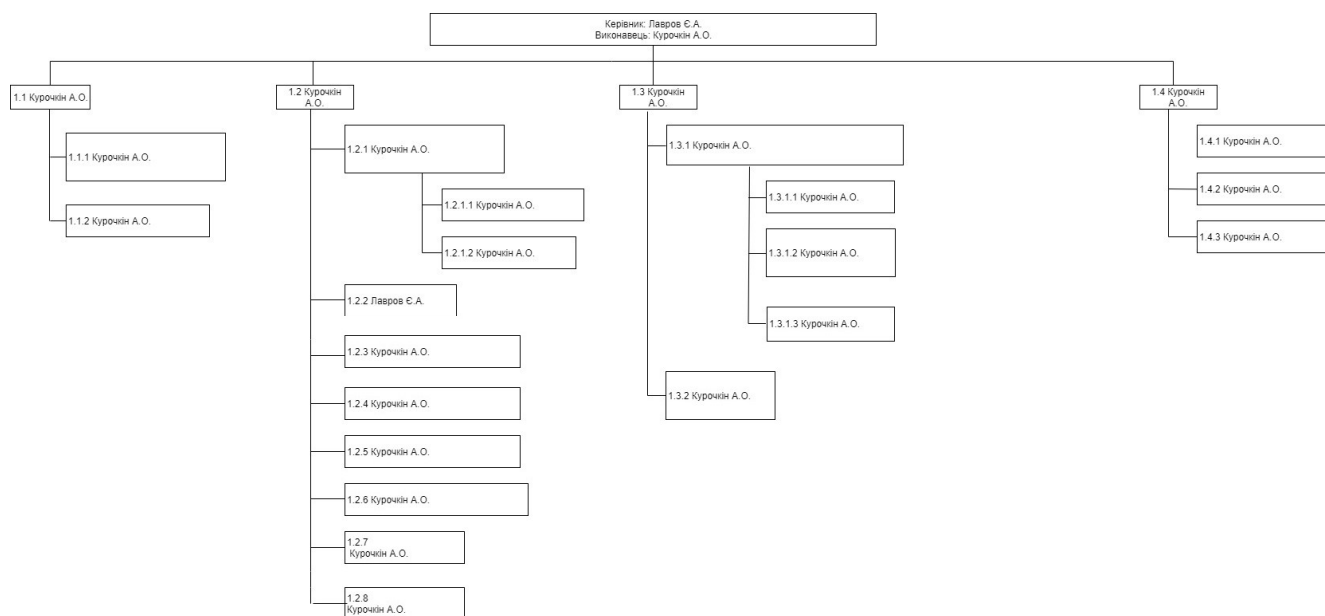


Рисунок Б.2 – OBS структура

**Побудова матриці відповідальності.** Матриця відповідальності (Responsibility Assignment Matrix) – матриця, яка зображує опис структури

відповідальності персон за виконання робіт проекту. Список виконавців та їх відповідальність зображена у таблиці Б.2

Таблиця Б.2 – Матриця відповідальності

№/ППБ	Курочкін А. О.	Лавров Є.А.
1.1.1		
1.1.2		
1.2.1.1		
1.2.1.2		
1.2.3		
1.2.4		
1.2.5		
1.2.6		
1.2.7		
1.2.8		
1.3.1.1		
1.3.1.2		
1.3.1.3		
1.3.2		
1.4.1		
1.4.2		
1.4.3		

**Діаграма Ганта.** Діаграма Ганта - графічне зображення графіка проекту. Це тип гістограми, що показує дати початку та закінчення кількох елементів проекту, які включають ресурси, етапи, завдання та залежності. Американський інженер-механік Генрі Гант розробив діаграму Ганта.

На графіку завдання відображаються на вертикальній осі, а запланований час - на горизонтальній осі. Кожне завдання представлено смугою, що показує час, необхідний для виконання всього пакету робіт. Графік також показує залежності, які просто означають взаємозв'язок між різними видами діяльності в проекті. Розуміння взаємозв'язку між діяльністю дуже важливо для моніторингу, а діаграми Ганта допомагають у цьому. Для побудови діаграми Ганта використовувалось ПЗ MS Office Project 2016 року. Діаграма Ганта для проекту показана на рисунку Б.3.

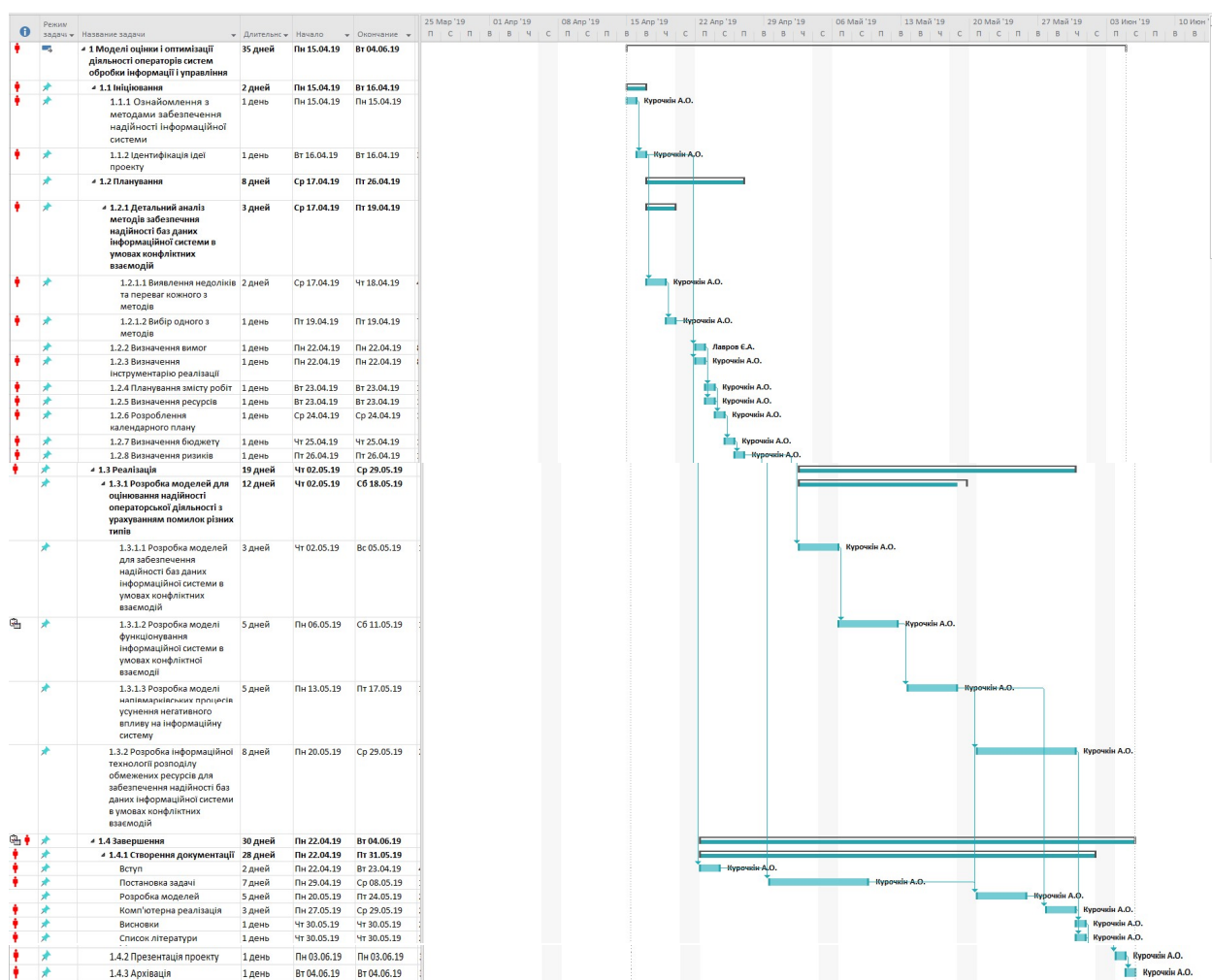


Рисунок Б.3 – Діаграма Ганта

**Аналіз ризиків.** Після визначення всіх структурно-організаційні структур, починаємо виконувати процес визначення та аналізу потенційних проблем, які можуть негативно вплинути на ключові ініціативи бізнесу або критичні проекти, щоб допомогти уникнути або пом'якшити ці ризики.

Визначимо ризики:

- R1 – зміна ТЗ при реалізації проекту;
- R2 – невчасне виконання пакету робіт проекту;
- R3 – виявлення помилок після етапу реалізації;
- R4 – розбіжність версій ПЗ;
- R5 – хвороба розробника.

У таблиці Б.3 показана ймовірність виникнення кожного ризика.

Таблиці Б.3 - Ймовірність виникнення ризиків

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

У таблиці Б.4 показані втрати, які можуть виникнути при появі кожного з зазначених ризиків.

Таблиця Б.4 – Втрати при виникненні ризиків

Значимість впливу	R1	R2	R3	R4	R5
мінімальна					
низька					
середня					
висока					
максимальна					

Виходячи з цих двох, була побудована Матриця впливу (ймовірностей і наслідків) ризиків. В матриці світлим кольором позначено неважливі ризики-білий колір, темнішим – помірні, темним – критичні.

Таблиця Б.5 – Матриця ймовірність-втрати

Ймовірність			
			R1
	R5	R2	R4
			R3
	Втрати		

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

- R1 – зміна ТЗ при реалізації проекту;
- R4 – розбіжність версій ПЗ;

Перший ризик можна уникнути, остаточно визначивши вимоги з керівником.

Другий ризик може привести до некоректної роботи інформаційної технології.

## ДОДАТОК В. 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[,] finanses(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 RoundTo2(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[,] Find_Classes(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[,] Composition(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];
    }
}
RoundTo2(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;
        }
    }
    RoundTo2(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);
RoundTo2(norm_ratio);

```

```

double[,] ham_dist = fuzzy_sim(norm_ratio);
RoundTo2(ham_dist);
double[,] fuzzy_tranz;

```

```

double[,] temp = ham_dist;

```

```

int check, iter = 0;

```

```

do
{
    check = 0;
    fuzzy_tranz = Composition(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 = Find_Classes(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 = finanses(fuzzy_tranz);

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

for (int i = 0; i < results.Count(); i++)
{
    Financee.Rows.Add();
    Financee[0, i].Value = (i + 1);
    Financee[1, i].Value = results[i];
    Financee[2, i].Value = C0_Value;
    Financee[3, i].Value = results[i] * C0_Value;
}
}

private void button2_Click(object sender, EventArgs e)
{
    MessageBox.Show("Для выполнения ранжирования элементов системы необходимо:"
        + "\n1)Выбрать количество элементов"
        + "\n2)Ввести экспертную оценку влияния элементов друг на друга"
        + "\n3)Ввести индексы элементов с наименьшим влиянием на другие элементы"
        + "\n(например: для x1-индекс 2, это значит, что x1 наименее влияет на x2)"
        + "\n4)Нажать на кнопку `Расчитать`");
}

private void min_i_CellContentClick(object sender, DataGridViewCellEventArgs e)
{
    int help;
    if (start > 0)
    {
        for (int i = 0; i < input_values.Rows.Count; i++)
        {
            for (int j = 0; j < input_values.Columns.Count; j++)
            {
                if (i != j)
                    input_values.Rows[i].Cells[j].ReadOnly = false;
            }
        }
    }
}

```





## ДОДАТОК Г. КОМП'ЮТЕРНІ ЕКСПЕРИМЕНТИ

Ранжирование элементов

Размерность массива: 4

Очистить Рассчитать

№	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

№	Коэффициент 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 Г.1 – Виконання експерименту із кількістю баз даних 4

Ранжирование элементов

Размерность массива: 5

Очистить Рассчитать

№	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

№	Коэффициент 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 Г.2 - Виконання експерименту із кількістю баз даних 5

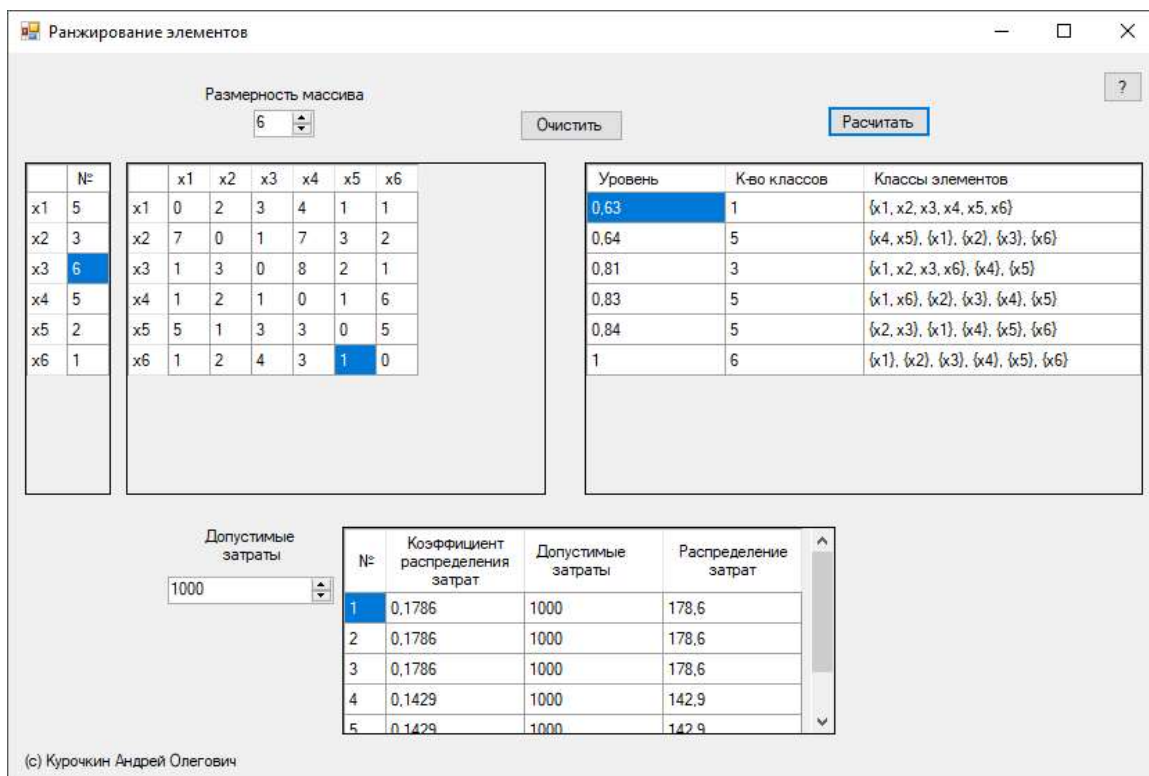


Рисунок Г.3 - Виконання експерименту із кількістю баз даних 6

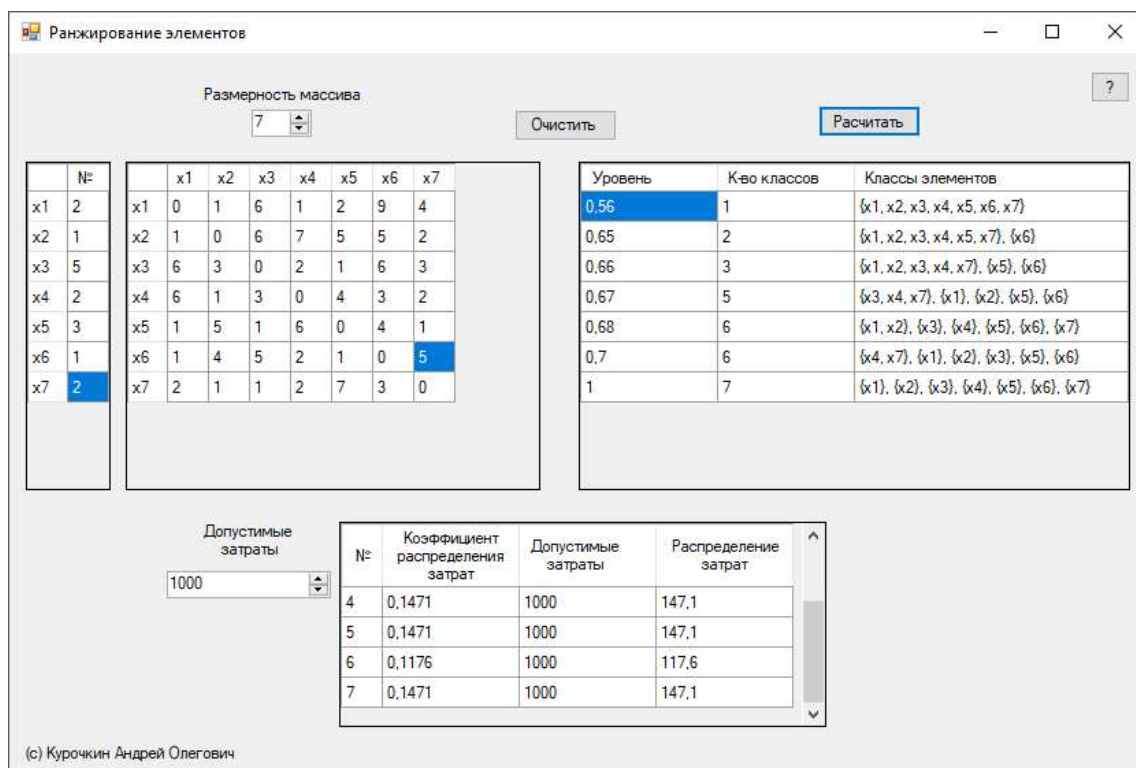


Рисунок Г.4 - Виконання експерименту із кількістю баз даних 7

Ранжирование элементов

Размерность массива: 8

Очистить Расчитать

№	x1	x2	x3	x4	x5	x6	x7	x8
x1	5							
x2	3							
x3	8							
x4	1							
x5	2							
x6	3							
x7	1							
x8	6							

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 Г.5 - Виконання експерименту із кількістю баз даних 8

Ранжирование элементов

Размерность массива: 9

Очистить Расчитать

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

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 Г.6 - Виконання експерименту із кількістю баз даних 8

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}, {x...}

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 Г.7 - Виконання експерименту із кількістю баз даних 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}, {x...}

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 Г.8 - Виконання експерименту із кількістю баз даних 11

Ранжирование элементов

Размерность массива: 12

Очистить      Рассчитать

№:	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12
x4 6	1	7	7	6	6	3	6	5	1	
x5 7	8	1	2	2	3	4	6	5	2	
x6 2	0	6	5	1	9	9	2	3	5	
x7 3	5	0	1	1	3	8	8	7	2	
x8 4	2	6	0	7	1	6	6	1	2	
x9 7	6	3	4	0	1	8	1	6	8	
x10 8	1	7	4	6	0	8	8	1	7	
x11 1	3	1	4	6	7	0	5	4	2	
x12 2	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 Г.9 - Виконання експерименту із кількістю баз даних 12

Ранжирование элементов

Размерность массива: 13

Очистить      Рассчитать

№:	x6	x7	x8	x9	x10	x11	x12	x13
x5 2	6	7	2	3	9	2	6	8
x6 3	8	7	6	3	6	4	8	6
x7 6	4	4	7	5	2	8	6	7
x8 9	1	1	7	8	8	7	6	2
x9 10	4	8	7	2	6	6	4	7
x10 11	0	5	9	8	7	5	2	4
x11 12	1	0	9	6	6	3	1	6
x12 13	1	3	0	1	5	6	6	2
x13 10	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}, {x13}
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 Г.10 - Виконання експерименту із кількістю баз даних 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 Г.11 - Выполнение эксперимента с количеством баз данных 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 Г.12 - Выполнение эксперимента с количеством баз данных 15

## **ДОДАТОК Г. АКТ ВПРОВАДЖЕННЯ**

Акт впровадження за темою:

«Вибір технологій забезпечення цілісності баз даних інформаційної системи в умовах конфліктної взаємодії»

**Затверджую**  
Перший проректор  
Сумського державного  
Університету  
\_\_\_\_\_ Карпуша В.Д.  
« \_\_\_\_\_ » \_\_\_\_\_ 2019 р.

**Акт**  
**впровадження в навчальних процес**  
**СУМСЬКОГО ДЕРЖАВНОГО УНІВЕРСИТЕТУ**  
**результатів наукової роботи**  
студента групи ІТ-51 Сумського державного університету  
Курочкіна Андрія Олеговича  
на тему

«Вибір технологій забезпечення цілісності баз даних інформаційної системи в умовах  
конфліктних взаємодій»

Складений 5 січня 2019 р. комісією у складі:

**Голова комісії:**

*Доцент кафедри комп'ютерних наук, зав. Секції «Інформаційні технології проектування», кандидат технічних наук, доцент Шендрик В.В.*

**Члени комісії:**

1. Професор кафедри комп'ютерних наук, доктор технічних наук, професор Лавров С.А.
2. Доцент кафедри комп'ютерних наук, кандидат технічних наук, доцент Чибіряк Я.І.
3. Старший викладач кафедри комп'ютерних наук, кандидат технічних наук Кузнецов Е.Г.

В період з 3 січня 2019 р. по 5 січня 2019 р. комісія провела роботу з визначення впровадження результатів Курочкіна А.О. в навчальний процес кафедри комп'ютерних наук.

**Результати роботи комісії**

1. На кафедру комп'ютерних наук передано комплекс програм «Вибір технологій забезпечення цілісності баз даних інформаційної системи в умовах конфліктних взаємодій».
2. Матеріали використані в дисциплінах:
  - «Системи підтримки прийняття рішень» для слухачів магістратури, що навчаються за спеціальністю «Інформатика», при розробці теми «Прийняття рішень в умовах ризику» (лабораторна робота – 2 год.).
  - «Організація людино-машинної взаємодії» для слухачів магістратури, що навчаються за спеціальністю «Інформаційні технології проектування», при розробці теми «Ергономіка автоматизованих виробництв» (лабораторна робота – 2 год.).

Голова комісії  
Члени комісії

*Шендрик В.В.*  
*Лавров С.А.*  
*Чибіряк Я.І.*  
*Кузнецов Е.Г.*



## ДОДАТОК Д. 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



*Новіков*

О.М. Новіков  
05 квітня 2019 р.

